Press & Tilt: One-handed Text Selection and Command Execution on Smartphone

Toshiyuki Ando, Toshiya Isomoto, Buntarou Shizuki, and Shin Takahashi
University of Tsukuba, Japan
{ando,isomoto,shizuki,shin}@iplab.cs.tsukuba.ac.jp

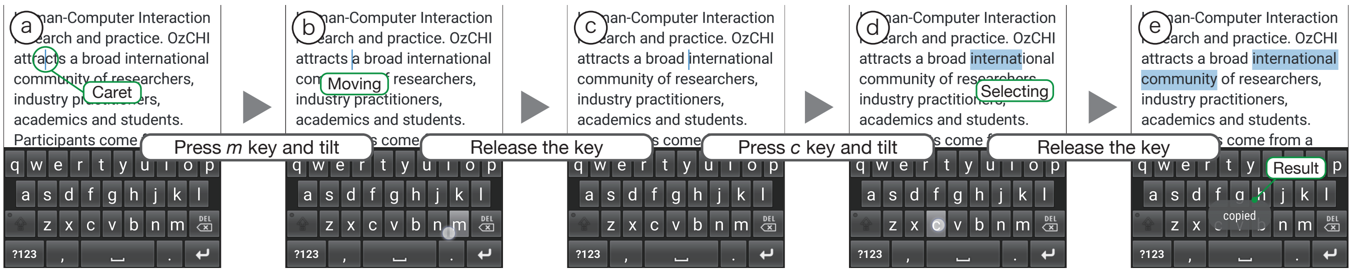

Figure 1: Press & Tilt allows text selection and command execution using only the thumb of the hand holding the smartphone. For example, to copy "international community" in (a), the user presses the *m* key and tilts the smartphone right to initiate carat navigation; the caret moves right (b). Then, after terminating carat navigation by releasing the key (c), the user presses the *c* key and tilts the smartphone right to initiate text selection and expand the selection range to the right (d). Releasing the key copies the text to the clipboard (e).

ABSTRACT

We show a text selection and text command execution method for a smartphone by tilting called Press & Tilt. The user can perform caret navigation or text selection by tilting the smartphone while pressing a key of the software keyboard. Then, by releasing the pressed key, text commands such as copy, search, and translate based on the selected text is executed; the executed text command depends on the pressed key. Neither occlusion nor the fat finger problem is of concern, because our method can perform these operations without the need to have a finger touch the upper region of the touchscreen. Also, the user can execute text commands with only one-hand.

CCS CONCEPTS

• **Human-centered computing** → **Gestural input**; **Keyboards**; *Interaction techniques*;

KEYWORDS

Soft keyboard, caret navigation, single-hand, tilt operation

ACM Reference Format:

Toshiyuki Ando, Toshiya Isomoto, Buntarou Shizuki, and Shin Takahashi. 2018. Press & Tilt: One-handed Text Selection and, Command Execution on Smartphone. In *Proceedings of the 30th Australian Computer-Human Interaction Conference (OzCHI '18), December 4–7, 2018, Melbourne, VIC, Australia.*ACM, New York, NY, USA, 5 pages.
https://doi.org/10.1145/3292147.3292178

OzCHI '18, December 4–7, 2018, Melbourne, VIC, Australia

ACM ISBN 978-1-4503-6188-0/18/12…$15.00
https://doi.org/10.1145/3292147.3292178

1 INTRODUCTION

Currently, in order to select text and execute a text command (i.e., copy, cut, or paste) on a smartphone, the user presses the text for a few moments, moves adjusters for adjusting the selection range, and then chooses a text command from a menu that appears near the selected text. Although this process is easy, several problems are apparent. First, finger occlusion renders precise selection difficult. Second, there is an issue called "fat finger problem", may arise; selection on a touchscreen is difficult if targets smaller than a fingertip are densely placed. In the context of text selection, the finger then selects not only the desired text, but also unwanted letters next to the text. Finally, a finger cannot be used to reach the upper region of the touchscreen if the device is held in one hand.

