Rapid Selection of Hard-to-Access Targets by Thumb on Mobile Touch-Screens

Neng-Hao Yu [1] **Da-Yuan Huang** [2] **Jia-Jyun Hsu** [1] **Yi-Ping Hung** [2]

[1] Department of Computer Science,
Master Program of Digital Contents Technologies
National Chengchi University, Taipei, Taiwan

[2] Department of Computer Science and Information Engineering
National Taiwan University, Taipei, Taiwan

jonesyu@nccu.edu.tw

ABSTRACT
Current touch-based UIs commonly employ regions near the corners and/or edges of the display to accommodate essential functions. As the screen size of mobile phones is ever increasing, such regions become relatively distant from the thumb and hard to reach for single-handed use. In this paper, we present two techniques: CornerSpace and BezelSpace, designed to accommodate quick access to screen targets outside the thumb's normal interactive range. Our techniques automatically determine the thumb's physical comfort zone and only require minimal thumb movement to reach distant targets on the edge of the screen. A controlled experiment shows that BezelSpace is significantly faster and more accurate. Moreover, both techniques are application-independent, and instantly accommodate either hand, left or right.

Author Keywords
Mobile devices; one-handed interaction; thumb-based interaction; touch-screens; interaction techniques.

ACM Classification Keywords
H.5.2. User Interfaces: Input devices and strategies, Interaction styles, Screen design.

General Terms
Human Factors; Design; Measurement.

INTRODUCTION
Mobile phones are commonly utilized with a single hand only. Karlson et al.[3] shows that users prefer to use smart phones with only one hand in the majority of the time. As more mobile devices with larger screens enter the market and are employed by users, users may encounter difficulties in reaching distant target when using only their thumbs during single-handed use. We address this issue as the "*thumb's reach problem*". According to current mobile UI patterns, regions of screens' corners and edges are usually used to accommodate essential functions (Figure 1a). For example, Apple's human interface guidelines suggest that "Back" and "Action" buttons should be placed at the top-left and top-right corners; while the "Tab" bar should be placed at the bottom of the screen. Such configurations make solving the thumb reach problem more urgent. Alternative target acquisition techniques on mobile devices have been proposed. ThumbSpace[4] requires one-time setup for a proxy view that uses a sub-region of the display to map all available screen targets. MagStick[6] provides a telescopic stick to control a "magnetized" cursor to indicate an on-screen target by dragging one's finger in the direction opposite from the target vis-a-vis the initial point of display screen surface contact. These techniques only have been tested on smaller devices (from 2.8" ~ 3.5" screen). Would they perform well on larger devices and for modern UIs? These questions are examined in the following pilot study.

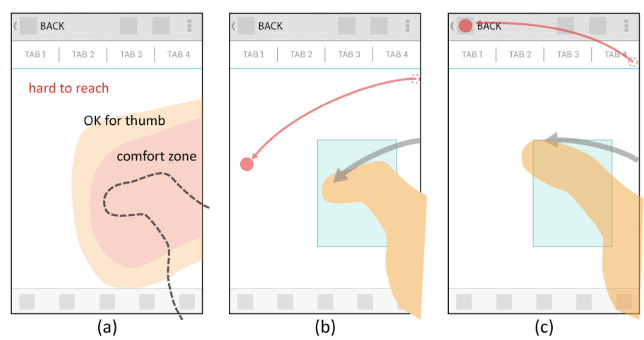

Figure 1: (a) essential functions are usually located in the thumb's hard-to-reach area, (b)(c) the design of BezelSpace: Bezel-Swipe casting a cursor (like an extended fingertip) on the screen while the proxy region adaptively moves under the thumb's location. Lift up the thumb to select the target.

