Fingerprint Scroll: Comparison of Touchless and Touch-Based Scroll Navigation Methods

Saurabh Garg, I. Scott Mackenzie

Fingerprint Scroll: Comparison of Touchless and Touch-based Scroll Navigation Methods

Saurabh Garg[1] and Scott Mackenzie[2]

[1] York University, Toronto ON M3J 1P3, Canada
saurabh.garg5991@gmail.com
[2] York University, Toronto ON M3J 1P3, Canada
mack@cse.yorku.ca

Abstract. We conducted a user study comparing the efficiency and speed of three scroll navigation methods for touch-screen mobile devices: Tap Scroll (the traditional touch-based method), Kinetic Scroll (a touch-based gestural method), and Fingerprint Scroll (our newly introduced hybrid method). The study involved 12 participants and employed a Google *Pixel* device. The accuracy results for Fingerprint Scroll were higher, with an average accuracy of 97.6% compared to 96.3% for Tap Scroll and 88.7% for Kinetic Scroll. The completion time for Fingerprint Scroll was 2.43 times longer than that for Kinetic Scroll and 2.32 times longer than for Tap Scroll. Despite the long completion times, participants showed a preference for Fingerprint Scroll because of ease-of-use, high accuracy, and interruption-free vision while reading ebooks. The latter is a significant issue for Kinetic and Tap Scroll because gestures are touch-based and hence the display view is obscured. Participants did not value the higher speed of navigation using Kinetic Scroll over Fingerprint Scroll for web browsing.

Keywords: Fingerprint Scroll · Fingerprint sensor · scroll navigation · scroll method · mobile interaction · hybrid interaction · Human - Computer Interaction

1 Introduction

Mobile devices such as tablets, phones, handhelds, or personal digital assistants (PDAs), provide easy and convenient access to information at the tap of your finger virtually anywhere and anytime. However, the small screen and limited input capabilities impact the user experience with these devices. This is additionally muddled by the very idea of these gadgets – versatility and mobility. Mobile phones are frequently utilized in dynamic and busy environments, for example, while walking, running, standing, etc. This makes designing interaction techniques for mobile phones difficult and challenging. The classical techniques used on a desktop computer may not be always accurate.

1.1 Input Styles for Mobile Devices

A significant challenge with mobile devices is providing appropriate input that is easy to use and can accurately work on tasks. Numerous tasks such as target selection, text entry, or navigating user interfaces are challenging. Broadly speaking, input styles can be classified into three categories: software, hardware, and hybrid. Hardware-based input methods utilize physical components on the device. Software-based input methods are built inside real applications such as touch software buttons or scrollbars. Hybrid-based input methods combine the approaches from software and hardware input techniques.

Hardware-based Input Most mobile phones have hardware components and accessories to provide different kinds of inputs. For example, physical buttons can be programmed as a "home" button or a "mode" button suitable for different purposes depending on the context. Hence, the button is not fixed absolutely but changes based on the current perspective of the screen. Another example is a scroll wheel for navigating a document vertically. In 1996, Rekimoto introduced the idea of navigating using the device itself and added a sensor that would detect device tilt and rotation when the user moved the device [16]. Figure 1 demonstrates the use of tilt for navigating a map. The utilization of tilt for navigation and selection tasks [1–3, 6, 7] and text input on mobile devices [13, 20] has been explored extensively. Researchers have combined tilt with other external inputs like buttons [13, 19], gestures, and other sensors [9]. A key advantage of hardware-based input interactions is that they can provide single-handed interaction. NaviPoint [9] and ScrollPad [4] are other examples of hardware-based inputs.

Software-based Input Software-based input ordinarily utilizes a stylus or finger on a touch-sensitive display. For general navigation in smartphones, the most common strategy uses scrollbars that are created in software. Even though this technique is familiar to most users, it has limitations on the desktop [22]. Using the same technique on a mobile device is challenging. A study by Smith and schraefel [18] identified three ways of user interaction with scrollbars and potential challenges with each. Figure 2 illustrates the operation of a scrollbar. Users can drag the handle of the scrollbar by maintaining constant pressure on the button of the mouse. This might result in skipping important parts caused by users unintentionally or letting go of the thumb and skipping the desired parts of the document. Due to smaller screen sizes, this problem can become even more critical on mobile devices. Secondly, users can click on the arrow buttons at each end of the scrollbar to move the document. But, this can be slow and tiring. Users can also click positions on the scrollbar to jump to a particular segment of the document. But, this can be disorienting [3, 22] and the problem is compounded on smaller screens. Scrollbars require the user to draw their attention from the document to a software-designed scrollbar which may require additional cognitive effort and motor resources. Also, scrollbars restrict

