

Virtual Touchpad for Cursor Control of Touchscreen Thumb Operation in the Mobile Context

Yu Ren Lai and T.K. Philip Hwang^(✉)

National Taipei University of Technology, P.O. Box 2813 no.1, Sec. 3, E. Rd.,
Taipei 10608, Taiwan, R.O.C.
phwang@ntut.edu.tw

Abstract. This study formulated existing virtual pointing techniques for cursor control in the mobile context of touchscreen thumb operation. Three virtual pointing models were developed, including: **Virtual Touchpad**, **Virtual Joystick** and **Virtual Direction Key**. In order to verify their usability and feasibility, a user study was employed to evaluate the usability of three virtual pointing models, followed by the focus group interview to experienced usability designers, in which, constraints of touchscreen cursor control in mobile context were defined and rated against three virtual pointing models. Research findings: (1) Virtual Touchpad was significantly efficient than others, while Virtual Direction Key presented lower error rate, although insignificantly. (2) Constraints of touchscreen cursor control in mobile context include: stable and simple operation, Interruptible operation is better and avoids accurate pointing. Virtual Direction Key stood out as the most stable, simple and interruptible pointing control.

Keywords: Virtual touchpad · Cursor control · Thumb operation

1 Introduction

Before the popularity of smartphones, most people commonly used mobile phones with only one hand [1]. Interfaces that allow one-handed operation interaction can provide a significant advantage in which users can use the other hand for mobile tasks [2]. With the smart-phones enter people's lives. There are more and more applications let us to use that also mean we will get a lot of information on smartphone. In this way people need the bigger screen to display more information on screen. Therefore, more and more smartphone manufacturers have been launched large-screen smartphones. Apple also launch two new iPhones, the 4.7-inch iPhone 6 and the 5.5-inch iPhone 6 Plus. It seem like large-screen is the trend of smartphone in the future. However, there exist a comfort zone on the smartphone screen in thumb operation scenarios [3]. Hooper [4] shows that users prefer to use large-screen smartphones with only one hand in the majority of the time (Fig. 1). Unfortunately, according to the design guidelines of the smartphone. Apple and Android are all suggested that navigation and functionality buttons should be placed on the top of screen. The top bar in the screen facilitates the

discovery of functions, where the user's hands would never cover them. Since smartphone screens are getting larger, it is much difficult for thumbs to reach the upper area of the screen especially in one-handed operation scenarios. In order to build interfaces that explicitly accommodate thumb interaction by ensuring that all targets are thumb sized and within thumb reach, this paper proposes three virtual pointing techniques (touchpad, joystick and direction key) interfaces located in comfort zone of thumb operation. All models of Virtual Touchpad aimed to control the cursor for reaching every part of the screen.

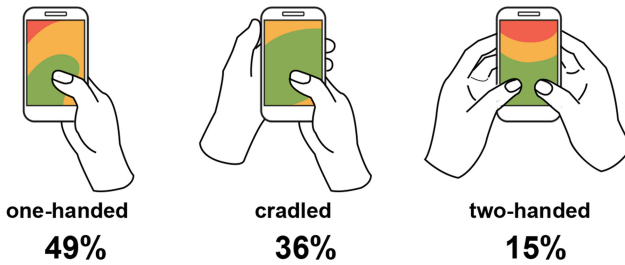


Fig. 1. User held their phones in three ways (source adapted from Hooper, 2013)

2 Related Work

2.1 Pointing Technique

Pointing to targets is a fundamental task in graphical user interfaces (GUI's) [5]. According whether the input and display space unified, the pointing technique can classify as direct interaction and indirect interaction [6].

Direct Pointing Technique. A direct pointing device has a unified input and display surface such as touch-screens, or display tablets operated with a pen. Direct touch has a lot of advantage: (1) it's a form of direct manipulation that is easy to learn. (2) It's a fastest pointing way. (2) It has easier hand-eye coordination than mouse or keyboards. (3) It doesn't need additional desk space [7]. But is also exists some dis-advantages like high error rate, lower accuracy and occluded problem by finger [8, 9].

Indirect Pointing Technique. An indirect pointing device isn't providing input in the same physical space as the output. For example, when users use a mouse that they must move the mouse on one surface (like a desk) to indicate a point on the screen. Typically they require more explicit feedback and representation of the pointing device (such as a cursor), the intended target on the screen (such as highlighting icons when the cursor hovers over them), and the current state of the device (such as whether a button is held or not). However, indirect pointing has an obviously advantage that indirect input is a better way to point large or far interaction surfaces, since it requires less body movement, and also allows interaction at a distance from the display [6].

