Table of Contents

International Journal of Mobile Human Computer Interaction

Volume 8 • Issue 1 • January-March-2016 • ISSN: 1942-390X • eISSN: 1942-3918
An official publication of the Information Resources Management Association

Editorial Preface

Research Articles

COPYRIGHT

The *International Journal of Mobile Human Computer Interaction* is indexed or listed in the following: ACM Digital Library; Bacon's Media Directory; Cabell's Directories; Compendex (Elsevier Engineering Index); DBLP; GetCited; Google Scholar; HCIBIB; INSPEC; JournalTOCs; Library & Information Science Abstracts (LISA); MediaFinder; Norwegian Social Science Data Services (NSD); PsycINFO®; SCOPUS; The Index of Information Systems Journals; The Standard Periodical Directory; Ulrich's Periodicals Directory

International Journal of Mobile Human Computer Interaction
Volume 8 • Issue 1 • January-March 2016

BezelCursor:
Bezel-Initiated Cursor for One-Handed Target Acquisition on Mobile Touch Screens

Wing Ho Andy Li, City University of Hong Kong, Kowloon, Hong Kong
Hongbo Fu, City University of Hong Kong, Kowloon, Hong Kong
Kening Zhu, City University of Hong Kong, Kowloon, Hong Kong

ABSTRACT

The authors present BezelCursor, a novel one-handed thumb interaction technique for target acquisition on mobile touch screens of various sizes. Their technique combines bezel-initiated interaction and pointing gesture to solve the problem of limited screen accessibility afforded by the thumb. With a fixed, comfortable grip of a mobile touch device, a user may employ the tool to easily and quickly access a target located anywhere on the screen, using a single fluid action. Unlike the existing technologies, the authors' technique requires no explicit mode switching to invoke and can be smoothly used together with commonly adopted interaction styles such as direct touch and dragging. Their user study shows that BezelCursor requires less grip adjustment, and is more accurate or faster than the state-of-the-art techniques when using a fixed secure grip.

KEYWORDS

Bezel, Mobile Devices, One-Handed Interaction, Target Acquisition, Thumb Interaction, Touch-Screen

INTRODUCTION

It is reported that a user generally prefers one-handed interaction with a mobile touch device whenever possible (A.K. Karlson, Bederson, & Contreras-Vidal, 2007). That is, only a single hand is used to both hold and interact with the device (Figure 1 (left)). Unimanual interaction allows the user to operate the mobile device in a distracted, multitasking scenario and frees the other hand for tasks like carrying a bag, writing a relevant note etc. In such scenarios, the thumb of the hand holding the device is normally the only available finger for touch input (Boring et al., 2012; Hirotaka, 2003). However, mainly due to biomechanical limitations of the thumb, only a subregion of the touch screen is comfortable to access by the thumb (A. Karlson & Bederson, 2007) causing awkward hand postures

DOI: 10.4018/IJMHCI.2016010101

to reach the rest of the screen (Figure 1 (right)). This problem of limited screen accessibility by the thumb deteriorates with screens of increasingly bigger sizes, which, however, are getting more and more popular (FINGAS, 2012).

Redesigning UIs tailored for one-handed interaction is certainly a possible solution but suffers from a few problems, e.g., causing additional constraints on interface design, sacrificing naturalness of direct manipulation. In contrast, designing new thumb interaction techniques which can be directly applied to existing UIs is more practical. Several one-handed target acquisition techniques such as *ThumbSpace* (A. Karlson & Bederson, 2007) and *MagStick* (Roudaut, Huot, & Lecolinet, 2008) have been proposed and allow the thumb to virtually access anywhere on a screen of moderate size. However, they are not compatible with commonly used interaction styles such as direct touch and dragging, forcing users to switch between input modes explicitly (Amy K. Karlson & Bederson, 2008), e.g., via selecting modes via buttons.

To address the above problems, we present *BezelCursor*, a new technique for one-handed target acquisition technique on mobile touch screens. Our solution is inspired by bezel-initiated interaction and pointing gesture. BezelCursor allows the user to select any object on the screen using a single fluid action, which is a seamless integration of bezel swiping for tool invocation and virtual pointing for target acquisition. More specifically, as illustrated in Figure 2, the user swipes with the thumb of the holding hand, from the outside of the physical bezel of the touch screen to the on-screen target in order to activate the BezelCursor. A visible cursor will be shown and pushed towards the target by the thumb's movement, similar to the use of the touch screen as a trackpad. The cursor will select the underneath element and disappear (deactivated) when the controlling thumb is lifted up.

BezelCursor can be invoked and used without any explicit mode switching. As a secondary input method our technique is complementary to the existing direct touch interaction paradigm. For example, it might be smoothly used together with direct touch, the former for distant targets and the latter for targets within thumb reach, as demonstrated in the accompanying video (*BezelCursor*, 2014). Our user study shows that BezelCursor allowed securer grip with fewer posture adjustments. With a fixed secure grip, BezelCursor gave significantly lower error rate than direct touch and TapTap, significantly faster than MagStick. It was also significantly more accurate than ThumbSpace for picking small targets.

Figure 1. One-handed mobile interaction makes the thumb be the only finger for input and only a subregion of the screen, shown as the region with dashed line, is easily accessible with a comfortable grip of the device (left). Awkward hand postures to access the rest of the screen cause more fatigue, less precision, and unstable grip of the device (right).

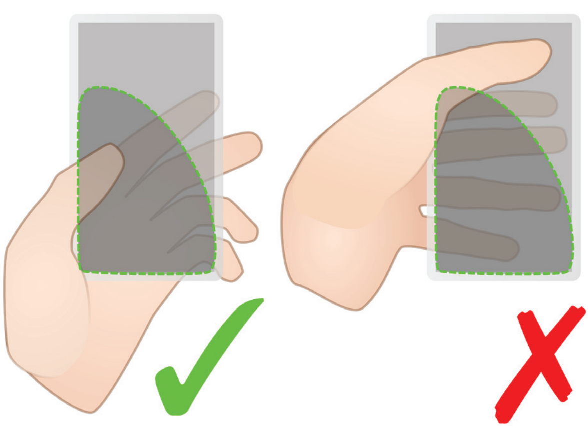
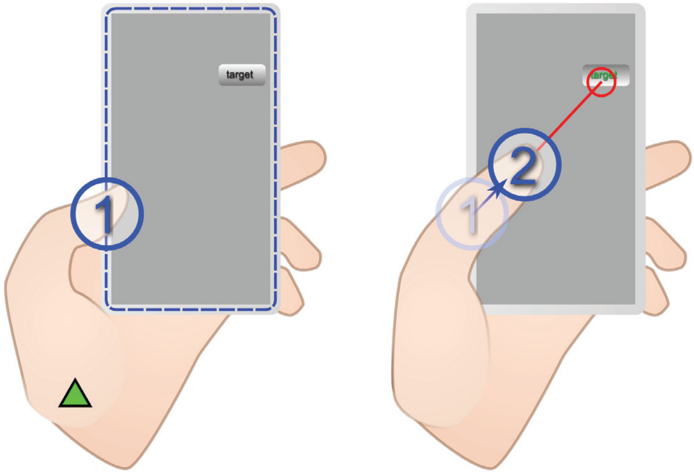


Figure 2. BezelCursor allows a user to easily access a target located anywhere on the screen, by combining bezel swipe for invocation (left) and pointing gesture for acquisition (right) into a single fluid action



RELATED WORK

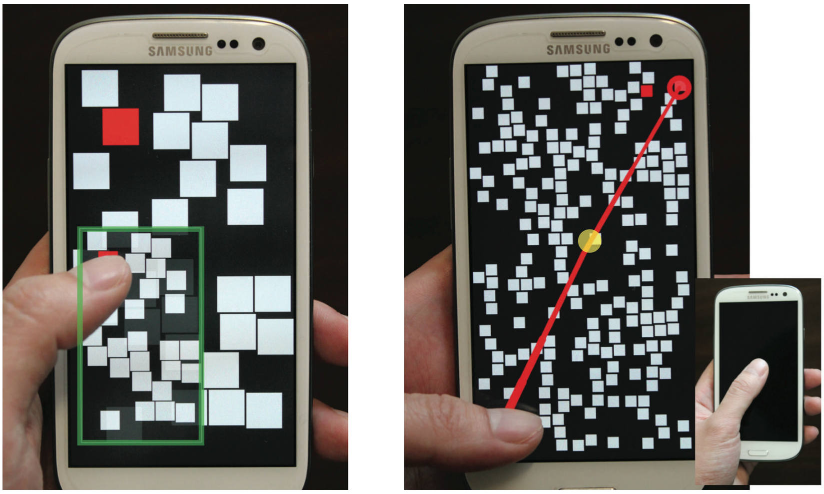
One-handed target acquisition on mobile touch screens raises several issues, including screen accessibility, visual occlusion, and accuracy. Since our focus is on screen accessibility, reviewing the techniques for reducing visual occlusion and/or improving accuracy is beyond the scope of this paper. In fact, we believe that our method can and should be used together with the techniques such as *Shift* (Vogel & Baudisch, 2007) and *Escape* (Yatani, Partridge, Bern, & Newman, 2008) in practice, which are designed for fast and accurate selection of targets within thumb reach.

Little attempt has been made to the problem of reaching distant objects that are out of thumb reach and thus require grip adjustments for direct access, often causing momentary device instability. Serving as an absolute touchpad superimposed on the screen, ThumbSpace proposed by Karlson and Bederson (A. Karlson & Bederson, 2007) adopts a user-defined thumb reachable area as input space (the rectangle in green in Figure 3 (left)), which is mapped to the entire screen for accessing targets located anywhere on the screen. ThumbSpace offers better screen accessibility but is much slower than direct touch. It is thus suggested to use ThumbSpace with other target acquisition techniques, e.g., ThumbSpace for distant targets and Shift for near objects.

However, ThumbSpace requires explicit mode switching and the constant cost of launching ThumbSpace (e.g., pressing the center of a DPad) might make users less willing to use it (Amy K. Karlson & Bederson, 2008). In addition, the superimposed input space is visually distracting and its manipulation causes thumb occlusion (Figure 3 (left)).