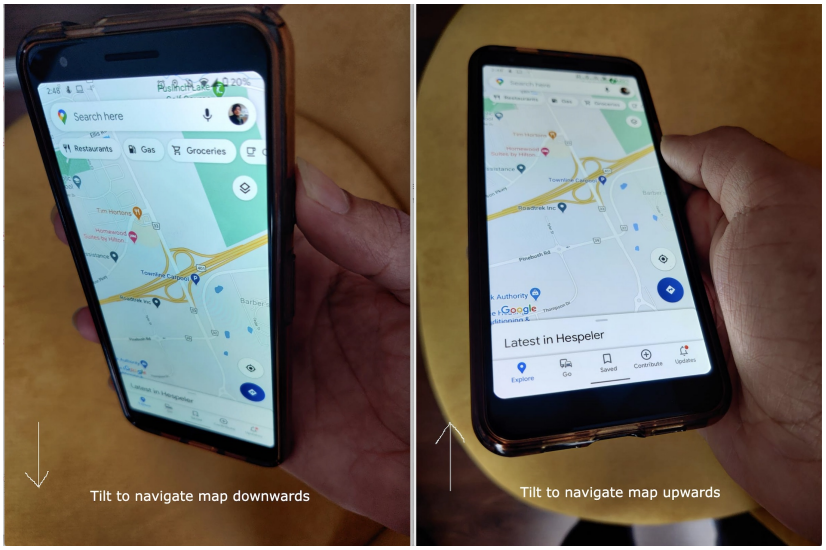

Fig. 1: Navigation of map by tilting a display.

movement to a single direction (i.e., horizontal or vertical).

Another software-based method for navigation on a mobile device is tap-and-drag. This technique requires the user to tap anywhere on the screen with a stylus and drag in the display area in any direction. This is a familiar interaction for desktop users. Johnson [8] compared different drag techniques with edge navigation on a touch screen and found that users were faster and had a preference for drag to navigate. One major disadvantage of using dragging is that it is not suitable for documents containing many selection targets. Also, dragging is limited by the size of the screen as users can only drag in increments of screen size [10]. Research has also studied the use of touch gestures on mobile devices. For example, Harrison et al. [6] used finger gestures to mimic flipping pages in a digital document. The advantage of software-based input is that it is built into the application itself and can use paradigms similar to a desktop application. The main drawback is that these interactions often require the use of both hands. Sometimes users prefer to hold a device with one hand while the other hand interacts with the device to avoid missing targets.

Hybrid-based Input Hybrid methods for scrolling combine external hardware-based input with software-based input. Usually, this involves using the hand to hold the mobile device while navigating displayed information (e.g., tilt) and using the freehand to make selections. Peephole displays [21] utilize a spatially-aware device that is moved to reveal different parts of an information space while making selections with a stylus. Eslambolchilar and Murray-Smith [2] and

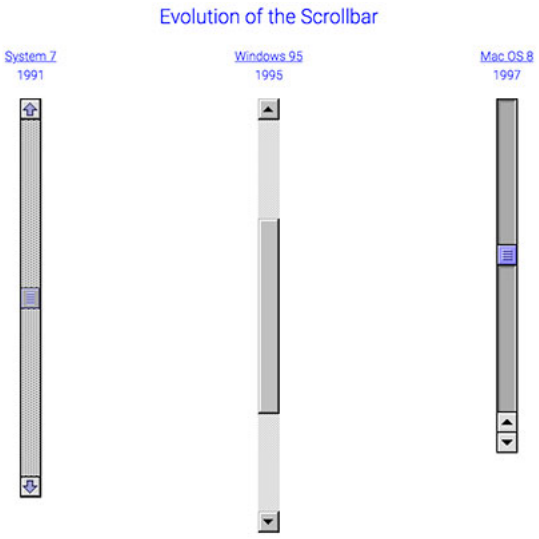

Fig. 2: Evolution of the Scrollbar.

Eslambolchilar et al. [3] coupled an SDAZ (speed-dependent automatic zooming) system with tilt to navigate and scroll through information while using a stylus for selection.

Interaction techniques for mobile devices all face the common challenge of needing to be used while the user is mobile. In general, it is hard for users to navigate and select items while mobile, regardless of the input method they are using. Research has shown that tactile [15] or audio feedback [3, 14] can improve users' accuracy and efficiency when performing navigation and selection tasks.