The indirect pointing usually includes mouse, trackballs, touchpads, joysticks and arrow keys. For typical pointing tasks the mouse is the most common to use on a desktop computer, one can point with the mouse about as well as with the hand itself [6]. A trackball senses the relative motion of a partially exposed ball in two degrees of freedom. Trackballs require frequent clutching movements because users must lift and reposition their hand after rolling the ball a short distance [6]. A Touchpad is a small and touch-sensitive tablets often found on notebook. Touchpads usually use relative mode for cursor control because they are too small to map to an entire screen, they necessitates frequent clutching when user use it. Most touchpads also have an absolute mode to allow interactions such as character entry [6]. There are two kind of joystick: (1) an isometric joystick is a force-sensing joystick that returns to center when released. (2) Isotonic joysticks sense angle of deflection. In isometric joysticks, the rate of cursor movement is proportional to the force exerted on the stick; as a result, users must practice in order to achieve good cursor control. The arrow keys are buttons on a computer keyboard that is programmed or designated to move the cursor in a specified direction. Arrow keys are commonly used for selection around documents [10].

2.2 Existing Solutions in Remote Screen Operation

As the screen size of mobile phones is ever increasing, such screen area become difficult for thumbs to cover and reach on one-handed operation scenarios. It causes that some target in this area can't be touched by thumb because there exist a big distance between thumb and targets. The solutions must shorten the distant between the target and thumb. We classify these further into direct and indirect interaction methods.

Direct Interaction: The Target Closed to the Thumb. This method is like put the target into the comfort zone of thumb. ArchMenu and ThumbMenu [11] apply the stacked pie menu that makes items to surround the thumb (Fig. 2a and b). This way facilitates one-handed interaction on small touchscreen devices. Nudelman [12] presented a C-Swipe gesture that let user use C-Swipe to raise a semicircular pie menu on the screen of smartphone, which surround the thumb. This method allows the user to tap the options which originally in the top of the screen (Fig. 2c). However, those methods can only tap the functions, it can't tap into the content, such as a list. As Apps become more complex, this approach does not necessarily meet the all of needs.

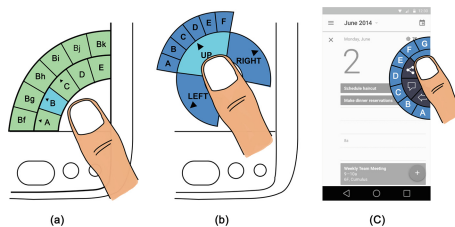


Fig. 2. (a) ArchMenu (b) ThumbMenu (c) using c-swipe to raise a semicircular pie menu (Source adapted from Hout et al. 2007 and Nudelman, 2013).

Faced with this problem there are plenty of manufacturers now that offer some interface features to facilitate the one-handed operation of today's big-screen phones (as shown in Fig. 3). When user open the one-handed mode that will resize the feature or entire screen and place it on the right or left of the screen. Apple also has one-handed mode and invented a name for the new feature, which called Reachability. It is activated by double tapping on the home key. It basically shrinks the interface, too, but in a manner that just slides it halfway down, so that you can reach whatever was unreachable at the top of the screen before. In summary those methods all move the distant target to the comfort zone of thumb then let the thumb can direct touch the target.