Roudaut et al. present two one-handed target acquisition techniques: *TapTap* and *MagStick* (Roudaut et al., 2008). Although it is said that their techniques support easy access to the entire touch screen, this statement is true only for screens of small size (only 2.8-inch touch screen used in their experiments). TapTap involves two sequential taps, with the first tap required to be reasonably close to a target. TapTap's screen accessibility is thus roughly the same as that supported by direct touch. MagStick employs a two-part telescopic stick to control a magnetized offset cursor, which starts at a reference point (in yellow in Figure 3 (right)) specified by a thumb tap and moves in the opposite direction of the thumb dragging. Due to a 1:1 correspondence between motor and display movement, the thumb's reach limit is at most doubled. In addition, since TapTap interferes with direct touch and

Figure 3. Left: ThumbSpace. A user-defined thumb reachable input area (the rectangle in green) is displayed after pressing an activation button. Pressing onto the input area hides the input area and highlights the object associated with the touch location. Use dragging to move the object cursor to the desired target, and lift the thumb to perform the selection. Right: MagStick. Press on a point near the desired target to set a reference point (the yellow dot at the center). Drag away from the target, and a two-part stick then appears and moves in the opposite direction of the thumb. The cursor always snaps to the nearest target. Lift the thumb to perform the selection.



MagStick is incompatible with traditional dragging-based operations, explicit mode switching needs to be addressed to integrate them into existing touch-based interaction systems.

This work has been initially presented as a demonstration in the Symposium on Mobile Graphics and Interactive Applications at SIGGRAPH Asia 2013 (Li & Fu, 2013). *Concurrently* with (Li & Fu, 2013), Kim et al. addressed our problem in a similar perspective (Kim, Yu, & Lee, 2012). By examining two triggering methods (edge and large touch) and two acquisition methods (extensible cursor and sliding screen), they concluded that edge triggering with extensible cursor, similar to our BezelCursor design, is both effective and favorable. However, no study was conducted to evaluate the effectiveness of their techniques in comparison to the state-of-the-art techniques such as MagStick and ThumbSpace. Such an evaluation, together with the resulting findings, is one of the main contributions of this work.

Bezel-initiated interaction invokes a certain action, when a bezel swipe gesture is detected. For example, Bragdon et al. introduce a set of bezel-initiated gestures to be used in diverse environments with various distraction levels (Bragdon, Nelson, Li, & Hinckley, 2011). Their work has been recently extended by Jain and Balakrishnan, who present bezel menus for eyes-free interaction (Jain & Balakrishnan, 2012). Due to their specific applications, these two solutions are largely target-free and thus cannot be applied to our problem. Our work is more relevant to Bezel Swipe (Roth & Turner, 2009), which supports multiple target selection and other operations such as copying and pasting. However, since Bezel Swipe requires direct-touch-like interaction to select targets after invocation, it suffers from the problem of screen accessibility caused by one-handed target acquisition. In fact, only bimanual interaction is demonstrated in their experiments. In addition, Bezel Swipe is susceptible to finger occlusion.

Target acquisition on large, wall-sized displays suffers from a similar screen accessibility problem, where targets are often out of arm's reach, rendering acquisition techniques based on absolute mapping between the motor and display space (e.g., Shift) inefficient or even infeasible. To address this problem, Baudisch et al. propose *Drag-and-Pick*, which brings distant objects located in direction of drag motion close to the user (Baudisch et al., 2003). Bezerianos and Balakrishnan present *Vacuum*, a distance reaching technique similar to *Drag-and-Pick* but supporting more operations like easy selection of multiple targets (Bezerianos & Balakrishnan, 2005). While these two techniques are promising, it is unclear how to seamlessly integrate them with acquisition techniques based on direct touch or absolute pointing, which are more natural to operate targets within reach. Forlines et al. introduce *HybridPointing* to enable fluid switching between absolute and relative pointing (Forlines, Vogel, & Balakrishnan, 2006). However, their method needs explicit mode switching (via the so-called *trailing widget*) and involves a rather complicated state transition model, requiring the user to understand the current input state based on visual feedback. Finally, the use of our BezelCursor essentially leads to a pointing gesture, which is intuitive to use and has been extensively studied for interaction with large displays from a distance (e.g. (Vogel & Balakrishnan, 2005)).

DESIGN AND IMPLEMENTATION

In this section we first give an overview of BezelCursor design and then discuss its implementation details.

Design Overview

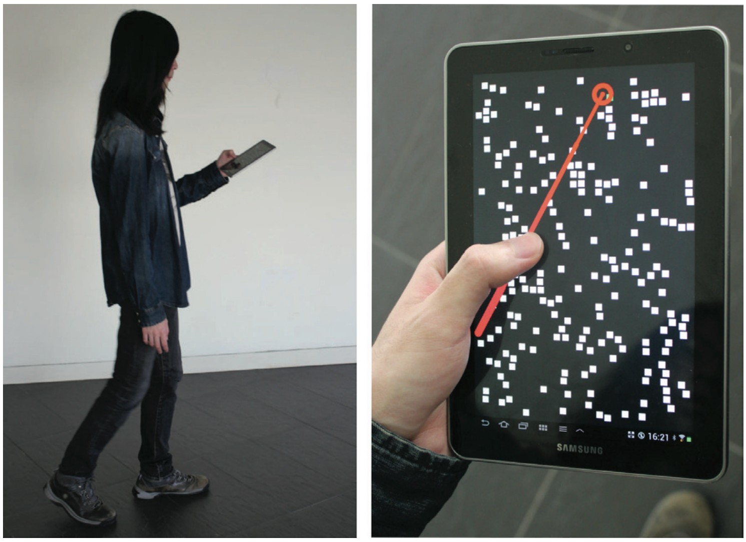
The bezel of a device is the physical touch insensitive frame surrounding its touch screen, as illustrated in Figure 2 (left), i.e., the dotted rectangle in blue. BezelCursor is invoked by swiping the thumb from the bezel inwards to the screen (see a live demo in the accompanying video (*BezelCursor*, 2014)). The user then keeps dragging the thumb to drive a cursor towards the target (Figure 2 (right)). BezelCursor can be dismissed anytime by lifting up the thumb, followed by selection validation.

Despite its simplicity in design, BezelCursor has the following nice properties:

- **Supporting One-Handed Thumb Interaction:** For almost all commonly used hand postures for holding the device with a single hand, at least one of the four bezel edges can be easily accessed by the thumb for tool invocation.
- **Single Fluid Action:** The adoption of the *take-off* selection paradigm (Potter, Weldon, & Shneiderman, 1988) allows BezelCursor seamlessly combines invocation, selection, and dismissal into a single fluid action, resulting in low overhead to use it.
- **Compatible with Existing Interaction Styles:** Since the invocation strategy, bezel swipe, is not commonly used, it can be employed together with major used interaction styles such as direct touch and dragging, without explicit mode switching.
- **Intuitive to Use:** The use of BezelCursor is analogous to that of pointing gesture and thus can be easy to learn.
- **Minimal Visual Distraction:** BezelCursor is unobtrusive when not needed and BezelCursor in action visually causes nothing except for a cursor and an optional stick going from the point of invocation to the cursor, as shown in Figure 2 (right).
- **Scalable to Screens of Large Size:** By properly mapping thumb movement to cursor movement, BezelCursor is applicable to 7-inch or even larger tablets (Figure 4), which would be difficult to fully handle by MagStick or TapTap.
- **No Thumb Occlusion:** The use of relative pointing essentially eliminates finger occlusion.

It is reported by Karlson et al. that certain thumb motions, e.g., moving the thumb's tip to and away from the palm, were ergonomically more difficult (A.K. Karlson, Bederson, & Contreras-Vidal, 2006). To perform BezelCursor in a way as shown in Figure 2, the user may temporarily move the carpometacarpal (CMC) joint (i.e., the joint enabling thumb rotation around the wrist, as illustrated by a triangle in green in Figure 2 (left)) slightly away from the device before tool invocation. Note that moving the CMC joint is often needed for performing MagStick as well. We found that such a grip is

Figure 4. BezelCursor is applicable to one-handed target acquisition even for a 7-inch device while walking (see the accompanying video (BezelCursor, 2014))

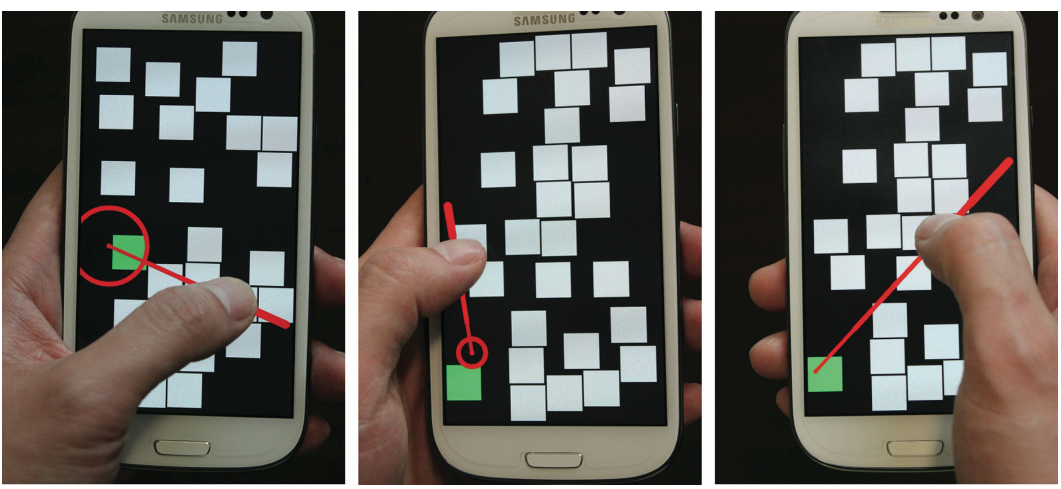


still secure and is able to use BezelCursor in mobile environments, e.g., while walking (Figure 4). In addition, the user may start BezelCursor from different parts of the bezel to select a given target. For example, the user may choose to rotate the thumb around the wrist to operate BezelCursor. Figure 5 shows such three examples.

Cursor Movement Control

BezelCursor maps small movements of the thumb to large movements of the cursor. This is the key to fluid access to all areas of the screen using the areas within the thumb's reach as input space. We have experimented two simple mapping functions in our implementation: *linear function* and *acceleration function*, analogous to computer mouse control without and with acceleration, respectively.

Figure 5. BezelCursor in action. It can be used with either hand, and activated from any side of the screen.