We introduce a new method called Fingerprint Scroll that utilizes a hybrid approach using an external hardware fingerprint sensor placed on a mobile device. In most devices, it exists on the rear of a mobile device. It does not block the user's view of the display while in use. Swiping up on the sensor scrolls up the page and swiping down on the sensor scrolls down the page. For comparison, we will also include two other methods for scroll navigation. Kinetic Scroll utilizes swipe gestures on the screen to navigate and scroll. Tap Scroll is another method that uses on-screen taps over arrow buttons to scroll in the desired direction.

In this paper, we evaluate and compare the performances of touch-based and touch-less scroll navigation methods. We compare the three scroll methods mentioned above. The results obtained are presented after that.

2 Related Work

Table 1 summarizes the results from four papers where user studies were performed utilizing scroll navigation methods.

Table 1: Summary of results from four papers where user studies were performed utilizing scroll navigation methods.

1st Author	Type	n	Notes
Smith [18]	Touch	8	Radial Scroll performs better than traditional methods on short scroll target acquisition.
MacKay [10]	Touch	18	The techniques of touch-n-go and tap-and-drag outperformed the traditional scrollbar technique for the simple navigation tasks.
Moscovich [11]	Touch	10	The virtual scroll ring is a tenable scrolling alternative. This is especially true when most scrolling actions are expected to be longer than half a page.
Oakley [12]	Touchless	12	User performance using the position-based input was good and provided promise, and allowed us to select optimum parameters for position-based list navigation.
n = number of participants			

Smith [18] assessed a scroll navigation method called *radial scroll*, which was effective for variable fast document scrolling for touch screen devices. He also anticipates improving radial scrolling by presenting distant headings and giving the user the ability to change aspects of the scrolling mid-scroll, such as switching between per-line verses per-page scrolling.

MacKay [10] compared software-based navigation techniques with the new *touch-n-go* approach on a mobile device. In terms of preference, users found *touch-n-go* easier than the *scrollbar* and *tap-and-drag* methods during multiple levels of mobility. Participants achieved better performance while sitting but were considerably slower while walking for all of the navigation techniques.

Moscovich [11] assessed the virtual scroll ring as a tenable scrolling alternative. This was especially true when scrolling is longer than half a page. In the case of smaller than a half-page, extra care should be taken to ensure that enough data is collected for a robust estimate of the circle. Since the virtual scroll ring scrolls the view smoothly in increments as small as one pixel, it allows users to read the text while they scroll.

Oakley [12] believed position-based input mapping had considerable potential. One reason is that users can learn to reach specific list items, which in turn, may lead to an open loop interaction, where an item can be selected with confidence without explicitly requiring feedback from the system. It would also be interesting to look at the implications of using this input technique with nested menu systems, where multiple selections are made to reach a single goal. In conclusion, the author believed that interfaces based around motion input and vibrotactile output have an important role in next-generation mobile interaction techniques.

3 Methodology

A user study was conducted to compare the three scroll navigation methods. The goal was to compare the three scroll methods in terms of quantitative measures, user preference, and ease of use.

3.1 Participants

Twelve participants were recruited remotely from different universities across Canada. Six were male, six were female. Ages ranged from 22 to 26 years. All participants were comfortable with using smartphones. All were right-handed and sat during testing. Participants were constrained to use only a single hand. They had prior experience of using the Kinetic Scroll method but no prior experience in using Tap Scroll or Fingerprint Scroll method. Participants were compensated $20 for their assistance.

3.2 Apparatus

The experiment was conducted on Google *Pixel 3a* running the Android 11.0 operating system. The device has a 5.6-inch OLED display with a resolution of 2220 x 1080 pixels and a density of 441 PPI. The weight of the device is 147 grams.

The software was developed in Java using the Android SDK in the Android Studio. This experimental application was developed specifically for this research. Three scroll methods were implemented. The application begins with a configuration activity that prompts the user to select the participant code and other experimental parameters, like input method, session code, group code (for counterbalancing). Once configured, the user presses the SUBMIT button to initiate the testing process. The main activity contains a scrolling user interface. See Figure 3.

3.3 Procedure

Participants were informed and explained the purpose of the user study. They were requested to keep the Internet switched on during the entire experiment. Participants watched a video explaining the interaction methods and did some practice trials. They were asked to use just one hand for all interactions. Arm support was not allowed. With this brief introduction, testing began. Participants completed five blocks for each method. A block consists of five trials. A participant took about 15-20 minutes to complete the experiment. The data were stored in a remote database in Google *Firebase* and analyzed later for results and meaningful insights. After testing was complete, the user was prompted with a questionnaire to gather feedback regarding the preference of scroll methods. Participants were asked the following questions -

– How many hours do you use a smartphone per day?