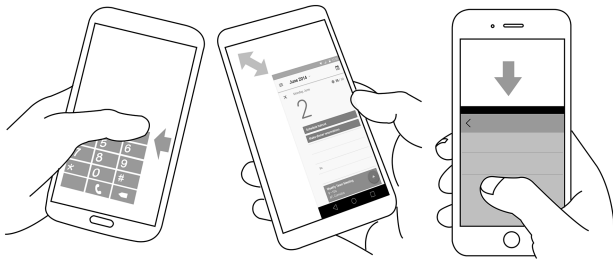


Fig. 3. Existing solution in large-screen smartphone

Indirect Interaction: Extending Thumb to Reach the Target. To solve this problem, we observe the operation scenarios of a touchpad TV remote control suitable for thumb use on one-handed operation scenarios. Choi et al. [13] show that a remote with small touchpad can control the cursor to pointing at the far target on large-screen TV (Fig. 4a). ThumbSpace [2], it's inspiration from the large screen devices and wall-sized displays both confront issues with out-of reach interface objects. ThumbSpace requires setup a proxy view that like an absolute touchpad (as show in Fig. 4b). When user's thumb touches a ThumbSpace area, the associated object on the screen is highlighted. Yu et al. [14] also introduced BezelSpace, a proxy region is the same as ThumbSpace, but the location of proxy region adaptively shifts according to any bezel swipe initial location on the screen (Fig. 4c). When use it, users need to continue to drag ones finger from the edge of screen and across the screen to control the mapped "magnetized" cursor and aim it towards the target. In summary those methods also utilized a cursor (pointing device) to select targets positioned at the farther end of the screen and it is like extending your thumb to reach the distant areas on screen.

The research shows that extendible cursor methods have the better performance than move the distant target to the comfort zone of thumb [15]. Yu showed that setting a virtual touchpad control the cursor to select the target has the perfect performance. It shows that the indirect methods may better than direct methods for resolving the thumb reach problem. However, the indirect pointing usually includes mouse, trackballs, joysticks, touchpads and arrow keys but which method has the better performance on pointing. The most common evaluation measures the efficiency of pointing devices are

speed and accuracy. Cart et al. [16] measured mean pointing times and error rates for the mouse, rate-controlled isometric joystick, step keys, and text keys. They found the mouse to be the fastest and lower error rate of the devices. MacKenzie et al. [17] compares the efficiency of pointing devices for the mouse, trackballs, touchpads and joysticks. The publication showed the mouse to be the fastest pointing device but they're no significant different between each method. In those researches, they evaluate several pointing devices in real but we were interested in users' behavior and efficiency when we employ those pointing techniques on virtual scenario.

In this paper we focus on the evaluation of indirect pointing devices in precision cursor positioning tasks on smartphone. Considering the operation behavior of touchscreen, two methods include mouse and trackballs may not suitable for using on touchscreen. They require frequent clutching movements because users must lift and reposition their hand after rolling the mouse and ball a short distance. The other hand those methods aren't suitable for virtualization. Finally we chose the touchpad, joysticks and arrow keys to evaluate the efficiency of pointing on touchscreen of smartphone.

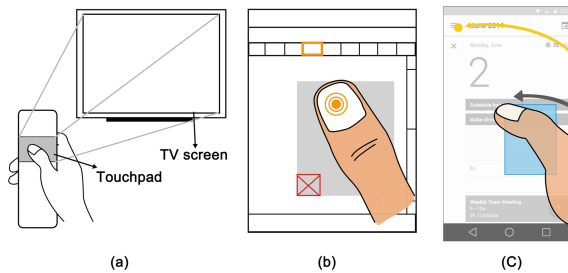


Fig. 4. (a) Using touchpad for TV remote control scenarios (b) ThumbSpace (c) BezelSpace (Source adapted from Choi et al. 2011, Kalson et al. 2007 and Yu et al. 2013).

3 Developing Interface of Virtual Pointing Control

We understand that the pointing technique can help us to point the target accurately by the cursor. However, there have not been any researches compare the efficiency of pointing techniques on large mobile touchscreen. We developed three mobile interaction techniques using pointing technique: Virtual Touchpad, Virtual Joystick and Virtual Directing Key. Our method includes two steps: (1) firstly, user performs the triggering gesture to open the virtual pointing interface (2) Secondly, user employ their thumb to operate the virtual pointing interface to select a target. We use the bezel swipe gesture as the triggering gesture. The bezel swipe gesture has the advantages of enabling users' thumbs to easily access functionality by activating a thin button [18]. Yu et al. [14] show that swipe gesture can adaptively find individual users comfortable range of motion for the thumb.

3.1 Virtual Touchpad

In this method, we set a semitransparent rectangle proxy region which mapping to the whole touch-screen of smartphone and it like an absolute touchpad. Avoiding the cursor suddenly jumps to the new position, we set a red dot on the rectangle area which position is mapping to the cursor (Fig. 5). In this way user can employ their thumb to drag the red dot on the rectangle area to control the cursor toward the target and it also fit the Fitt's law. The Virtual Touchpad operation scenario as following: (1) when user bezel swipe from the edge of screen, the touchpad and mapping cursor appear. (2) A user tap the red dot and drag it to control the cursor to the target and the cursor can dynamic capture closest target as the bubble cursor [5]. (3) The target is selected when a user's thumb lifts from the screen.