In this paper, we show Press & Tilt which is a text control method (we collectively refer to text selection and text command execution as text control) that solves the abovementioned problems. The user selects text and executes a text command with the thumb of the hand that holds the smartphone, as follows: 1) Press a key on the software keyboard; that chooses the text command. 2) Tilt the smartphone with the key pressed; the selection range changes according to the degree of tilt. 3) Release the key; the text command is then executed. Caret navigation is similarly performed. Figure 1 shows caret navigation with text copying. This design resolves the occlusion and fat finger problem because the user can perform text control without the need to touch the upper region of the touchscreen.

2 RELATED WORK

Several methods allowing text control without touching the upper region of the touchscreen have been proposed. Some are general purpose methods that can be used for text control (e.g., MagStick [11], iPhone Reachability [4], and the techniques of Kim et al. [6]). Some feature caret navigation and text selection (e.g., the methods of Scheibel et al. [12], Fix and Slide [14], and iPhone 3D Touch [5]). Also, text commands can be executed using Gestures and Widgets [2] and PalmTouch [7]. In contrast to these methods, we use tilting to resolve the perceived problems.

Tilting has often been used to operate devices (e.g., [1, 3, 8, 9, 16–18]). Among them, some researches (e.g., [13, 18]) use tilting to avoid occlusion and the fat finger problem because tilting does not require use of the touchscreen. We use tilting for the same purpose as these researches.

3 PILOT STUDY: AN INVESTIGATION OF HOW USERS TILT A SMARTPHONE

We investigated how users tilted smartphones. We measured smartphone *Roll*, *Pitch*, and *Yaw* (Figure 2) on tilting in four directions: *Left*, *Right*, *Up*, and *Down*.

3.1 Setup

12 volunteers (nine males and three females) aged 21–24 years (M = 22.7 years) participated in the study. In daily use, they use a smartphone with their right hand. The experimenter measured the length from the top of the middle finger to the base of the palm (hand length) using a tape measure; the hand length ranged from 16.3–20.1 cm (M = 18.4 cm).

We used an Xperia XZ (dimensions: 46 × 72 × 8.1 mm, display size: 5.2 inches, OS: Android 8.0.0) as the smartphone in this study.

3.2 Task

We asked the participants to sit on a chair and to hold the smartphone in the right hand in the usual manner (base posture). We instructed them to tilt the smartphone to the *Left*, *Right*, *Up*, and *Down* in this order from the base posture. In order to eliminate discrepancies between the instructed directions and the directions perceived by the participants, the tilt directions were demonstrated by the experimenter beforehand. Moreover, we asked the participants to tilt the smartphone within the range where the participants can see the text and caret displayed on the touchscreen since the user needs to watch the touchscreen for text control. Furthermore, we asked them to tilt the smartphone while pressing a key (specifically the *m* key) for simulating text control and to release the key after tilting the smartphone to the maximum degree within the range. We recorded smartphone *Roll*, *Pitch*, and *Yaw* as the *m* key was pressed and released.

Tilting in one direction was termed a trial and tilting in all four directions a session. Each participant engaged in five sessions. We thus collected data on 240 (4 directions × 5 sessions × 12 participants) trials; each participant required about 15 minutes to perform all sessions.

3.3 Results

Table 1 shows the *Roll*, *Pitch*, and *Yaw* data when the *m* key was released. In the following discussion, we focus only on data from the horizontal tilting (i.e., *Left* and *Right*) because our method employs such tilting only. These results show that the amount of change in *Roll* when the user tilts the smartphone to *Left* is larger than that of *Right*. This would be because the user can watch the touchscreen even if the user tilts the smartphone largely to *Left* since the head is positioned on the left of the smartphone if the user holds the smartphone with the right hand; by contrast, the user soon cannot watch the touchscreen when the user does so to *Right*.

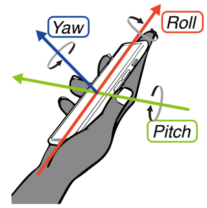

Figure 2: Smartphone tilt axes.