Pilot study
We replicate aforementioned techniques on current mobile devices with a larger screen size -- Google Galaxy Nexus (13.55 x 6.79 x 0.89 cm, 4.65" display) and Samsung Galaxy Note (14.685 x 8.295 x 0.965 cm, 5.3" display). 12 participants were recruited to try freely each technique on a simulated web app. First, we discovered that participants overwhelmingly prefer direct-touch interactions for accessing on-screen targets. Participants only consider enabling the tested techniques when the targets are really hard to reach (e.g., screen corners and edges opposite of the

thumb when gripping single-handedly a device). Second, accessing more distant on-screen targets requires that participants alter how they grip their devices. Participants may not be aware that their grip has slightly changed and reported that the proxy view of ThumbSpace was not perfect match the thumb's normal interactive range. Third, the participants with small hands felt more fatigue when using MagStick. Since MagStick uses an initial contact point as a fulcrum from which the user extends a telescopic pointer from that fulcrum to the distant target by dragging one's finger in the direction opposite from the target vis-a-vis the fulcrum, the further away the target is the farther thumb must move. Some participants even reported they were not able to reach far targets at the corners of the screen. Beyond the main question of inquiry, our study results indicate that participants like using the "magnetized" cursor (semantic pointing [2]) in MagStick more than the highlighted items (object pointing) in ThumbSpace because they can easily predict the accuracy of their thumb movements via the cursor. Finally, participants felt ThumbSpace is more intuitive because it was like an absolute trackpad. These factors motivated us to design alternative techniques to solve the thumb reach problem.

DESIGN

From our findings in this pilot study, we offer several considerations of design as follows: (1) Users encounter the thumb's reach problem regarding out-of-reach screen areas, especially at corners and edges, at which are located frequent-use targets, (2) Users need a quick mode switch between direct touch and assistive technique so that does not hinder task performance, (3) thumb's interactive region should be adaptively moved to the thumb's location in case of the grip has changed.

Adaptive comfort zone and moding technique

In order to adaptively find individual users comfortable range of motion for the thumb as the interaction region, we use the bezel-swipe gesture. Bezel-Swipe[5] takes advantage of the edge of a touch display, enabling users' thumbs to easily access functionality by activating a thin button. As Bezel Swipe is triggered, we can predict that the comfort zone of the thumb is near the start position and the lift-up position. Li et al.[1] also conclude that bezel gestures have the advantages of being a lesser attentional load over soft buttons, eyes-free, and distinguishable from scrolling or other on-screen gestures. Thus it can be used as a seamless mode switch between direct-touch interaction and target acquisition technique.

CornerSpace

Considering Mobile UI patterns, many essential functions are near the corners and the edges of a screen. We assume that corners are the most frequently used area. We propose that CornerSpace works as follows: (1) after the bezel swipe, the UI buttons are shown at the lift-up position (Figure 2a). Each button represents its own corresponding corner except the red button is for canceling the mode. To minimize the aiming effort, the screen is split into 4 parts (Figure 2b). Tapping the button or anywhere on those quarters will trigger the corresponding corner target. (2) As for edges and other on-screen targets, users can tap the nearest corner and drag out a "magnetized" cursor to indicate the target (Figure 2c). The CD-gain was set to a fix ratio about 2:1. (3) The target is selected when a user's thumb lifts from the screen. We use MagStick's "magnetized" mechanism to improve stability while dragging and lifting. Based on these design strategies, users only require minimal movement and can quickly access corner targets.

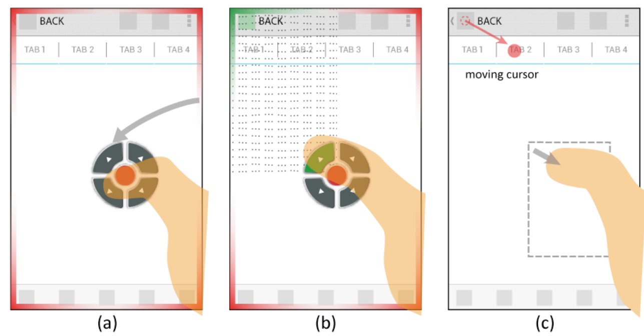

Figure 2: The design of CornerSpace, (a) CornerSpace UI appears at the thumb's final contact location of a Bezel-Swipe (b) Quick access of the corner target (top-left): tapping on the arrow button or anywhere inside the dotted region will trigger the corner target. (c) Accessing a target near the corner: trigger the nearest corner in previous action for a reference point and drag out a "magnetized" cursor to access the target.