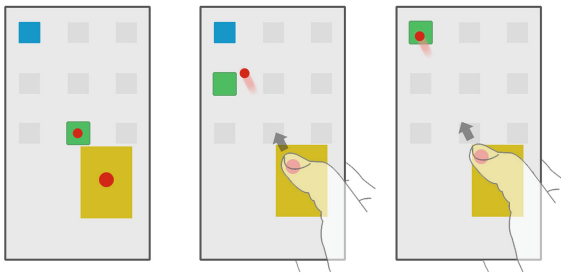


Fig. 5. The design of Virtual Touchpad

3.2 Virtual Joystick

This method use the joystick pointing technique and it control mode inspire from operation way of the game on the tablet. The red dot on the interface represents the stick of the joystick and the circular region represents the red dot can movable range (Fig. 6). When the cursor moves, it is based on the distance between red dot and the center of circular region, if the red dot farther form the center of circular region the cursor moves faster. The Virtual Joystick operation scenario as following: (1) when user bezel swipe from the edge of screen, the virtual joystick and cursor appear (Fig. 6b). (2) A user drag the red dot to control the cursor and it can dynamic capture closest target. (3) The target is selected when a user's thumb lifts from the screen.

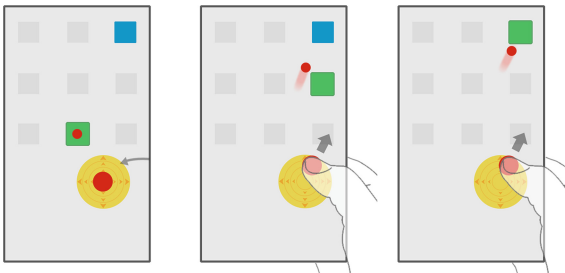


Fig. 6. The design of Virtual Joystick

3.3 Virtual Direction Key

This method use the arrow key pointing technique, it common use on computer keyboard and usually arranged in an inverted-T layout. Furthermore, this method is also common use to navigation or changes the function in feature phone by the number key. The keys can move the cursor to jump a target to another target in a specified direction (as object pointing). We refer to step key [19] and set the Virtual Direction key (Fig. 7). We propose that Virtual Direction Key works as follows: (1) object cursor appears when the bezel swipe occurs. (2) Users tap the direction key to control the object cursor and aim it towards the target. (3) The target is selected when a user's thumb tap the ok key.

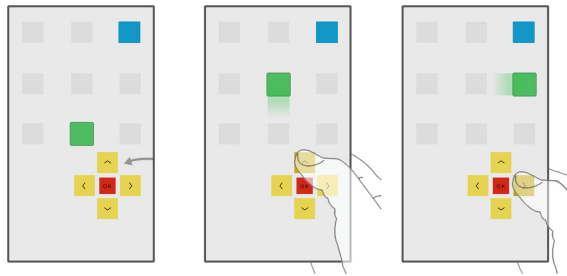


Fig. 7. The design of Virtual Direction Key

4 Usability Study and Feasibility Analysis

4.1 Usability Study

A usability study was designed to evaluate three kinds of cursor control interface: Virtual Touchpad, Virtual Joystick and Virtual Direction Key in terms of usability, efficiency and user satisfaction.

Devices and Participants. In this study, these interface techniques are implemented on the Android Platform, and the experiments ran on the Samsung Galaxy Note2 ($80.5 \times 151.1 \times 9.4$ mm, 5.5" display, 1280×720 screen resolution). Ten participants (7 Male, 3 Female), ranging in age from 21 to 38 years of age with an average age of 27 years of age, and all participants were right-handed and had experience with touchscreen based smartphones. The experiment required an hour per participant, and they were received NT\$150 upon the completion of the experiment.

Tasks. Kalson show that most people are used to operate the phone with one hand while walking or standing. Thus, in this study each participant was asked to stand and hold the device with the dominant hand and manipulate it only with the thumb while the experiment performed. They were asked to conduct a series of target selection tasks. According to the minimal touch area command in current mobile UI design guide, we set $7 \text{ mm} \times 7 \text{ mm}$ as the rectangular target size and the target color is grey in

normal situation. Based on the arrangement of icon on the home screen in current smartphone, we divided the screen into a 5×6 grid and the target will be evenly distributed in the grid. To ensure the users utilize the cursor control technique in every selection tasks, we ask each target should appear outside the thumb comfort zone. Targets appeared 9 times in a random order for each block. When target appear on screen there is only one target was painted blues for each trial; other keep grey. When the target was focused or selected, the color changed to green. When a participant succeeded in correctly selecting a target, they received haptic feedback through a vibration motor and the next target was generated immediately. If a participant failed to use pointing technique to select a target, no feedback was provided to ensure that the participant would try again. The participants were instructed to select the blue targets as quickly and accurately as possible.