Direction	*Roll*	*Pitch*	*Yaw*
Left	−61.34°	−24.80°	25.30°
Right	56.78°	−22.19°	15.35°
Up	121.85°	2.39°	34.06°
Down	−10.33°	−31.81°	9.53°
Base	13.03°	67.60°	−37.44°

Table 1: Smartphone *Roll*, *Pitch*, and *Yaw* at maximal tilts.

Furthermore, we observed not only *Roll* but also *Pitch* and *Yaw* change when a user tilts a smartphone in the horizontal directions. Figure 3 shows the relationships between *Roll* and *Pitch*, and *Roll* and *Yaw*. Both relationships were similar for all participants except P12, whose *Roll–Pitch* curve differed markedly from these of other participants. P12 may have a distinctive habit of tilting or because P12's hand length was the smallest (16.3 cm).

4 DESIGN AND IMPLEMENTATION

By using the results of the pilot study, we designed and implemented Press & Tilt.

4.1 Tilt Operation

In our method, the text starting at the initial position (i.e., the caret position when the *c* key is pressed) with the length of *N* letters is selected, where *N* is determined by the degree of tilt. It is possible to simply calculate *N* using a position decision method based on a trigonometric function such as that of [13]. However, the ease of text selection changes depending on letters with this method

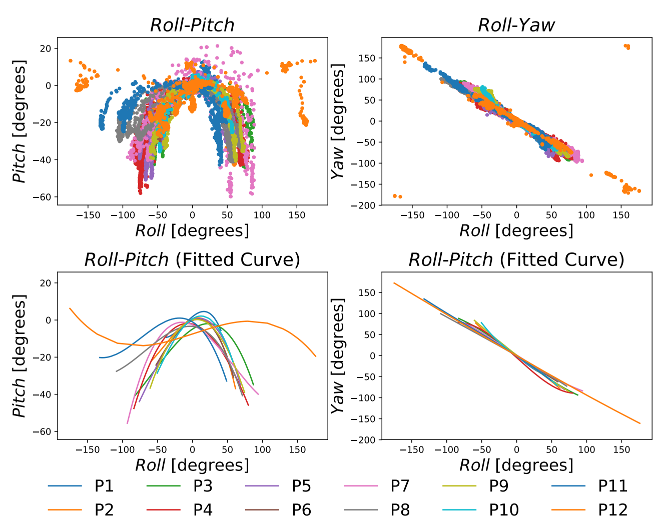

Figure 3: Relationships among the degrees of tilt.

Key	Text Command	Key	Text Command
c	Copy	*v*	Paste
Shift	Convert upper/lower	*b*	Internet search
s	Word search	*l*	Launch application
m	Caret navigation	*t*	Translation

Table 2: Example assignment.

because alphabet widths differ. Thus, we designed a position decision method allowing all letters to be easily selected. Also, we considered the differences between *Left* and *Right*, because there are differences of degrees of tilt between *Left* and *Right* in the result of the pilot study. Since our current implementation uses tilting only in the horizontal directions, our method calculates N by the following steps: (1) determine the tilting direction (*Left* or *Right*) using *Roll* and then (2) obtain N by the equation below:

$$N = \alpha \times \sqrt{(nR/mR)^2 + (nP/mP)^2 + (nY/mY)^2},$$

where, nR, nP, and nY are the current *Roll*, *Pitch*, and *Yaw*, respectively; and mR, mP, and mY are the maximum angles recorded in the pilot study. If the tilt direction is *Left*, the figures in the *Left* column of Table 1 are used. α is a constant and represents the ratio of N to the degree of tilt (currently 20). Moreover, we used *Pitch* and *Yaw* as well as *Roll* because *Pitch* and *Yaw* change according to *Roll* in the pilot study.

4.2 Text Commands

After pressing a key and tilting the smartphone, the text command assigned to the key is executed using the selected text; if the key is released without tilting, the letter is input. Table 2 shows an example assignment of text commands to keys; we sought to allow the user to memorize the assignment (e.g., we assigned copy command to the *c* key). It is possible to assign many text commands to keys in a manner similar to keyboard shortcuts on desktop/laptop computers.