BezelSpace

Compared to CornerSpace's two-step operation, BezelSpace (Figure 1bc) combines moding + targeting in single step for continuous operation: (1) cursor appears when the bezel swipe occurs without lifting the thumb. (2) A user must continue to drag ones finger across the screen to control the mapped "magnetized" cursor and aim it towards the target. (3) The target is selected when a user's thumb lifts from the screen. By this design users can directly stretch the thumb toward the target. The mapping of BezelSpace is the same as ThumbSpace but the proxy region adaptively shifts according to any bezel swipe's initial location on the screen. In our preliminary study, we found users use different way to do bezel swipe for different location of target. As for the targets above the thumb, users tend to bend the thumb triggering the first contact point on the bezel and then push the thumb toward the upper direction. Meanwhile, for the targets below the thumb, users tend to stretch out the thumb triggering the first contact point and swipe to the lower direction. We utilize this characteristic to set the position of proxy region a bit higher than bezel swipe's initial location.

BezelSpace works like an extended fingertip. Users use a bezel swipe to produce a pseudo-fingertip (cursor), which can be stretched to any distant target. The fingertip motion continues to the target, without requiring the user to lift and reposition the thumb. In addition, because the thumb is continuously in contact with the display, it can provide tactile feedback to the user, enabling them to fine tune their selections before committing to them. Moreover, the physical limitation of the thumb could be used as an advantage. According to Fitt's law, any corner and border has infinite width. For corner and edge on-screen targets, users just stretch the thumb all the way to the corresponding direction of the target. Then the target will be selected. If users want to cancel the mode, they can drag the cursor inside the proxy region. Because the region is actually the ease-of-reach area, which is not required assistive technique.

EVALUATION

We conduct a user study to validate CornerSpace and BezelSpace in terms of interaction effectiveness, usability, and user satisfaction. To offer a performance baseline, we utilize ThumbSpace as a comparative model. However we did not utilize MagStick because it cannot be used on relatively larger screens. The original ThumbSpace uses hard buttons to activate the technique; we change this to a bezel-swipe gesture for consistency with CornerSpace and BezelSpace. Based on our design strategies, we previously hypothesized that CornerSpace and BezelSpace would outperform ThumbSpace in terms of selection time and error rate for the corner- or edge-located targets while maintaining prior levels of performance for other targets.

Task

Participants were presented with a series of individual target selection trials. Based on our pilot study, users prefer to select targets with direct touch until finding that a target is out of range. The technique is then activated when users encountered the *thumb reach problem*, so that only the out-of-reach targets were included in the trial. In the beginning of the test, participants must perform a one-time calibration for setting a proxy view of ThumbSpace. The area outside of that proxy view is regard as the out-of-reach area. We divided the screen into a 5x9 grid and distribute the targets from the out-of-reach area onto the grid. We further defined three types of on-screen targets: (1) *Corner* targets, which are located at the four corners. (2) *Edge* targets, which are located on the edges. (3) *Other* targets, which are found in the remaining out-of-reach area. Each type of target was randomly assigned and evenly distributed within each block. That is 4 targets of each type and a total of 12 targets for each block. Only one target was painted red for each trial; others were painted blue as a distraction in order to improve realism of the target selection task. When the target was focused or selected, the color changed to green. The participants were instructed to select the red targets as quickly and accurately as possible. We use 7mm^2 as the target size and 2 mm as the gap since this value is reported to be the actual minimal size/gap in current mobile UI design[2].

Apparatus and participants

These techniques are implemented in Java on the Android Platform, and the experiments are performed on the Samsung Galaxy Note2 (80.5 x 151.1 x 9.4 mm, 5.5" display). Fifteen volunteers, ranging in age from 18 to 33 years of age (M=26, SD=3.9, 6 female, all right-handed and owners of touch-screen phones), were recruited on campus, and each participant received NT$100 (approx. US$3.50) for a half hour-long test. Participants' ease-of-reach region covers 35%~55% area of the 5.5" screen.