The first mapping function leads to a completely linear cursor response. Specifically it is formulated as $p_c = s \cdot \left(p_t - p_i\right) + p_i$, where p_i, p_t and p_c are the positions of the point of invocation on the bezel, the current touch point of the thumb, and the cursor, respectively. s (> 1) is a scale factor that can be customized based on the ratio of the length of the thumb to the size of the touch screen. In our experiments we always set $\mathrm{s} = 3.0$ for Samsung Galaxy S III (with 4.8-inch display) as the main testing device. The linear mapping function gives easily predictable control. However, it is theoretically less accurate for selecting targets of small or tiny size, since the thumb's movement always gets magnified.

The acceleration mapping function is introduced to solve this problem by varying the control-display (CD) gain between the thumb and cursor movement as a function of thumb movement velocity. More specifically, instead of a pre-defined value for s of the linear mapping function, the scale factor will be dynamically dependent on the current velocity of the touch point. The user is thus able to perform fine cursor adjustment by slowly moving her thumb. Initially $s = 1.5$. It linearly increases when touch velocity is at $0.3\ \mathrm{inch} / \sec$, until velocity is at or over $1.5\ \mathrm{inch} / \sec$, $\mathrm{s} = 3.5$. The values are set based on our initial testing with the prototype application.

Accessing Near-Bezel Targets

Through our preliminary experimentation, we found that the linear mapping function might make BezelCursor relatively difficult to select targets near the edge of the bezel where BezelCursor is activated. This is mainly because moving the cursor to the activation bezel requires user to touch very closely to the bezel. However, pixels close to the bezel are usually hard to touch due to the fact that a touch point is always represented as the center of the contact area of the thumb with the touch-sensitive display but excluding the contact area with the touch-insensitive bezel (Roth & Turner, 2009). To ease the selection of targets near the bezel edge of invocation, we slightly shift the position of p_i away from the point of invocation (0.2 inches used in our experiments). Note that this effect is less significant when using the acceleration mapping function, since the user could always vary the speed of thumb movement such that the cursor reaches the bezel edge of invocation prior to the thumb. Moreover, in practical deployment of BezelCursor, for the parts of bezels that are within thumb reaching distance, the user may simply use direct tapping instead of invoking BezelCursor. However, we observed that some prototype testers with larger hands preferred to use the bezel from the opposite side of the holding hand to activate BezelCursor to select out-of-reach targets on the top near to the same bezel. Some testers with smaller hands were not able to reach the upper part of the bezel near the holding hand. These suggested the importance of this adjustment.

Integration with Area Cursors

BezelCursor can be easily integrated with many existing cursor-based target acquisition techniques. In our experiments, we have tested its integration with the state-of-the-art techniques: *Bubble Cursor* (Grossman & Balakrishnan, 2005) and *DynaSpot* (Chapuis, Labrune, & Pietriga, 2009). Both of them are based on the idea of dynamic area cursor, with the former varying the cursor activation area with respect to the proximity of surrounding targets (Figure 5 (left)) and the latter changing the cursor size according to the speed of the cursor (Figure 5 (middle) and (right)). The integration of both Bubble Cursor and DynaSpot into our tool is rather straightforward.

EVALUATION

We have conducted a user study, which consisted of three parts: user study 1, 2, and 3. The first part was to test the effectiveness of BezelCursor in different settings. The second and the third parts were to compare BezelCursor with the state-of-the-art techniques to show the advantages of BezelCursor qualitatively and quantitatively, respectively. Following will first introduce the basic setup and procedure used in all the three parts of the user study.

Apparatus: Samsung Galaxy S III, which was one of the most popular smartphones, was adopted in the user study. Running Android 4.0, this device has a 4.8-inch capacitive touch screen with 1280×720 pixel resolution, making one-handed direct touch difficult to access distant area of the screen. Our application was written in Haxe and compiled to a native Android application utilizing the Android NDK. We based graphics rendering on OpenGL and set a frame rate of 30.

Tasks: Participants were required to hold the device in portrait orientation with a single hand comfortably, and to use only the thumb of the hand holding the device for touch input. There was no additional support (e.g. table) for the device or their hands. While either sitting or standing, participants were then instructed to employ a given interaction technique to perform single selections of screen targets one by one as quickly and accurately as possible. During operation, they were asked not to significantly change their initial holding postures by moving their fingers holding the device, unless it would be possible to reach the targets.

User Study 1

This part of the user study aimed to evaluate the performance of BezelCursor using different mapping functions (linear or acceleration) for cursor movement control and area cursors methods (Bubble Cursor or DynaSpot), leading to four techniques (i.e., 2×2 combinations) to be tested using the above evaluation procedure. Sixteen unpaid volunteers (4 female, 12 male) participated in the study. 4 out of 16 participants preferred to use their left hands to operate the device. All of the participants were experienced touch-enabled smartphone users. In total, there were

$$16(\text{participants}) \times 4(\text{techniques}) \times 12(\text{target areas}) \times 2(\text{target sizes}) \times 3(\text{blocks}) = 4608\,\text{trials}$$

All participates finished the tasks within half an hour, including optional breaks (typically less than 1 minute for each).

We studied 2 *target sizes* with squared shape, small ($3mm \times 3mm$) and large ($9.6mm \times 9.6mm$). The former (Figure 3 (right)) corresponds to the actual minimal widget size in mobile applications (Ren & Moriya, 2000; Vogel & Baudisch, 2007) while the latter (Figure 3 (left)) corresponds to the recommended target size for one-handed direct touch input without degrading performance and preference for serial tasks (Parhi, Karlson, & Bederson, 2006).

Similar to the setup of Karlson et al. (Amy K. Karlson & Bederson, 2008), we arranged targets to be selected using a 3×4 uniform grid of *target areas*. As shown in Figure 3 each trial involved a single target (in red) placed inside one of the target areas with a random position, and a set of distracters (in white) randomly spread in the rest of the screen, with minimal separation of $1mm$ to each other, leading to a non-uniform layout space. The size of the distracters was set the same as that of the target. For trials with small targets, there were in total 199 distracters in each trial. For trials with large targets, there were 27 distracters in each trial. The color of a target changed to green when the cursor hovered above it. Once a selection was made, a message was displayed to indicate whether the selection was successful or not. A new target would be generated after the message disappeared.

A repeated measures within-subject experiment was conducted. The presentation of the BezelCursor variants, and target sizes were randomized for each participant. For a given BezelCursor variant, each participant was asked to complete 3 blocks of 12 trials with respect to the 12 target areas for one target size, followed by another 3 blocks of 12 trials for the other target size. The 12 target area trials were randomized within blocks. Before starting the experiment with a new acquisition technique, participants were briefed on the technique and practiced until they felt comfortable.

Participants were given an optional break between techniques. During the experiment, our system recorded the following information for quantitative analysis: the completion time of individual trials, errors, and thumb movement trajectories.

Overall BezelCursor gave competitive performance. On average it had an error rate of 7.57% for target size of $9.6mm \times 9.6mm$, and 18.35% for target size of $3mm \times 3mm$. The mean of selection time was $1.73s$ for target size of $9.6mm \times 9.6mm$, and $2.72s$ for target size of $3mm \times 3mm$.

Figure 6 plots individual performance results of the four versions of BezelCursor. Repeated measures analysis of variance (ANOVA) found no statistically significant difference in error rate among the 4 combinations $\left(F_{3,140} = 0.54, p = 0.66\right)$.

However, there was a significant difference in the average completion time per trial across the participants $\left(F_{3,140} = 9.09, p < 0.01\right)$.

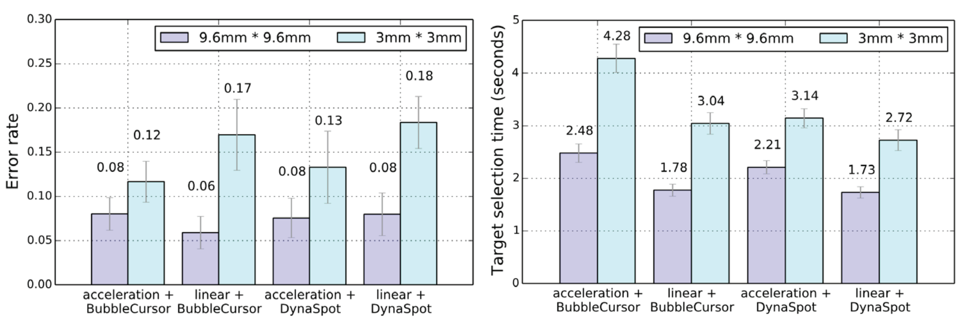
Pair-wise comparisons showed that the combination of acceleration + Bubble Cursor was significantly slower than the others. Among the other three combinations, a pairwise significant difference was found only between linear + DynaSpot and acceleration + DynaSpot: the former was significantly faster. Although the overall performance for linear + DynaSpot and linear + Bubble Cursor was largely the same, we would simply use the former for the default implementation of BezelCursor. We speculate that it might be possible to improve the performance of individual versions of BezelCursor by selecting better parameters, which would be left for future studies.

USER STUDY 2

After selecting the mapping function and area cursor method for BezelCursor, we compared the performance of five techniques: direct touch, TapTap, MagStick, ThumbSpace, and BezelCursor (i.e., with the combination of linear + DynaSpot). As this work focused on the limited thumb reaching distance problem, we chose to compare only the techniques that attempted to solve the same problem. TapTap slightly extends the reaching distance, as it requires a first tap close to but not exactly on the target to bring it up to the center of the screen. MagStick is able to double the reaching distance. ThumbSpace has the same design objective as ours to improve large mobile screen accessibility.

Although integrating other cursor movement mapping functions or area cursors into MagStick and ThumbSpace might improve their performance and/or usability, this is beyond the scope of this research. In fact, the cursor movements and cursor types are well defined for both techniques, and thus choosing another mapping function or area cursor is non-trivial and requires additional testing to ensure no performance or usability degradation. Our implementation of the techniques in our testing application straightly followed their original designs unless explicitly stated otherwise.

Figure 6. The performance of BezelCursor using different cursor mapping functions and different types of area cursor



For TapTap, we set the zooming factor as 3 for both the background and the targets. The original paper set the zooming factors of the background and the targets as 2 and 3, respectively. In our testing application it would cause the targets to overlap with each other (Table 1).