5 USER STUDY: COMPARISON WITH BUILT-IN METHOD

We conducted a user study to compare Press & Tilt with the built-in method in terms of time and usability.

5.1 Setup

We recruited six graduate students (P1–P6: four males and two females) as participants; the range of age was 22–24 years (M = 23.3 years), four of them (P1–P3, P5) had participated in the pilot study. All participants were right-handed. Hand length ranged between 16.3–19.2 cm (M = 18.3 cm). We used the smartphone employed in the pilot study.

5.2 Task

We asked all participants to sit on a chair and hold the smartphone as they usually use the smartphone. A task is to perform a text command on a blue-colored text (target text) in the whole texts. A text command was performed with the following two types of method.

Press & Tilt: To determine the start position of the selection, participants moved the caret to just before the target text by tilting the smartphone while pressing the *m* key. Next, they performed text selection by tilting the smartphone while pressing any key except the *m* key, and then executed the text command on the target text by releasing the key.

Built-In Method: The participants initiate text selection by pressing the text for a few moments and then moving adjusters to select range. They then execute the text command on the target text by tapping *Next* on the menu that appeared near the text (*Next* was the only command on the menu).

A trial was to correctly perform text control (i.e., text selection and command execution) once; if the text command was correctly executed, the target text was updated, but otherwise not; the participant adjusted the selection range and executed the command again. A session featured eleven trials. Data collection began after the first text control was correctly performed; therefore the time required for trials 2–11 (10 trials) were recorded. All participants completed 10 sessions using each method. In total, we collected data on 1200 (6 participants × 10 trials × 10 sessions × 2 methods) trials. To reduce any effect of fatigue, all participants rested for at least one minutes between sessions. To eliminate any effect of order, we divided all participants into two groups: one used our method first and the other used the built-in method. To assess the participants' impression, we used the System Usability Scale (SUS) that was filled after each method was completed.

We used randomly generated English sentences[1]; we composed the text used in the study by randomly choosing sentences among the sentences to avoid scrolling the text. We used the same text when evaluating both methods. In each trial, the target text was a text substring: we randomly chose a sentence and then a substring thereof with a length greater than or equal to 1; if the first or last letter was blank, we repeated the procedure.

This study required approximately 75 minutes per participant. All participants were paid JPY 1640 (approximately USD 14.8) for their time.

5.3 Results

All participants performed all tasks using only their right hands. Figure 4 shows the task completion times (both methods) for each session in the form of fitted curves [10]. As we did not provide training, the times taken to apply either method fell as the session number increased. We divided the 10 sessions into five groups (1–2, 3–4, 5–6, 7–8, and 9–10 sessions) and compared the times required to implement either method using the paired t–test. As a result, there were significant differences between the first (p = .016) and third (p = .022) groups; no significant differences in the second (p = .072), fourth (p = .215), and fifth (p = .072) groups. From this result, the difference between the methods would decrease as the user becomes accustomed to our method. When we extrapolated the fitted curves, we found that they should cross at approximately session 56, although this requires experimental validation.

Figure 5 shows times required by each participant. All participants became faster with experience, except P1 (P12 of pilot study; that participant also differed from others in the pilot study). Upon

[1] http://randomtextgenerator.com/

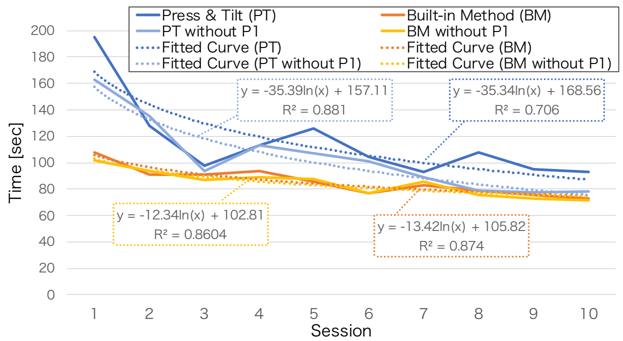

Figure 4: Times of the two methods in each session.