Experimental design

We use a two-way *repeated measures within-subjects* design. The independent variables are *Method* (ThumbSpace, CornerSpace, and BezelSpace) and *Type* (Corner, Edge, and Other). Presentation of Method is counter-balanced across participants. For each Method, we allow one practice and five timed blocks for the experiment. After completing one Method, participants are then asked to fill out an assessment questionnaire. After the study, they are asked to rank the three Methods.

In summary, the experimental design is:

15 participants

x 3 *Methods* (ThumbSpace, CornerSpace, and BezelSpace)

x (1+5) *Blocks* (training + measured)

x 12 trials (with 3 *types* of target: *Corner*, *Edge*, and *Other*)

= 3,240 trials completed

RESULTS AND DISCUSSION

We compare selection time and error rate with a separate *repeated measures within-subjects* analyses of variance (ANOVA). For pair-wise post hoc tests, we use Bonferroni-corrected confidence intervals to compare against α=0.05. In cases where the assumption of sphericity is violated, we correct the degrees of freedom using Greenhouse-Geisser. For ease of reference, we use "TS, CS, and BS" to represent ThumbSpace, CornerSpace, and BezelSpace respectively in the following paragraphs.

Selection time

We measured task time from the moment a bezel swipe gesture occurs until a participant's thumb is lifted up from the screen. Trials with selection errors were excluded from analysis. We perform a 3x3 (*Method* x *Type*) RM-ANOVA and find significant main effects for *Method* ($F_{2,28} = 26.16$, $p < .001$), *Type* ($F_{2,28} = 33.52$, $p < .001$), and a *Method* x *Type* interaction ($F_{2.32,32.50} = 14.90$ $p < .001$). Post hoc multiple means comparison tests show that BS differs significantly from CS and TS for *Edge* and *Other* targets (all $p < .001$). Figure 3b shows that BS requires slightly less time for *Other* than *Corner*/*Edge* targets while CS and TS requires less time for *Corner/Edge* than *Other* targets. This is consistent with our hypotheses: (1) BS relies on continuous operation from bezel to target, since *Other* is closer to thumb than *Corner*/*Edge* targets thus participants utilize minimal thumb movement for *Other* targets. (2) When using CS and TS, the participants tried to use a corner as a reference to aim towards nearby targets. To our surprise, CS did not perform best for *Corner* targets and did not render better results than TS. We will explain more in regard to these findings in later discussion. Overall, BS is significantly faster (M= 1456.6ms, SD = 412.7 ms) than CS (M = 2213.7 ms, SD = 415.16 ms) and TS (M = 2222.44 ms, SD = 657.69 ms). (Figure 3a)

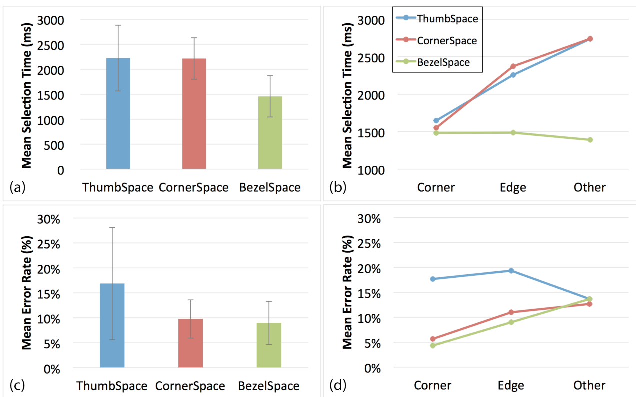

Figure 3: (a) overall selection time, (b) selection time per target type, (c) overall error rate, (d) error rate per target type

Error rate

The error rate measurement aggregates both empty and wrong target selections. We perform a 3x3 (*Method* x *Type*) RM-ANOVA and find significant main effects for *Method* ($F_{1.26,\ 17.65}$ = 7.00, $p = 0.012$), and a *Method* x *Type* interaction ($F_{4,56}$ = 3.28, $p = 0.017$). Figure 3c shows mean error rate among *Method* and *Type*. Post hoc multiple means comparison tests show that BS and CS differ significantly from TS for *Corner* targets. As for *Edge* target, BS also shows significant difference from TS (all $p < 0.05$). This may be explained by the difference between Object pointing and Semantic pointing. Because the participant may also involuntarily move the stretched thumb when releasing it to the corner or edge, Semantic Pointing prevents the cursor from leaving the target when the thumb is slightly, and involuntarily, moved.