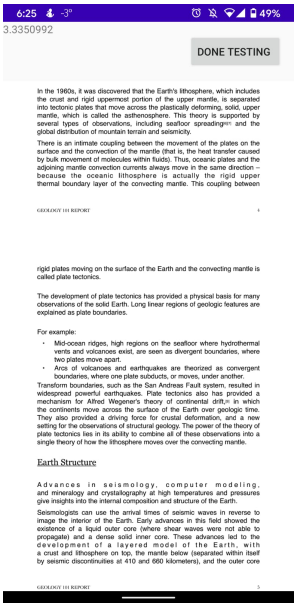

Fig. 3: Main activity for the scroll navigation application

– Do you regularly use a smartphone?
– Do you browse the web on a smartphone?
– Do you read books on your smartphone?
– Which method would you prefer for web browsing?
– Which method would you prefer for reading books?
– Overall preference of method on a scale of 1 (Least likely) to 5 (most likely) for scroll method?

3.4 Design

The experiment was a 3 × 5 within-subjects design with the following independent variables and levels:

– Scroll Method: Kinetic Scroll, Tap Scroll, Fingerprint Scroll
– Block Code: 1, 2, 3, 4, 5

Each participant completed five blocks of five trials for each scroll method. As such, the total number of trials was 12 Participants × 3 Scroll Methods × 5 Blocks × 5 Trials = 900.

Participants were counterbalanced using a Latin square method. The application grouped the participants into three groups using a group code assigned to every participant. Each group tested in a specific sequence of scroll methods. The dependent variables were completion time (s) and error rate (%). The duration of the experiments was roughly one hour per participant.

4 Results and Discussion

All trials were completed successfully. The data were later imported into a spreadsheet tool where summaries of various measures were calculated and charts were created. The analysis of variance[3] test was performed using the GoStats[4] application.

4.1 Completion Time

The grand mean for completion time was 13.62 s. From Figure 4, we observe that Tap Scroll took 4.71% more time to complete a similar trial than Kinetic Scroll. The completion time of Fingerprint Scroll was 2.43 times longer than that of Kinetic Scroll and 2.32 times longer than Tap Scroll.

The effect of the group on completion time was not statistically significant ($F_{2,9} = 0.494$, ns), thus indicating that counterbalancing had the desired effect of offsetting order effects. The effect of scrolling method on completion time was statistically significant ($F_{2,18} = 1099.3$, $p < .0001$). The Scrolling Method $\times$ Group interaction effect was not statistically significant, however ($F_{4,18} = 0.357$, ns). A Fisher LSD post hoc pairwise comparison test indicated a significant difference between the Fingerprint Scroll method and each of the Kinetic Scroll and Tap Scroll methods ($p < .05$).

The effect of block on completion time was statistically significant ($F_{4,36} = 4243.7$, $p < .0001$). The Block $\times$ Group interaction effect was not statistically significant ($F_{8,36} = 0.241$, ns). The Scrolling Method $\times$ Block interaction effect was statistically significant ($F_{8,72} = 375.2$, $p < .0001$).

4.2 Error Percentage

Errors were logged when a user fails to do a task correctly. For example, the task can be "go to page 6 and come back to page 1". If the user goes to a different page than instructed, it counts as an error. Error percentage is the percentage of total such error cases with respect to total cases. The effect of group on error percentage was not statistically significant ($F_{2,9} = 0.150$, ns). The effect of scrolling method on error percentage was statistically significant ($F_{2,18} = 73.50$, $p < .0001$). The scrolling method $\times$ group interaction effect was also not statistically significant ($F_{4,18} = 0.288$, ns). The Fisher LSD pairwise comparison

[3] A parametric analysis of variance was used even though the data did not meet the underlying distribution assumptions. There are a few reasons for this. The dependent variables were all ratio-scale user performance measurements, as opposed to measurements on an interval or ordinal scale. So, a parametric test, which is intended for ratio-scale data, is the natural test to use. Notably, as well, using the parametric analysis of variance avoids the inevitable loss of information that occurs when ratio-scale data are down-graded to ranks for a non-parametric test. Finally, the parametric analysis of variance is known to be robust to violations in the underlying distribution assumptions [5, 17].