We adopted the same tasks of user study 1. To evaluate the pure performance of each technique, direct tapping on the target was disallowed, except when testing with the direct touch technique. This removed the cognitive overhead of choosing between direct touch and the other techniques. The color of a target changed to green when the touch point of direct touch or TapTap, or the cursor of BezelCursor, MagStick or ThumbSpace (in fine adjustment stage) was over it.

TapTap, MagStick and ThumbSpace are not compatible with common interaction techniques, tapping and/or dragging. We added a semi-transparent on-screen widget ($18mm \times 9mm$) as an activation button to reflect that mode-switching is required in real-life usage of these techniques. Participants had to tap on this widget before using each of the techniques. At the beginning of testing a new technique or target size, the widget was located at the bottom center of the screen, but could be moved within the whole screen by dragging. It was design to avoid consuming any target area. The widget would be hidden once it was tapped, and it reappeared with a next target.

There were two selection timing values recorded for each trial. The first one (t_1) was the time difference between the moment a target appeared and that a selection was made. The second one (t_2) was the time difference between technique activation and selection. The former was recorded when the floating widget was tapped for TapTap, MagStick and ThumbSpace, or the bezel of the screen was touched for BezelCursor. In other words, t_1 included the time of participant looking for the target, and the time of technique activation, but t_2 did not. Although t_2 did not exist for direct touch, we used its values of t_1 to compare with other techniques.

We adopted a repeated measures design. Within-subject independent variables were interaction technique, target size and target area. Dependent variables were target selection times (t_1 and t_2), and error rate of interaction technique. The testing orders of interaction technique and target size

Table 1. Design comparison of BezelCursor and other acquisition techniques, which were used in user study 2 and 3

	Direct touch	BezelCursor	TapTap	MagStick	ThumbSpace
Supporting one-handed thumb interaction	partial (far away area is hard to reach)	Yes	partial (far away area is hard to reach)	Yes	Yes
Scalable to screens of large size	No	Yes	No	At most double the target selection area.	Yes
Compatible with existing interaction styles	Yes	Yes	No (conflict with direct touch)	No (conflict with dragging)	No (requires activation button)
Intuitive to use	Yes	Yes	Yes	movement is opposite to direction to target	Yes
Visual distraction	None	Minimal	High	Minimal	High
Target occlusion	Yes	No	partial (only reduced after zooming in)	No	partial (Target in thumbnail view is occluded. Fine adjustment does not guarantee no occlusion)
Require activation / mode-switching button	No	No	Yes (but could be auto activated when need)	Yes	Yes

were counterbalanced using a Latin square design. The testing order of target area was randomized for each participant.

We hypothesized that the overall performance in terms of selection times and error rate of BezelCursor would be similar to TapTap, MagStick and ThumbSpace; BezelCursor would be slower than direct touch but with lower error rate. For distant areas, BezelCursor would have the lowest error rate among all the tested techniques.

There were twenty new participants, aged between 23 and 43 (mean: 25.9). Among them, 8 were female and 12 were male; all of them were right-handed and 18 of them preferred to use their right hands to operate the device. Again, all of them were experienced touch device users. In total, there were

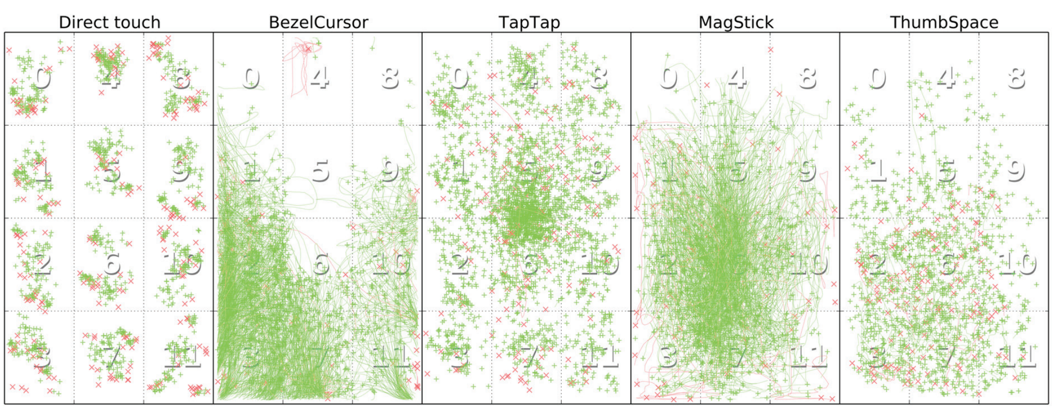
$$20(\text{participants}) \times 5(\text{techniques}) \times 12(\text{target areas}) \times 2(\text{target sizes}) \times 3(\text{blocks}) = 7200\,\text{trials}\,.$$

All participates finished the tasks within 45 minutes.

Figure 7 plots the thumb movement trajectories of individual acquisition techniques. The touch movement data from users using right hand was flipped horizontally for consistent comparison. We observe that there was a strong preference for the bezel edge of BezelCursor invocation, which was highly dependent on the handedness: most of the time BezelCursor was invoked from the left bezel and the left portion of the bottom bezel, for left-handed users. On the other hand, a few participants started BezelCursor from the right bezel to select targets on the right hand side. It is seen that many failed cases for MagStick occurred near the bottom bezel. We speculate that for such cases the reference point was not close enough to the target and thus the magnetized offset cursor could not reach the target even when the thumb almost moved outside the screen.

Grip Adjustments: We observed that when using direct touch and TapTap, most participants adjusted their holding postures to select targets that were out of thumb reach, leading to unstable device grabbing. Most of them kept using a posture similar to the right hand side of Figure 1 even when selecting a near target. Two participants even nearly dropped the device. Much fewer posture adjustments were observed for BezelCursor or ThumbSpace. Although MagStick doubled the reach limit of the thumb, several participants still decided to adjust their holding postures to make the reference point of MagStick closer to distant targets.

Figure 7. Touch movements when using different acquisition techniques during user study 2. Paths of successful selection are shown in green and their touch release points are marked as +. Paths of failed selection are shown in red and their touch release points are marked as ×. The data is normalized to left hand usage.



Participant Feedback: Participants gave qualitative feedback that suggested the weaknesses of each of the techniques. The participants expressed great frustration when using direct touch to select either distant or small targets. Several of them reported that MagStick was not intuitive and hard to use when they were first presented with the method. Many participants also mentioned that the $3mm \times 3mm$ targets mirrored in the touch space of ThumbSpace were too small. Although the participants were instructed how to perform fine adjustment with ThumbSpace by thumb dragging, from our observation most of them seemed not very willing to use this feature, thus causing high error rate for selecting small targets. As expected, for BezelCursor, a number of participants tried to select targets that were very close to the point of invocation, and expressed difficulty, mainly due to finger occlusion. But most of them quickly realized and started to invoke BezelCursor from a distance. This reiterated the importance of using our technique (as well as other distant target acquisition techniques) together with techniques designed for accessing near objects.

Quantitative Performance Comparison: We performed an analysis on the target selection times and error rate. However, BezelCursor did not give very competitive result. Figure 8 and Figure 9 plot the individual performance of the five acquisition techniques.

Figure 8. The error rates of BezelCursor and other acquisition techniques in user study 2

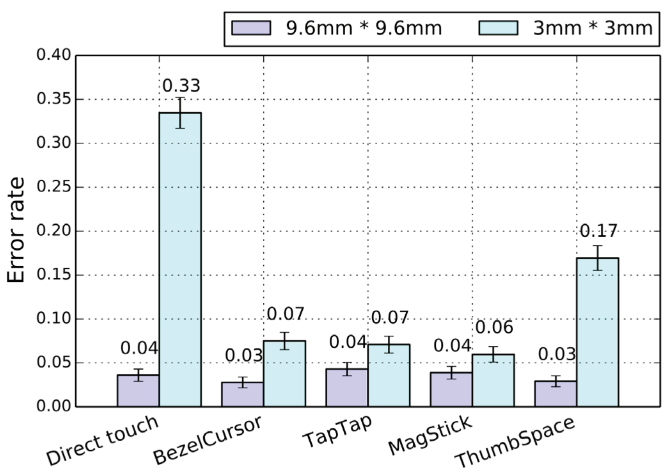
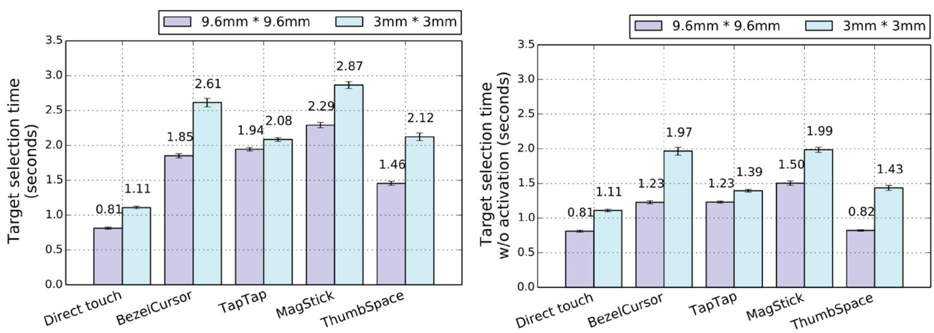


Figure 9. The target selection times of BezelCursor and other acquisition techniques in user study 2, including (t_1, left) and excluding (t_2, right) the technique activation time



For target selection performance, we performed a repeated measures analysis of variance (ANOVA) on the target selection times and the error rates of the 5 techniques. There were significant main effects on selection performance of interaction technique (t_1: $F_{4,72} = 85.51, p < 0.001$; t_2: $F_{4,72} = 45.77, p < 0.001$; error rate: $F_{4,72} = 29.09, p < 0.001$) and target size ($t_1$: $F_{1,18} = 97.08, p < 0.001$; t_2: $F_{1,18} = 105.75, p < 0.001$; error rate: $F_{1,18} = 98.02, p < 0.001$). There were also significant main effects on selection times of target area (t_1: $F_{11,176} = 6.50, p < 0.001$; t_2: $F_{11,176} = 7.76, p < 0.001$). However, the main effect on error rate of target area was not significant (error rate: $F_{4,72} = 1.71, p > 0.075$). Significant interactions between technique and target size, on selection performance was found (t_1: $F_{4,72} = 6.45, p < 0.001$; t_2: $F_{4,72} = 8.88, p < 0.001$; error rate: $F_{4,72} = 28.20, p < 0.001$). There were also significant interactions between technique and target area (t_1: $F_{44,704} = 1.58, p < 0.02$; t_2: $F_{44,704} = 1.94, p < 0.001$; error rate: $F_{44,704} = 1.58, p < 0.02$).