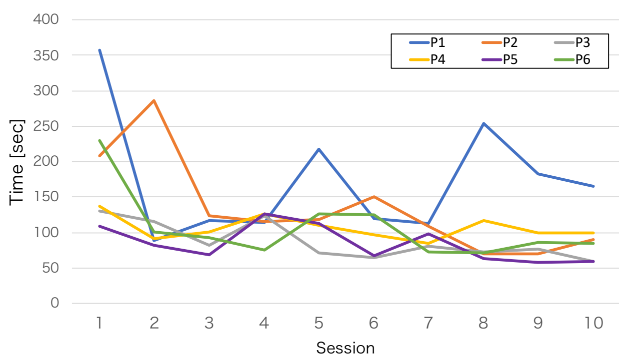

Figure 5: Time per participant in our method.

analysis of all data except those of P1, the two fitted curves cross at approximately session 13; our method was thus appropriate for most participants. From this result, it is necessary to improve implementation for users such as P1. One possibility is to use only *Roll* to determine the selection range because *Pitch* and *Yaw* were constituted noise for P1 (P12, Figure 3). Moreover, conducting the pilot study with more participants is necessary for further investigation into the relations between the axes.

The SUS scores of our method and the built-in method were 61.7 and 55.8, respectively. Although the difference is not significant, we recorded some positive comments: "I felt that fine adjustment was very difficult with your method. I think that your method will be easier to handle than the built-in method if you can improve the adjustment", "It took a long time to get used to, but it was easy to move the cursor quickly". Thus, it is possible that our method can become better than the built-in method.

6 DISCUSSION AND FUTURE WORK

Influence of Fonts: Text selection with both Press & Tilt and all other methods is influenced by font size and width. Therefore, user studies using some font sizes as in [12, 14] are necessary for investigating the effects of font size. The effects of font width also need to be investigated, especially in the case of a monospace font. In this case, the pointing methods shown in [13, 15] could work better than our current implementation that considers font width. For this reason, we plan to implement such methods to find the appropriate method for each condition.

User Situations and Attributes: Smartphones are used not only while sitting, but also while walking and when supine, creating unexpected behavior. For example, our method does not work when the user lies on the back because the tilt directions now differ. Also, our method will not work in elevators or when boarding an airplane. User posture and situation require further study. For example, we will extend our method to various scenarios including walking with a bag in one hand, or when a hand strap is held on a train. Furthermore, we plan to change our implementation to dynamically change the parameters (e.g., ignores one or two tilt axes in a specific situation) by recognizing postures and situations using the readings of inertial measurement unit (IMU). Moreover, handedness is another issue; we conducted our user studies with users who use a smartphone with their right hand in daily use. Since the results with users who use a smartphone with their left hand would be different, we plan to investigate the difference between right-handed and left-handed users.

Future Improvements: In the user study, we collected some comments about caret navigation such as "I often missed one letter" and "It moved too much". Therefore, optimization is required. Moreover, some comments about text selection, such as "I want to not select 'space' " and "I want to change the selection range on a word-to-word basis", suggests the necessity for improving our method for the efficient text selection including space and a word-to-word basis selection. Furthermore, since our user study focused only text-selection time; we will in future evaluate error rates and the number of operations required.

Occlusion: Our method minimizes occlusion caused by touching the text, but it is difficult to completely eliminate occlusion; specifically, our method requires that the keyboard be displayed; this occludes text on the lower region of the touchscreen. Also, the finger occludes the keys. It is difficult to solve such touchscreen problems; we will explore whether side buttons can be employed instead.