[4] http://www.yorku.ca/mack/GoStats/

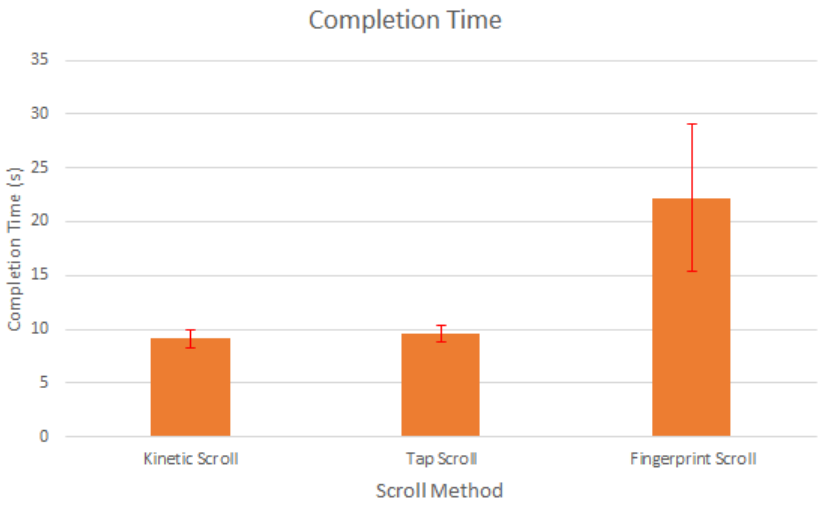

Fig. 4: Completion Time by scroll method. Standard deviation is shown as red error bars.

test revealed a significant difference between the Kinetic Scroll method and each of the Tap Scroll and Fingerprint Scroll methods ($p < .05$).

The error percentage differences between different methods can also be talked about in terms of accuracy. More error percentage leads to lower accuracy.

From the graph in Figure 5, The accuracy results for fingerprint scroll were higher, with an average accuracy of 97.6% compared to 96.3% for tap scroll and 88.67% for Kinetic Scroll.

4.3 Participant Feedback

In the post-experiment questionnaire, participants were asked their preference of the scroll navigation method on a scale of 1 (least likely) to 5 (most likely), and their answers were recorded. Using a Friedman test, the differences in opinion were statistically significant ($\chi^2 = 15.54$, $p = .0004$). The standard deviation for the preference rating of the kinetic scroll, tap scroll, fingerprint scroll was recorded as 0.37, 0.58, and 0.62. Using a post hoc test, it was also observed that all three pairwise comparisons were statistically significant. See Table 2.

Most participants stated that they found kinetic scroll frustrating to use while reading ebooks since it blocks the view and creates a bad user experience. Most preferred using Fingerprint Scroll as it does not block the vision and they do not consider the speed that important for a majority of tasks like reading ebooks or web browsing. One participant noted:
Fingerprint Scroll is great as I read a lot of books on my phone during commuting. I don't want to see the finger coming in my way of vision all the time.

Fig. 5: Error Percentage by scroll method. Standard deviation is shown as red error bars.

Table 2: Post-hoc pairwise comparison test of scroll methods.

Comparison	Difference	Significance
Kinetic vs *Tap*	$abs(2.708 - 1.125) > 0.422$	Significant
Kinetic vs *Fingerprint*	$abs(2.708 - 2.167) > 0.422$	Significant
Tap vs *Fingerprint*	$abs(1.125 - 2.167) > 0.422$	Significant

Overall participants praised Fingerprint scrolling and Kinetic Scroll in a majority for various usage. The average rating for Fingerprint Scroll, Tap Scroll, and Kinetic scroll was 4.3, 3, and 4.1 on a scale of 1-5.

5 Conclusion

A user study was conducted comparing the performance of three scroll navigation methods. The results indicated higher accuracy for our newly introduced method (Fingerprint Scroll) compared to two alternatives. The accuracy for fingerprint scroll was higher, with an average accuracy of 97.6% compared to 96.3% for tap scroll and 88.67% for Kinetic Scroll. The completion time was relatively long for Fingerprint Scroll compared to other traditional alternatives like Kinetic Scroll. Overall participants gave a favorable and preferential rating for Fingerprint Scroll for ebook reading and web browsing over mobile devices.