For each of the target sizes, we did post-hoc pairwise comparisons using paired t-tests with Bonferroni P value adjustment. For large targets: No significant different in error rate was found between all the techniques. BezelCursor was significantly faster than MagStick in both t_1 and t_2 (both $p < 0.001$), but not significantly different from TapTap in t_1 or t_2. It was significantly slower than direct touch and ThumbSpace ($p < 0.001$ for both t_1 and t_2). For small targets: the mean of error rates of BezelCursor was significantly lower than direct touch's and ThumbSpace's (both $p < 0.001$). No significant difference in error rate was found between BezelCursor, TapTap and MagStick. BezelCursor was faster than MagStick. It was significant in t_1 ($p < 0.01$) but not in t_2 ($p = 1$). BezelCursor was significantly slower than direct touch, TapTap and ThumbSpace ($p < 0.001$ for both t_1 and t_2).

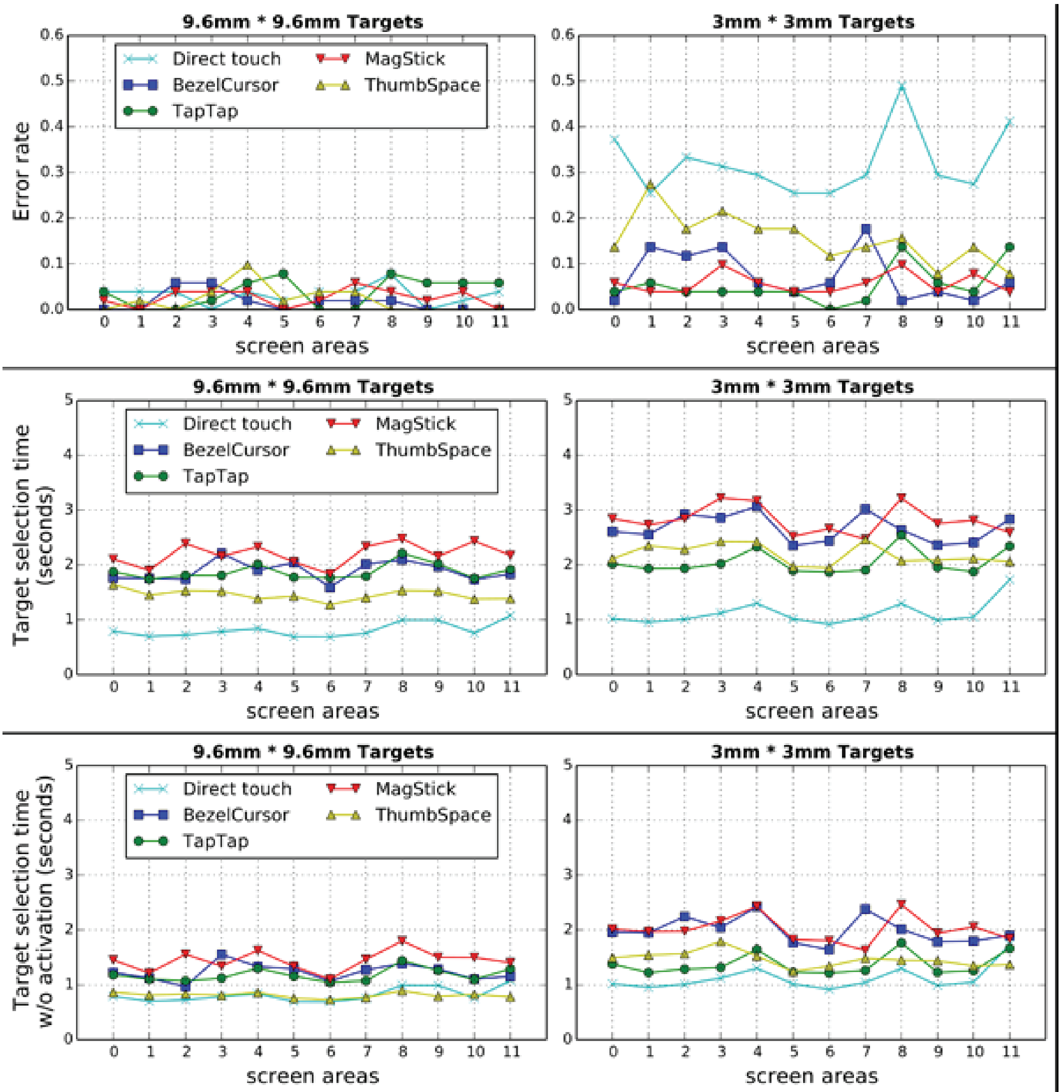
Figure 10 shows the performance of each technique in terms of error rate and target selection time with respect to individual target areas. They did not show much difference across the areas for large targets. For small targets, direct touch was less accurate in area 0, 8, 11. Interestingly BezelCursor was less accurate in relatively near areas (1, 2, 3, 7) but gave excellent accuracy in distant areas (0, 8, 9, 10). TapTap was less accurate in distant areas (8, 11). ThumbSpace was less accurate in area 1. We performed pairwise comparisons using paired t-tests with Bonferroni P value adjustment for each of the target sizes. The result is summarized in Table 2.

We speculate that under the setting of user study 2, the advantage of BezelCursor was mostly reflected from the qualitative results including grip adjustments and participants feedback. Since we let our participants to sit stationary to operate the device, and we allowed them to change gripping postures, they tended to frequently change postures in order to maintain high selection performance. In order to better capture the advantage of BezelCursor also in quantitative selection performance data, we designed user study 3.

USER STUDY 3

User study 3 shared the same task of user study 2. However, we required the participants to use only one secure holding posture for each of the assigned techniques. They were instructed to try different postures during the practice period and were told not to move any of their fingers except their thumb once the study began. They could use any comfortable holding posture in portrait orientation, but the grip should be secure so that the device would not fall even if they turned the holding hand around. For targets that were out-of-reach, the participants were asked to reach them as closely as they could. In this way we avoided recording grip adjustment, which was difficult. In fact grip adjustment consists of a number of properties, including gripping posture, pressure, and duration, which are hard to record

Figure 10. The performance of different methods in individual target areas during user study 2



and classify reliably and quantitatively. Note that since the device could still slip slightly towards the floor during the study due to its weight, we required our participants to adjust their grip to the original posture in this case.

We hypothesized that under this new constraint, the error rates of the five techniques would increase. BezelCursor would have the least error rate. Distant areas would have significantly higher error rate than near areas for the techniques other than BezelCursor.

There were ten new participants, aged between 22 and 27 (mean: 24.8). Among them, 3 were female and 7 were male; all of them were right-handed and 7 of them preferred to use their right hands to operate the device. Again, all of them were experienced touch device users. In total, there were

$$10(\text{participants}) \times 5(\text{techniques}) \times 12(\text{target areas}) \times 2(\text{target sizes}) \times 3(\text{blocks}) = 3600\,\text{trials}\,.$$

Table 2. Summary of the performance difference between BezelCursor and other methods in individual target areas during user study 2. Tick symbols (✓) are which BezelCursor were better (lower t_1, t_2, or error). Cross symbols (×) are which BezelCursor were worse. Double ticks or crosses are statistically significant differences (p<0.05). Dashes (-) are which the means are equal.

size	region	Direct touch			MagStick			TapTap			ThumbSpace		
		t_1	t_2	error	t_1	t_2	error	t_1	t_2	error	t_1	t_2	error
9.6mm * 9.6mm	0	✗✗	✗✗	✓	✓✓	✓✓	✓	✓	✗	✓	✗	✗✗	-
9.6mm * 9.6mm	1	✗✗	✗✗	✓	✓	✓	-	✗	✗	-	✗	✗✗	✓
9.6mm * 9.6mm	2	✗✗	✗✗	✗	✓✓	✓✓	✗	✓	✓	✗	✗	✗	✗
9.6mm * 9.6mm	3	✗✗	✗✗	✗	✗	✗	✗	✗	✗	✗	✗✗	✗✗	✗
9.6mm * 9.6mm	4	✗✗	✗✗	✓	✓✓	✓✓	✓	✓	✗	✓	✗✗	✗✗	✓
9.6mm * 9.6mm	5	✗✗	✗✗	✓	✓	✓	-	✗	✗	✓	✗	✗✗	✓
9.6mm * 9.6mm	6	✗✗	✗✗	✓	✓	✓	-	✓	✗	✗	✗✗	✗✗	✓
9.6mm * 9.6mm	7	✗✗	✗✗	✓	✓	✓	✓	✗	✗	✗	✗✗	✗✗	✓
9.6mm * 9.6mm	8	✗✗	✗✗	✓	✓	✓	✓	✓	✓	✓	✗✗	✗✗	✗
9.6mm * 9.6mm	9	✗✗	✗	-	✓	✓	✓	✓	✗	✓	✗	✗✗	-
9.6mm * 9.6mm	10	✗✗	✗✗	✓	✓	✓	✓	✓	✗	✓	✗✗	✗✗	-
9.6mm * 9.6mm	11	✗✗	✗	✓	✓	✓✓	-	✓	✓	✓	✗✗	✗✗	-
3mm * 3mm	0	✗✗	✗✗	✓✓	✓	✓	✓	✗✗	✗✗	✓	✗	✗✗	✓
3mm * 3mm	1	✗✗	✗✗	✓	✓	✓	✗	✗✗	✗✗	✗	✗	✗	✓
3mm * 3mm	2	✗✗	✗✗	✓✓	✗	✗	✗	✗✗	✗✗	✗	✗	✗	✓
3mm * 3mm	3	✗✗	✗✗	✓	✓	✓	✗	✗✗	✗✗	✗	✗	✗	✓
3mm * 3mm	4	✗✗	✗	✓✓	✓	✓	-	✗	✗	✗	✗	✗	✓
3mm * 3mm	5	✗✗	✗✗	✓✓	✓	✓	-	✗✗	✗✗	-	✗	✗✗	✓
3mm * 3mm	6	✗✗	✗✗	✓	✓	✓	✗	✗	✗✗	✗	✗	✗	✓
3mm * 3mm	7	✗✗	✗✗	✓	✗	✗	✗	✗✗	✗✗	✗	✗	✗	✗
3mm * 3mm	8	✗✗	✗✗	✓✓	✓	✓	✓	✗	✗	✓	✗✗	✗✗	✓
3mm * 3mm	9	✗✗	✗✗	✓✓	✓	✓	-	✗✗	✗✗	✓	✗	✗	✓
3mm * 3mm	10	✗✗	✗✗	✓✓	✓	✓	✓	✗✗	✗✗	✓	✗	✗✗	✓
3mm * 3mm	11	✗✗	✗	✓✓	✗	✗	✗	✗	✗	✓	✗✗	✗✗	✓

All participates finished the tasks within 45 minutes.