Methods. We use a one-way repeated measures within-subjects design. The independent variables are Method (Virtual Touchpad, Virtual Joystick and Virtual Direction Key). Pointing technique order was randomized. A demonstration and practice phase was provided before each experiment. When the study began, users need to complete 6 task blocks for one pointing type and each block has 9 trials, and then repeated the process with the second pointing type.

In summary, the experimental design is:

- 10 participants
- × 3 Methods (Virtual Touchpad, Virtual Joystick and Virtual Direction Key)
- × 6 Blocks
- × 9 trials = 1,620 trials completed

After completing one method, participants were asked to complete the satisfaction questionnaire (a seven-point Likert scale) and interviewed for ten minutes. They were asked for any opinions about our interfaces, such as the reasons for their answers, frustrating experiences, and suggestions for each configuration.

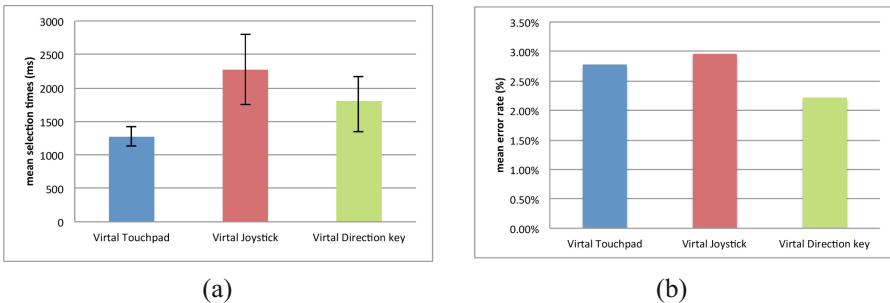


Fig. 8 (a) Mean selection times (b) Mean error rates

4.2 Data Analysis of User Study

Selection Time. Selection time was defined as the elapsed time between a target appearance and the target is selected successfully. Trials with selection errors were excluded from analysis. We analyzed results by one-way repeated-measure ANOVA and find a significant effect on selection time ($F_{2,27} = 29.649$, $p < .001$ with Mauchly Spherical Test, $p > .05$). Post hoc pairwise comparisons show that all methods differ significantly from each other, especial the Virtual Touchpad to Virtual Joystick and Virtual Direction Key. Overall, Virtual Touchpad is significantly faster ($M = 1271.01$ ms, $SD = 175.45$ ms) than Virtual Joystick ($M = 2272.179$ ms, $SD = 522.65$ ms) and Virtual Direction Key ($M = 1806.1$ ms, $SD = 374.4$ ms) (Fig. 8a).

Error Rate. The error rate was defined as the number of erroneous selections during one block. It includes empty and wrong target selections. We analyzed results by one-way repeated-measure ANOVA but we find there is no significant effect on error rate ($F_{2,27} = 0.251$, $p > .05$). Post hoc pairwise comparisons show that all methods no significant with each other. The object pointing and semantic pointing may explain this. Because in Virtual Touchpad and Virtual Joystick, the cursor can dynamic capture the closest target. In this approach, a user doesn't need to accurately move the cursor to target. Overall, Virtual Direction Key is more accuracy ($M = 2.78$ %, $SD = 3.3$ %) than Virtual Joystick ($M = 2.96$ %, $SD = 2.5$ %) and Virtual Touchpad ($M = 2.2$ %, $SD = 2.87$ %) (Fig. 8b).

User Satisfaction. After experiments done we do a questionnaire to evaluate the user satisfaction. In this questionnaire Virtual Touchpad has a great performance across all categories on a 7-point Likert scale. We believe the shape of virtual touchpad mapping to the touch screen and the cursor can dynamic capture the nearest target (Semantic pointing) cause this result. Participants also report that they can move their thumb easily by the corresponding direction of target on virtual touchpad (Fig. 9).