References

1. Bartlett, J.F.: Rock'n'scroll is here to stay. IEEE Computer Graphics and Applications **20**(3), 40–45 (2000)
2. Eslambolchilar, P., Murray-Smith, R.: Tilt-based automatic zooming and scaling in mobile devices: A state-space implementation. In: International Conference on Mobile Human-Computer Interaction. pp. 120–131. Springer, Berlin (2004)
3. Eslambolchilar, P., Williamson, J., Murray-Smith, R.: Multimodal feedback for tilt controlled speed dependent automatic zooming. In: Proceedings of Annual ACM Symposium on User Interface Software and Technology – UIST 2004. ACM, New York (2004)
4. Fallman, D., Lund, A., Wiberg, M.: Scrollpad: Tangible scrolling with mobile devices. In: Proceedings of the 37th Annual Hawaii International Conference on System Sciences. p. 6. IEEE, New York (2004)
5. Glass, G.V., Hopkins, K.D.: Statistical methods in education and psychology. Prentice-Hall, Englewood Cliffs, NJ, 3rd edn. (2008)
6. Harrison, B.L., Fishkin, K.P., Gujar, A., Mochon, C., Want, R.: Squeeze me, hold me, tilt me! An exploration of manipulative user interfaces. In: Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems – CHI '98. pp. 17–24. ACM, New York (1998)
7. Igarashi, T., Hinckley, K.: Speed-dependent automatic zooming for browsing large documents. In: Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology – UIST 2000. pp. 139–148. ACM, New York (2000)
8. Johnson, J.A.: A comparison of user interfaces for panning on a touch-controlled display. In: Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems – CHI '95. pp. 218–225. ACM, New York (1995)
9. Kawachiya, K., Ishikawa, H.: Navipoint: An input device for mobile information browsing. In: Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems – CHI '98. pp. 1–8. ACM, New York (1998)
10. MacKay, B., Dearman, D., Inkpen, K., Watters, C.: Walk'n scroll: A comparison of software-based navigation techniques for different levels of mobility. In: Proceedings of the 7th International Conference on Human-Computer Interaction with Mobile Devices & Services. pp. 183–190. ACM, New York (2005)
11. Moscovich, T., Hughes, J.F.: Navigating documents with the virtual scroll ring. In: Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology – UIST 2004. pp. 57–60. ACM, New York (2004)
12. Oakley, I., O'Modhrain, S.: Tilt to scroll: Evaluating a motion based vibrotactile mobile interface. In: First Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. World Haptics Conference. pp. 40–49. IEEE, New York (2005)
13. Partridge, K., Chatterjee, S., Sazawal, V., Borriello, G., Want, R.: Tilttype: Accelerometer-supported text entry for very small devices. In: Proceedings of the 15th Annual ACM Symposium on User Interface Software and Technology – UIST 2002. pp. 201–204. ACM, New York (2002)
14. Pirhonen, A., Brewster, S., Holguin, C.: Gestural and audio metaphors as a means of control for mobile devices. In: Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems – CHI 2002. pp. 291–298. ACM, New York (2002)
15. Poupyrev, I., Rekimoto, J., Maruyama, S.: Touchengine: A tactile display for handheld devices. In: Proceedings of Extended Abstracts of the ACM SIGCHI Confer-

ence on Human Factors in Computing Systems - CHI EA 2002. pp. 644–645. ACM, New York (2002)
16. Rekimoto, J.: Tilting operations for small screen interfaces. In: Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology – UIST 1996. pp. 167–168. ACM, New York (1996)
17. Sheskin, D.: Handbook of parametric and non-parametric statistical procedures. CRC Press, Boca Raton, FL, 5th edn. (2011)
18. Smith, G.M., schraefel, m.: The radial scroll tool: Scrolling support for stylus- or touch-based document navigation. In: Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology – UIST 2004. pp. 53–56 (2004)
19. Weberg, L., Brange, T., Hansson, Å.W.: A piece of butter on the PDA display. In: Proceedings of the Extended Abstracts of the ACM SIGCHI Converene on Human Factors in Computing Systems – CHI EA 2001. pp. 435–436. ACM, New York (2001)
20. Wigdor, D., Balakrishnan, R.: Tilttext: Using tilt for text input to mobile phones. In: Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology – UIST 2003. pp. 81–90. ACM, New York (2003)
21. Yee, K.P.: Peephole displays: Pen interaction on spatially aware handheld computers. In: Proceedings of the ACM SIGCHI Conference on Human Factors in Computing System – CHI 2003. pp. 1–8. ACM, New York (2003)
22. Zhai, S., Smith, B.A., Selker, T.: Improving browsing performance: A study of four input devices for scrolling and pointing tasks. In: Proceedings of INTERACT 97. pp. 286–293. Springer, Berlin (1997)