Figure 11 plots the thumb movement trajectories of individual acquisition techniques. Compared to Figure 7, the movements were even more concentrated on the lower left portion, suggesting that the thumb reachable area was smaller when using a secure holding posture. One thing that was not obvious in study 2 was that, there were a large number of missed touch points along the bottom-left bezel for direct touch and TapTap. Those were produced when the Mount of Venus accidentally touched the screen before a participant was able to use their thumb tip to reach a far target. This also happened for a few times with BezelCursor, when a participant tried to activate BezelCursor from the right bezel. But they quickly realized the problem and kept using the left and bottom bezel only. In fact, BezelCursor was relatively robust to this problem: once it was activated from the bezel, the Mount of Venus touching the screen could be detected and ignored.

Quantitative Performance Comparison: Figure 12 and Figure 13 plot individual performance of the five acquisition techniques. For each of the target sizes, we did pairwise comparisons using paired t-tests with Bonferroni P value adjustment. For large targets: BezelCursor had

Figure 11. Touch movements when using different acquisition techniques during user study 3. Paths of successful selection are shown in green and their touch release points are marked as +. Paths of failed selection are shown in red and their touch release points are marked as ×. The data is normalized to left hand usage.

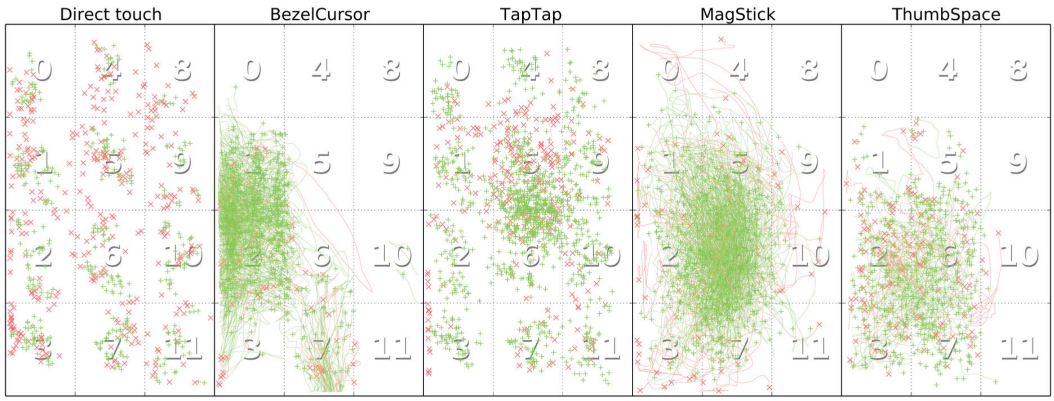


Figure 12. The error rates of BezelCursor and other acquisition techniques in user study 3

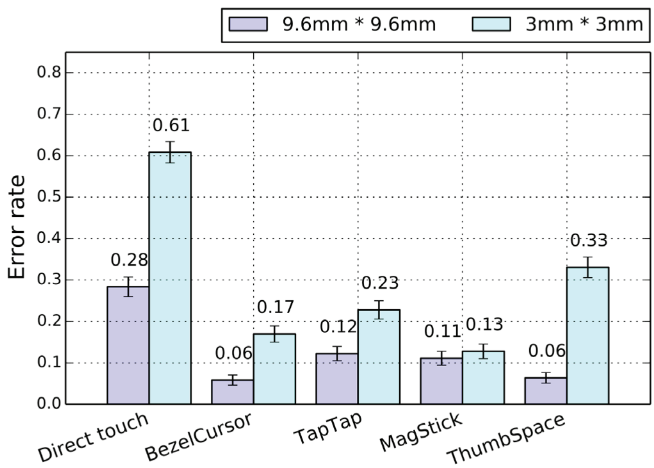
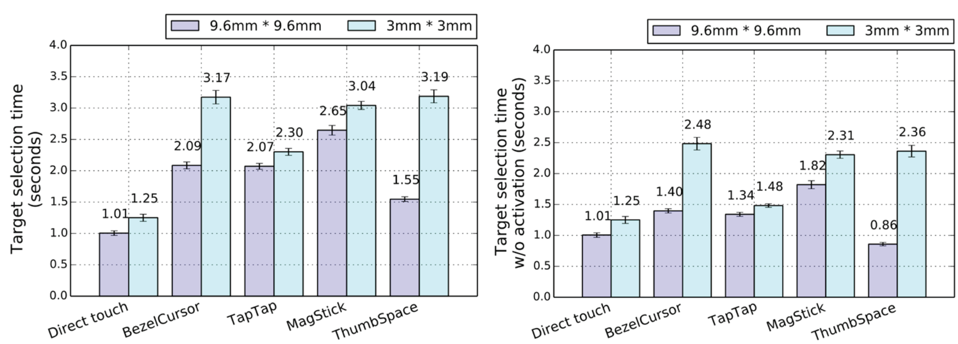


Figure 13. The target selection times of BezelCursor and other acquisition techniques in user study 3, including (t_1 , left) and excluding (t_2 , right) the technique activation time



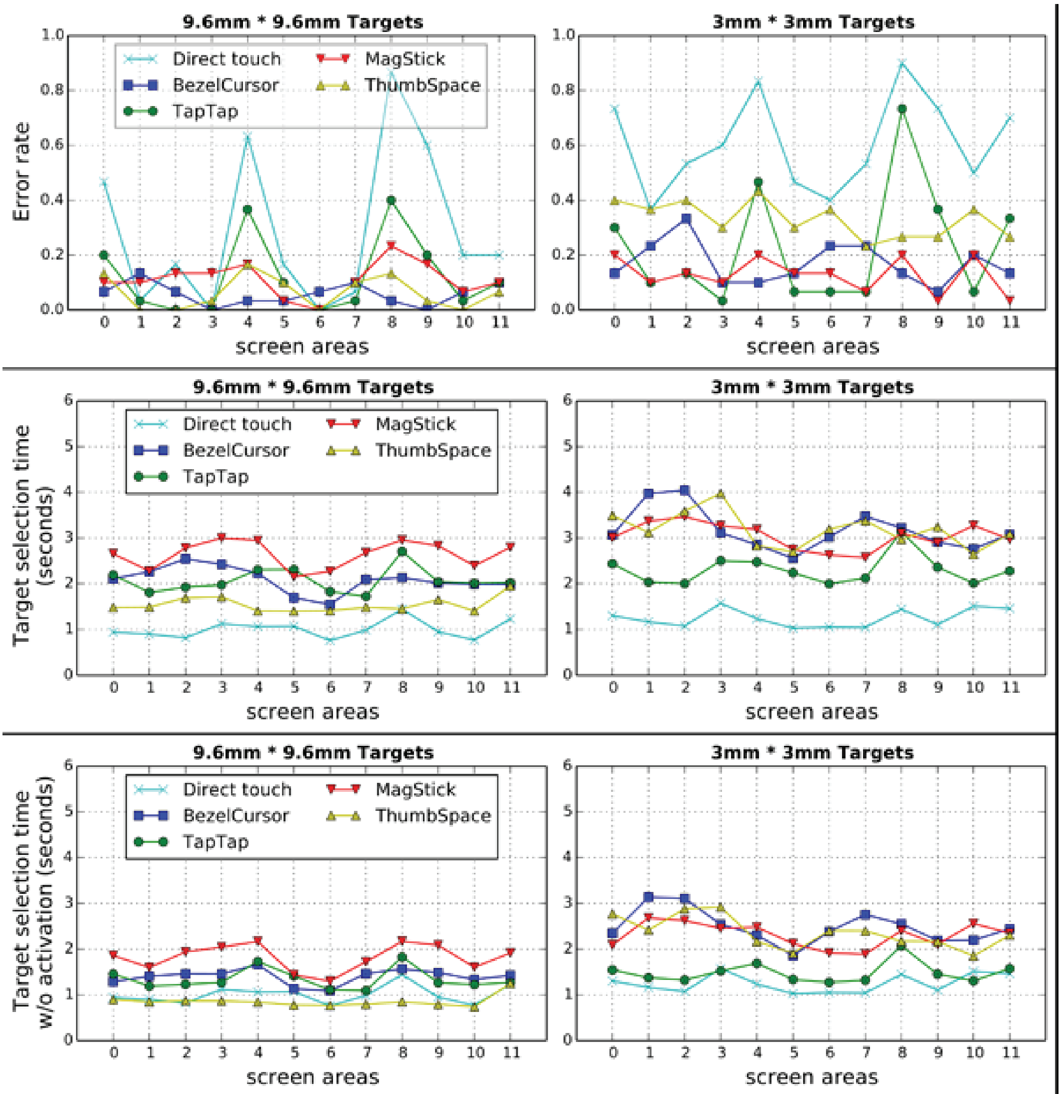
significantly lower error rate than direct touch and TapTap ($p < 0.001$ and $p < 0.04$). The mean of error rate of MagStick was higher than BezelCursor's, but it was not significant ($p > 0.14$). No significant difference in error rate was found between BezelCursor and ThumbSpace ($p = 1$). BezelCursor was significantly faster than MagStick ($p < 0.001$ for both t_1 and t_2). It was not significantly different from TapTap ($p = 1$ for both t_1 and t_2). It was significantly slower than direct touch and ThumbSpace ($p < 0.001$ for both t_1 and t_2). For small targets: BezelCursor had a significantly lower error rate than direct touch and ThumbSpace ($p < 0.001$ and $p < 0.04$). The mean of error rate of TapTap was higher than BezelCursor's, but it was not significat ($p > 0.4$). No significant difference in error rate was found between BezelCursor and MagStick ($p > 0.8$). BezelCursor was significantly slower than direct touch and TapTap ($p < 0.001$ for both t_1 and t_2). No significant difference in t_1 or t_2 was found between BezelCursor, MagStick and ThumbSpace.