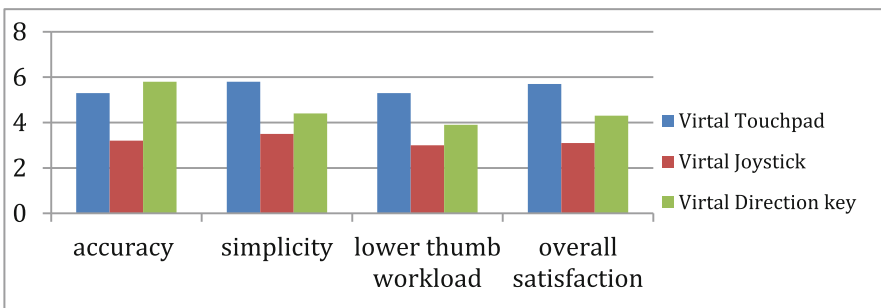


Fig. 9 User satisfaction score

4.3 Focus Group Interview and Evaluation

All models (design) in this study were developed to improve selection operation for targets that are out of thumb reach. While the usability study was carried out in a controlled environment, the real user's mobility context should take moving vehicle and surrounding crowds into consideration. Thus, the operation details of each model need to be carefully examined to ensure that they meet the challenge of mobile context.

A focus group interview to seven experienced usability designers was carried out. The **first stage** of the focus group interview was to define constraints of touchscreen cursor control in mobile context, followed by the **second stage**, in which every individual model was rated against constraints of cursor control in mobile context.

Findings of Focus Group Interview. Defined constraints of cursor in mobile context include:

1. Stable and simple operation are required. People may use mobile devices when they're standing, walking, riding a bus or train.
2. Interruptible operation is better. Any operation required continuous thumb tap and drag is not suitable in the mobile context, in which a user's operation might be disturbed by passersby, jolts on a vehicle or jerks of surrounding crowds.
3. Avoid accurate pointing. Selection targets in the virtual pointing control panel should not be smaller than 9.2 mm [20].

Evaluation of Experienced Usability Designers. The evaluation of cursor control models was done by the same group of experienced usability designers. The average rating is shown in Table 1. Virtual Direction Key stood out as the most stable, simple and Interruptible pointing control.

In addition, operation feedback should be taken into consideration, according to the interviewees.

Table 1. Evaluation of cursor control model against constraints of thumb operation in mobile context.

Constraints / virtual cursor control	Virtual Touch-pad	Virtual Joystick	Virtual Direction key
Stable and simple operation	Satisfied	Fair	Satisfied
Interruptible operation	Unsatisfied	Unsatisfied	Satisfied
Avoid accurate pointing	Fair	Fair	Satisfied

4.4 Discussion

The results in our experiment show that Virtual Touchpad performed a better efficiency than others did. Most users reported that they could smoothly control the cursor through Virtual Touchpad. This is because the shape and scale of virtual touchpad was designed for mapping to the touchscreen that make the cursor control predictable.

However, the results show that the Virtual Joystick has the worst performance in both efficiency and satisfaction. All participants reported that the movement of cursor is inconsistent with thumb operation. In our observation, when users intended to shift moving direction of the cursor, virtual joystick provided no feedback to the thumb, which confused the user. As a result, in the context of virtual pointing control, the more consistence between thumb and cursor, the better efficiency and satisfaction were reported.

Although Virtual Direction Key performed less efficient than Virtual Touchpad did, but most users remarked its accuracy. Meanwhile, users also reported that they sometimes feel impatient because this method requires frequent taping the direction key. However, experts regarded that Virtual Direction Key is a better means of touchscreen cursor control in mobile context.

5 Conclusion

This study formulated existing virtual pointing techniques for cursor control in the mobile context of touchscreen thumb operation. Three virtual pointing models were developed, including: Virtual Touchpad, Virtual Joystick and Virtual Direction Key.

The usability tests reveal: (1) Virtual Touchpad was significantly efficient than others, while Virtual Direction Key presented lower error rate, although insignificantly. (2) Virtual Touchpad has a favored performance across most categories (accuracy, simplicity, thumb workload, and overall satisfaction) in user satisfaction test.

The feasibility study remarks: (1) constraints of touchscreen cursor control in mobile context include: stable and simple operation, interruptible operation is better and avoid accurate pointing. (2) Virtual Direction Key stood out as the most stable, simple and interruptible pointing control. (3) All virtual pointing models require further improvement on operation feedback. As a result, taking usability and feasibility into consideration, the Virtual Direction Key is a better means of touchscreen cursor control in mobile context.