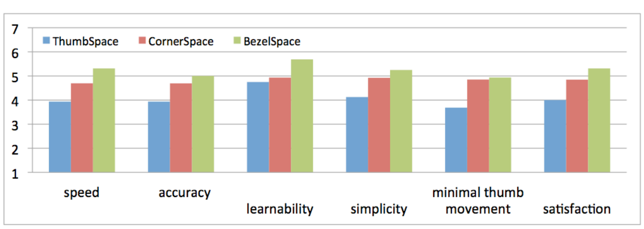

Figure 4: Questionnaire result (means)

Subjective preferences

After each *Method*, participants complete a questionnaire to assess each particular input technique. Both BS and CS scored consistently well across all categories on a 7-point Likert scale (Figure 4). We believe the Semantic pointing with cursor feedback does contribute to learnability and simplicity. Participants also report that they felt less thumb movement when using BS through the continuous operation. As for ranking results, BS and CS rank mostly as 1st and 2nd, though we did not find significant difference via ranking.

Discussion

The results of our experimentation show that BezelSpace is the most efficient and accurate method (Figure 3). We believe that the single-step continuous operation utilized shortens the task time. We did observe that many participants tend to stretch thumb directly toward the target when they hit the corner and edge targets. Surprisingly, CornerSpace only competes with ThumbSpace in its lower error rate. In the testing of CornerSpace, we observed that several participants tend to start from the top-left corner for every target because they did not bother to judge where the nearest corner was. This produced a penalty for longer movement if he starts from the opposite corner. In addition, the participants tend to aim for the UI button of CornerSpace even when the whole area of the screen is active. We will investigate other visual designs to address this issue.

CONCLUSION AND FUTURE WORK

In this paper, we address the *thumb reach problem* and propose design considerations that could be used in the development of interaction. We present two techniques to assist users in gaining easy access by thumb to distant on-screen targets within the thumb's normal interactive range while gripping a device in single hand. Our user study of target selection reveals that BezelSpace is fastest and most accurate when compared with previously created techniques. Moreover, these two methods are suitable for use by either hand without extra setup. They conform to arbitrary UI elements while also serving as an application-independent technique. We plan to extend these techniques to even larger touch-screens on tablets.

ACKNOWLEDGEMENTS

This study was partially supported by the National Science Council, Taiwan, under grant NSC101-2211-E-004-013, NSC101-2219-E-002-026 and NSC102-2221-E-004-004.

REFERENCES

1. Bragdon, A., Nelson, E., Li, Y., and Hinckley, K. Experimental analysis of touch-screen gesture designs in mobile environments. *CHI'11*, (2011).
2. Blanch, R., Guiard, Y., and Beaudouin-Lafon, M., Semantic Pointing: Improving Target Acquisition with Control-Display Ratio Adaptation. CHI'04, (2004)
3. Karlson, A.K., Bederson, B.B., and Contreras-vidal, J.L. Studies in One-Handed Mobile Design : Habit, Desire and Agility. *Tech report HCIL-2006-02*, (2006).
4. Karlson, A.K. and Bederson, B.B. ThumbSpace : Generalized One-Handed Input for Touchscreen-Based Mobile Devices. *INTERACT 2007*
5. Roth, V. Bezel Swipe : Conflict-Free Scrolling and Multiple Selection on Mobile Touch Screen Devices. (2009).
6. Roudaut, A., Huot, S., and Lecolinet, E. TapTap and MagStick : Improving One-Handed Target Acquisition on Small Touch-screens. AVI'08, (2008).