Analysis on Individual Target Areas: Figure 14 shows the performance of each technique in terms of error rate and target selection time with respect to individual target areas. Matched with our hypothesis, in distant areas (0, 4, 8), BezelCursor had the lowest error rate among all the techniques. We performed pairwise comparisons using paired t-tests with Bonferroni P value adjustment for each of the target sizes. The result is summarized in Table 3.

For Large Targets: BezelCursor had significantly lower error rates than direct touch in area 0, 4, 8 and 9 (all $p < 0.006$). In all the other areas, the differences in error rate between BezelCursor and direct touch were not significant (all $p > 0.6$). BezelCursor was also significantly more accurate than TapTap in area 4 and 8 (both $p < 0.03$). In all the other areas, the differences in error rate between BezelCursor and TapTap were not significant (all $p > 0.6$ except for area 9 $p > 0.1$). The means of error rate of BezelCursor were lower than MagStick's in area 0, 2, 3, 4, 8 and 9, however, they were not significant ($p > 0.2$). They were also not significant in the remaining areas ($p > 0.6$). No significant difference in error rate was found between BezelCursor and ThumbSpace in all areas (all $p > 0.4$). As expected, the means of t_1 of BezelCursor were higher than direct touch in every area, and were significant ($p < 0.05$) except in areas 5 and 8 (both $p > 0.06$). This was also similar for t_2 , but only areas 2, 6, 7, 9 and 10 were significant ($p < 0.05$). BezelCursor was faster than MagStick in both t_1 and t_2 in every area, but was only significant ($p < 0.05$) in 4 (t_1), 5 (t_2), 6 (t_1) and 8 (t_1). No significant difference in t_1 and t_2 was found in all areas between BezelCursor and TapTap (all $p > 0.08$). BezelCursor was slower than ThumbSpace in all areas, and was significant in t_1 for areas 1, 4, 7, 8, and 10 ($p < 0.05$), in t_2 for all area ($p < 0.05$) except 11 ($p = 1$).

For Small Targets: BezelCursor had lower error rates than direct touch in every area, and was significant for areas 0, 3, 4, 8, 9, and 11 ($p < 0.05$). BezelCursor also had lower error rates than ThumbSpace in every area, except 7. However, none of the areas gave a significant result (all $p > 0.05$). BezelCursor had significantly lower error rates than TapTap for areas 4, 8, and 9 ($p < 0.05$). In remaining areas, the differences in error rate between BezelCursor and TapTap were not significant (all $p > 0.3$). For none of the areas, the difference in error rate between BezelCursor and MagStick showed significance (all $p > 0.2$). BezelCursor was significantly slower than direct touch for all area except 1 (t_2), 3 (t_2), 10 (t_1 , t_2) ($p > 0.09$). BezelCursor was also slower than TapTap in all areas, and was significant for areas 1 (t_2), 2 (t_1), 3 (t_2), 4 (t_2), 6 (t_1 , t_2), 7 (t_1 , t_2), 9 (t_2), and 10 (t_1 , t_2) ($p < 0.05$). No significant difference in t_1 and t_2 between BezelCursor and MagStick was found in any area (all $p > 0.4$). No significant different in t_1 and t_2 between BezelCursor and ThumbSpace was found in any area either (all $p > 0.6$).

Figure 14. The performance of different methods in individual target areas during user study 3



To conclude, BezelCursor gave significantly lower error rate than direct touch and TapTap, especially in distant screen areas. BezelCursor was significantly faster than MagStick, and significantly more accurate than ThumbSpace for small targets.

LIMITATIONS

BezelCursor is performed using a single touch-down, touch-move and touch-up operation. While being modeless, lightweight and transient our technique is perfect for selecting single targets, it is less efficient for tasks involving multiple operations (e.g., making multiple selections from a toolbar, selecting an item from a drop-down list). Dragging based interactions on targets are also not supported, but they can be achieved for example by removing the select-by-release functionality, allowing the cursor to stay on screen, and requiring an addition touch to operate on the underlying target, which

Table 3. Summary of the performance difference between BezelCursor and other methods in individual target areas during user study 3. Tick symbols (✓) are which BezelCursor were better (lower t_1, t_2, or error). Cross symbols (×) are which BezelCursor were worse. Double ticks or crosses are statistically significant differences ($p<0.05$). Dashes (-) are which the means are equal.

size	region	Direct touch			MagStick			TapTap			ThumbSpace		
		t_1	t_2	error	t_1	t_2	error	t_1	t_2	error	t_1	t_2	error
9.6mm * 9.6mm	0	✗✗	✗	✓✓	✓	✓	✓	✓	✓	✓	✗	✗✗	✓
9.6mm * 9.6mm	1	✗✗	✗	✗	✓	✓	✗	✗	✗	✗	✗✗	✗✗	✗
9.6mm * 9.6mm	2	✗✗	✗✗	✓	✓	✓	✓	✗	✗	✗	✗	✗✗	✗
9.6mm * 9.6mm	3	✗✗	✗	-	✓	✓	✓	✗	✗	-	✗	✗✗	✓
9.6mm * 9.6mm	4	✗✗	✗	✓✓	✓✓	✓	✓	✓	✓	✓✓	✗✗	✗✗	✓
9.6mm * 9.6mm	5	✗	✗	✓	✓	✓✓	-	✓	✓	✓	✗	✗✗	✓
9.6mm * 9.6mm	6	✗✗	✗✗	✗	✓✓	✓	✗	✓	✓	✗	✗	✗✗	✗
9.6mm * 9.6mm	7	✗✗	✗✗	✗	✓	✓	-	✗	✗	✗	✗✗	✗✗	-
9.6mm * 9.6mm	8	✗	✗	✓✓	✓✓	✓	✓	✓	✓	✓✓	✗✗	✗✗	✓
9.6mm * 9.6mm	9	✗✗	✗✗	✓✓	✓	✓	✓	✓	✗	✓	✗	✗✗	✓
9.6mm * 9.6mm	10	✗✗	✗✗	✓	✓	✓	-	✓	✗	✗	✗✗	✗✗	✗
9.6mm * 9.6mm	11	✗✗	✗	✓	✓	✓	-	✓	✗	-	✗	✗	✗
3mm * 3mm	0	✗✗	✗✗	✓✓	✗	✗	✓	✗	✗✗	✓	✓	✓	✓
3mm * 3mm	1	✗✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✓
3mm * 3mm	2	✗✗	✗✗	✓	✗	✗	✗	✗✗	✗	✗	✗	✗	✓
3mm * 3mm	3	✗✗	✗	✓✓	✓	✗	-	✗	✗✗	✗	✓	✓	✓
3mm * 3mm	4	✗✗	✗✗	✓✓	✓	✓	✓	✗	✗✗	✓✓	✗	✗	✓
3mm * 3mm	5	✗✗	✗✗	✓	✓	✓	-	✗	✗	✗	✓	✓	✓
3mm * 3mm	6	✗✗	✗✗	✓	✗	✗	✗	✗✗	✗✗	✗	✓	✓	✓
3mm * 3mm	7	✗✗	✗✗	✓	✗	✗	✗	✗✗	✗✗	✗	✗	✗	-
3mm * 3mm	8	✗✗	✗✗	✓✓	✗	✗	✓	✗	✗	✓✓	✗	✗	✓
3mm * 3mm	9	✗✗	✗✗	✓✓	✗	✗	✗	✗	✗✗	✓✓	✓	✗	✓
3mm * 3mm	10	✗	✗	✓	✓	✓	-	✗✗	✗✗	✗	✗	✗	✓
3mm * 3mm	11	✗✗	✗✗	✓✓	✗	✗	✗	✗	✗	✓	✓	✗	✓

would be similar to a cursor controlled by a touch pad. Second, it is not theoretically guaranteed that the functions for mapping thumb movement to cursor movement can make all screen regions easily accessible, since BezelCursor is a single fluid action and does not allow to lift up the thumb for further adjustment, as we often do with touchpad control. However, both linear and accelerated mapping functions contain a few parameters that can be easily customized by users for different screen sizes. Thirdly, device of smaller screen may be less benefited from using BezelCursor. However, since BezelCursor is designed to help to acquire out-of-reach targets, there is little reason in using it on a device where the whole screen is easily accessible. Lastly, BezelCursor cannot be directly used with existing bezel-initiated gestures. For example, the top bezel has been occupied for the gesture bringing notification center in Android or iOS. The left and right bezel edges have not been commonly used but been gaining more attention very recently. In the recently introduced operation system of Windows 8, left bezel has been reserved for switching applications and right bezel for activating Windows 8 Charms. Observing that these bezel-initiated operations are all achieved by simple bezel swipe without any target constraint and thus are consistently faster to perform than BezelCursor, we expect that it is possible to integrate them with BezelCursor by examining the difference in operation time. Alternatively we can let users customize the parts of the bezel for BezelCursor activation and leave the remaining parts for existing bezel-initiated operations.

CONCLUSION AND FUTURE WORK

We presented BezelCursor, a one-handed target acquisition technique that is fast, simple to implement, easy to learn, compatible with commonly used interaction styles, scalable to screens of large size, and applicable to mobile environments. It is shown by our study that with secure gripping, BezelCursor gave significantly lower error rate than direct touch and TapTap, and BezelCursor was significantly faster than MagStick and significantly more accurate than ThumbSpace for small targets. Since BezelCursor is tailored for selecting out-of-reach targets, it would be interesting to quantitatively evaluate the relative benefits of using BezelCursor together with the techniques designed for selecting targets with reach (e.g., Shift) in the future. In addition, we plan to explore the possibility of applying BezelCursor to multiple target selection, dragging based target interactions, and more importantly, real world interfaces.