References

1. Karlson, A.K., Bederson, B.B., Contreras-Vidal, J.L.: Understanding single-handed mobile device interaction. In: Lumsden, J. (ed.) *Handbook of Research on User Interface Design and Evaluation for Mobile Technology*, pp. 86–101. National Research Council of Canada Institute for Information Technology, Hershey (2006)
2. Karlson, A.K., Bederson, B.B.: ThumbSpace: generalized one-handed input for touchscreen-based mobile devices. In: Baranauskas, C., Abascal, J., Barbosa, S.D.J. (eds.) *INTERACT 2007*. LNCS, vol. 4662, pp. 324–338. Springer, Heidelberg (2007)
3. Clark, J.: *Tapworthy: Designing Great iPhone Apps*. O'Reilly Media, Canada (2010)
4. Hooper, S.: How do users really hold mobile devices? *UX- matters* (2013). <http://goo.gl/SbDQXA>. Accessed 18 Sep 2014
5. Grossman, T., Balakrishnan, R.: The bubble cursor: enhancing target acquisition by dynamic resizing of the cursor activation area. In: *Proceedings of CHI 2005*, pp. 281–290 (2005)

6. Hinckley, K.: Input technologies and techniques. In: Jacko, J.A., Sears, A. (eds.) *The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies, and Emerging Applications*, pp. 151–168. Lawrence Erlbaum Associates, London (2002)
7. Shneiderman, B.: Touchscreens now offer compelling uses. *IEEE Softw.* **8**(2), 93–94 (1991)
8. Vogel, D., Baudisch, P.: Shift: a technique for operating pen-based interfaces using touch. In: *Proceedings of CHI 2007*, pp. 657–666 (2007)
9. Forlines, C., Wigdor, D., Shen, C., Balakrishnan, R.: Direct-touch vs. mice input for tabletop displays. In: *Proceedings of CHI 2007*. ACM, pp. 647–656 (2007)
10. Rose, C., Hacker, B.: *Inside Macintosh: More Macintosh Toolbox*. Apple Computer, Inc., Addison-Wesley Pub. Co., Cupertino, Boston (1985)
11. Huot, S., Lecolinet, E.: ArchMenu et ThumbMenu: Contrôler son dispositif mobile “sur le pouce”. In: *Proceedings of ICPS IHM 2007*, pp. 107–110 (2007)
12. Nudelman, G.: C-Swipe: An Ergonomic Solution To Navigation Fragmentation On Android. *Smashing Magazine* (2013). <http://goo.gl/OG3o1k>. Accessed 20 Sep 2014
13. Choi, S., Han, J., Lee, G., et al.: RemoteTouch: touch-screen-like interaction in the TV viewing environment. In: *Proceedings of CHI 2011*, pp. 393–402. ACM Press (2011)
14. Yu, N. H., Huang, D.Y., Hsu, J.J., Hung, Y.P.: Rapid selection of hard-to-access targets by thumb on mobile touch-screens. In: *Proceedings of MobileHCI 2013*, pp. 400–403 (2013)
15. Kim, S., Yu, J., Lee, G.: Interaction techniques for unreachable objects on the touchscreen. In: *Proceedings of OzCHI 2012*, pp. 295–298 (2012)
16. Oulasvirta, A., Tamminen, S., Roto, V., Kuorelahti, J.: Interaction in 4-second bursts: the fragmented nature of attentional resources in mobile HCI. In: *Proceedings of CHI 2005*, pp. 919–928. ACM Press (2005)
17. Smith, T.F., Waterman, M.S.: Identification of common molecular subsequences. *J. Mol. Biol.* **147**, 195–197 (1981)
18. Card, S., English, W., Burr, B.: Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a CRT. *Ergonomics* **21**(8), 601–613 (1978)
19. MacKenzie, I.S., Kauppinen, T., Silfverberg, M.: Accuracy measures for evaluating computer pointing devices. *Proceedings of ACM CHI 2001*, pp. 9–16 (2001)
20. Parhi, P., Karlson, A.K., Bederson, B.B. Target size study for one-handed thumb use on small touchscreen devices. In: *Proceedings of Mobile HCI 2006*, pp. 203–210. ACM Press (2006)