REFERENCES

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 INTERACT (Vol. 3, pp. 57–64). Retrieved from http://research.microsoft.com/apps/pubs/default.aspx?id=101827

BezelCursor. (2014). Retrieved from https://www.youtube.com/watch?v=7SrxJeJRi2M

Bezerianos, A., & Balakrishnan, R. (2005). The vacuum: facilitating the manipulation of distant objects. In CHI '05 (pp. 361–370). Portland, Oregon, USA. doi:10.1145/1054972.1055023

Boring, S., Ledo, D., & Chen, X. "Anthony," Marquardt, N., Tang, A., & Greenberg, S. (2012). The Fat Thumb: Using the Thumb's Contact Size for Single-handed Mobile Interaction. In Proceedings of the 14th International Conference on Human-computer Interaction with Mobile Devices and Services (pp. 39–48). San Francisco, California, USA: ACM. doi:10.1145/2371574.2371582

Bragdon, A., Nelson, E., Li, Y., & Hinckley, K. (2011). Experimental analysis of touch-screen gesture designs in mobile environments. In CHI '11 (pp. 403–412). Vancouver, BC, Canada. doi:10.1145/1978942.1979000

Chapuis, O., Labrune, J.-B., & Pietriga, E. (2009). DynaSpot: speed-dependent area cursor. In CHI '09 (pp. 1391–1400). Boston, MA, USA. doi:10.1145/1518701.1518911

FINGAS. J. (2012, September 3). Shocker: smartphone users like bigger screens, market share may respond accordingly. Retrieved July 24, 2014, from http://www.engadget.com/2012/09/03/shocker-smartphone-users-like-bigger-screens/

Forlines, C., Vogel, D., & Balakrishnan, R. (2006). HybridPointing: fluid switching between absolute and relative pointing with a direct input device. In UIST '06 (pp. 211–220). Montreux, Switzerland. doi:10.1145/1166253.1166286

Grossman, T., & Balakrishnan, R. (2005). The bubble cursor: enhancing target acquisition by dynamic resizing of the cursor's activation area. In CHI '05 (pp. 281–290). Portland, Oregon, USA. doi:10.1145/1054972.1055012

Hirotaka, N. (2003). Reassessing current cell phone designs: using thumb input effectively. In CHI EA '03 (pp. 938–939). Ft. Lauderdale, Florida, USA. doi:10.1145/765891.766081

Jain, M., & Balakrishnan, R. (2012). User Learning and Performance with Bezel Menus. In CHI '12 (pp. 2221–2230). Austin, Texas, USA. doi:10.1145/2207676.2208376

Karlson, A., & Bederson, B. (2007). ThumbSpace: Generalized one-handed input for touchscreen-based mobile devices. *Human-Computer Interaction–INTERACT*, *2007*, 324–338.

Karlson, A. K., & Bederson, B. B. (2008). One-handed touchscreen input for legacy applications. In CHI '08 (pp. 1399–1408). Florence, Italy. doi:10.1145/1357054.1357274

Karlson, A. K., Bederson, B. B., & Contreras-Vidal, J. (2007). Understanding one handed use of mobile devices. Handbook of Research on User Interface Design and Evaluation for Mobile Technology.

Karlson, A. K., Bederson, B. B., & Contreras-Vidal, J. L. (2006). Studies in One-Handed Mobile Design: Habit, Desire and Agility. In *Proceedings of the 4th ERCIM Workshop on User Interfaces for All*.

Kim, S., Yu, J., & Lee, G. (2012). Interaction Techniques for Unreachable Objects on the Touchscreen. In OzCHI '12 (pp. 295–298). Melbourne, Australia. doi:10.1145/2414536.2414585

Li, W. H. A., & Fu, H. (2013). BezelCursor: Bezel-initiated cursor for one-handed target acquisition on mobile touch screens. In SIGGRAPH Asia 2013 Symposium on Mobile Graphics and Interactive Applications.

Parhi, P., Karlson, A. K., & Bederson, B. B. (2006). Target Size Study for One-handed Thumb Use on Small Touchscreen Devices. In *Proceedings of the 8th Conference on Human-computer Interaction with Mobile Devices and Services* (pp. 203–210). New York, NY, USA: ACM. doi:10.1145/1152215.1152260

Potter, R. L., Weldon, L. J., & Shneiderman, B. (1988). Improving the accuracy of touch screens: an experimental evaluation of three strategies. In CHI '88 (pp. 27–32). Washington, D.C., United States. doi:10.1145/57167.57171

Ren, X., & Moriya, S. (2000). Improving selection performance on pen-based systems: A study of pen-based interaction for selection tasks. *ACM Transactions on Computer-Human Interaction*, *7*(3), 384–416. doi:10.1145/355324.355328

Roth, V., & Turner, T. (2009). Bezel Swipe: Conflict-free Scrolling and Multiple Selection on Mobile Touch Screen Devices. In CHI '09 (pp. 1523–1526). Boston, MA, USA. doi:10.1145/1518701.1518933

Roudaut, A., Huot, S., & Lecolinet, E. (2008). TapTap and MagStick: improving one-handed target acquisition on small touch-screens. In AVI '08 (pp. 146–153). Napoli, Italy. doi:10.1145/1385569.1385594

Vogel, D., & Balakrishnan, R. (2005). Distant freehand pointing and clicking on very large, high resolution displays. In UIST '05 (pp. 33–42). Seattle, WA, USA. doi:10.1145/1095034.1095041

Vogel, D., & Baudisch, P. (2007). Shift: a technique for operating pen-based interfaces using touch. In CHI '07 (pp. 657–666). San Jose, California, USA. doi:10.1145/1240624.1240727

Yatani, K., Partridge, K., Bern, M., & Newman, M. W. (2008). Escape: a target selection technique using visually-cued gestures. In CHI '08 (pp. 285–294). Florence, Italy. doi:10.1145/1357054.1357104

Wing Ho Andy Li is currently a PhD candidate in the School of Creative Media, City University of Hong Kong. He obtained Bachelor of Science in Creative Media at CityU in 2010. His research focus is interactive installation and natural user interface. His research project Augmented Mirror, which is an augmented reality interactive mirror system, was exhibited in the top conference of the field, SIGGRAPH 2012, in Los Angeles. It was also exhibited in InnoCarnival 2012 at the HK Science Park, and was reported in several major local newspapers. He is experienced in installation art. The "Mirage" installation series (2008-2012) created by Li and his teammates, was one of his most famous work. "City of Swing" (2011) from "Mirage" was conceived by the Macao Science Center, developed and has been showing as a permanent exhibit. "Miream" (2009), also from the series, was invited to be shown in K11 shopping mall during the summer of 2013. Additionally, Li has been actively participating in open source communities globally. He is owner of several open source projects, and often contributes to other open source libraries.

Hongbo Fu is an Associate Professor in the School of Creative Media, City University of Hong Kong. Before joining CityU, he had postdoctoral research trainings at the Imager Lab, University of British Columbia, Canada and the Department of Computer Graphics, Max-Planck-Institut Informatik, Germany. He received the PhD degree in computer science from the Hong Kong University of Science and Technology in 2007 and the BS degree in information sciences from Peking University, China, in 2002. His primary research interests fall in the fields of computer graphics and human computer interaction. He has served as an associate editor of The Visual Computer*, and* Computers & Graphics*.*

Kening Zhu is an Assistant Professor in School of Creative Media, City University of Hong Kong. He received his PhD degree from National University of Singapore. Kening's research interests cover various topics on human-computer interaction (HCI), including interaction design, paper computing, and rapid prototyping. His research focuses on theoretical analysis and technical toolkit for technology-enhanced movable paper craft.

Call for Articles

International Journal of Mobile Human Computer Interaction

Volume 8 • Issue 1 • January-March 2016 • ISSN: 1942-390X • eISSN: 1942-3918

An official publication of the Information Resources Management Association

MISSION

The primary objective of the **International Journal of Mobile Human Computer Interaction (JMHCI)** is to provide comprehensive coverage and understanding of the issues associated with the design, evaluation, and use of mobile technologies. This journal focuses on human-computer interaction related to the innovation and research in the design, evaluation, and use of innovative handheld, mobile, and wearable technologies in order to broaden the overall body of knowledge regarding such issues. IJMHCI also considers issues associated with the social and/or organizational impacts of such technologies.

COVERAGE AND MAJOR TOPICS

The topics of interest in this journal include, but are not limited to:

Case studies and/or reflections on experience on experience (e.g. descriptions of successful mobile user interfaces, evaluation set-ups, etc.) • Context-aware/context-sensitive mobile application design, evaluation, and use • Design methods/approaches for mobile user interfaces • Ethical implications of mobile evaluations • Field-based evaluations and evaluation techniques • Gestural interaction techniques for mobile technologies • Graphical interaction techniques for mobile technologies • Issues of heterogeneity of mobile device interfaces/interaction • Lab v. field evaluations and evaluation techniques • Lab-based evaluations and evaluation techniques • Mobile advanced training application design, evaluation, and use • Mobile assistive technologies design, evaluation, and use • Mobile commerce application design, evaluation, and use • Mobile HCI lab design/set-up • Mobile healthcare application design, evaluation, and use • Mobile interactive play design, evaluation, and use • Mobile learning application design, evaluation, and use • Mobile technology design, evaluation, and use by special (needs) groups (e.g. elderly, children, and disabled) • Multimodal interaction on mobile technologies • Non-speech audio-based interaction techniques for mobile technologies • Other emerging interaction techniques for mobile technologies • Other related issues that impact the design, evaluation, and use of mobile technologies • Speech-based interaction techniques for mobile technologies • Tactile interaction techniques for mobile technologies • Technology acceptance as it relates to mobile technologies • User aspects of mobile privacy, security, and trust • User interface architectures for mobile technologies • User interface migration from desktop to mobile technologies • Wearable technology/application and interaction design, evaluation, and use

ALL INQUIRIES REGARDING IJMHCI SHOULD BE DIRECTED TO THE ATTENTION OF:

Joanna Lumsden, Editor-in-Chief • IJMHCI@igi-global.com

ALL MANUSCRIPT SUBMISSIONS TO IJMHCI SHOULD BE SENT THROUGH THE ONLINE SUBMISSION SYSTEM:

http://www.igi-global.com/authorseditors/titlesubmission/newproject.aspx

IDEAS FOR SPECIAL THEME ISSUES MAY BE SUBMITTED TO THE EDITOR(S)-IN-CHIEF

PLEASE RECOMMEND THIS PUBLICATION TO YOUR LIBRARIAN

For a convenient easy-to-use library recommendation form, please visit:
http://www.igi-global.com/IJMHCI