7 CONCLUSIONS

We showed a one-handed text control method that uses tilting called Press & Tilt. A user can perform caret navigation or text selection by tilting the smartphone while pressing a key. When the key is released, a text command is executed. In our method, text control does not require touching of the upper region of the smartphone; neither occlusion nor the fat finger problem is of concern. We performed a pilot study to explore how users tilted smartphones. When tilting horizontally, only *Roll* changed symmetrically in the *Left* and *Right* directions; *Pitch* and *Yaw* also changed. We designed and implemented our method using the results of the pilot study. We conducted a user study to compare our method with a built-in method in operating time and usability. Initially, the built-in method was faster, but the time taken to apply our method fell with practice; it is possible that further practice will render our method faster than the built-in method.

REFERENCES

[1] Yuan Du, Haoyi Ren, Gang Pan, and Shjian Li. 2011. Tilt & Touch: Mobile Phone for 3D Interaction. In *Proceedings of the 13th International Conference on Ubiquitous Computing (UbiComp '11)*. ACM, New York, NY, USA, 485–486. https://doi.org/10.1145/2030112.2030183

[2] Vittorio Fuccella, Poika Isokoski, and Benoit Martin. 2013. Gestures and Widgets: Performance in Text Editing on Multi-touch Capable Mobile Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 2785–2794. https://doi.org/10.1145/2470654.2481385

[3] Guillaume Homberg, Alassane Sangott, and Benoît Martin. 2010. KeyTilt: Un Clavier Logiciel Par Inclinaison. In *Proceedings of the 22nd Conference on L'Interaction Homme-Machine (IHM '10)*. ACM, New York, NY, USA, 137–140. https://doi.org/10.1145/1941007.1941030

[4] Apple Inc. 2014. iPhone User Guide - Reachability. https://help.apple.com/iphone/11/?lang=en#/iph66e10a71c (accessed 2018-8-17).

[5] Apple Inc. 2015. 3D Touch - iOS - Apple Developer. https://developer.apple.com/ios/3d-touch/ (accessed 2018-8-17).

[6] Sunjun Kim, Jihyun Yu, and Geehyuk Lee. 2012. Interaction Techniques for Unreachable Objects on the Touchscreen. In *Proceedings of the 24th Australian Computer-Human Interaction Conference (OzCHI '12)*. ACM, New York, NY, USA, 295–298. https://doi.org/10.1145/2414536.2414585

[7] Huy Viet Le, Thomas Kosch, Patrick Bader, Sven Mayer, and Niels Henze. 2018. PalmTouch: Using the Palm As an Additional Input Modality on Commodity Smartphones. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 360, 13 pages. https://doi.org/10.1145/3173574.3173934

[8] Kurt Partridge, Saurav Chatterjee, Vibha Sazawal, Gaetano Borriello, and Roy Want. 2002. TiltType: Accelerometer-supported Text Entry for Very Small Devices. In *Proceedings of the 15th Annual ACM Symposium on User Interface Software and Technology (UIST '02)*. ACM, New York, NY, USA, 201–204. https://doi.org/10.1145/571985.572013

[9] Jun Rekimoto. 1996. Tilting Operations for Small Screen Interfaces. In *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology (UIST '96)*. ACM, New York, NY, USA, 167–168. https://doi.org/10.1145/237091.237115

[10] Seibel Robert. 1963. Discrimination Reaction Time for a 1,023-alternative Task. *Journal of Experimental Psychology* 66, 3 (1963), 215–226.

[11] Anne Roudaut, Stéphane Huot, and Eric Lecolinet. 2008. TapTap and MagStick: Improving One-handed Target Acquisition on Small Touch-screens. In *Proceedings of the Working Conference on Advanced Visual Interfaces (AVI '08)*. ACM, New York, NY, USA, 146–153. https://doi.org/10.1145/1385569.1385594

[12] Jean-Baptiste Scheibel, Cyril Pierson, Benoît Martin, Nathan Godard, Vittorio Fuccella, and Poika Isokoski. 2013. Virtual Stick in Caret Positioning on Touch Screens. In *Proceedings of the 25th Conference on L'Interaction Homme-Machine (IHM '13)*. ACM, New York, NY, USA, Article 107, 8 pages. https://doi.org/10.1145/2534903.2534918

[13] Keigo Shima, Kazusa Onishi, Ryosuke Takada, Takuya Adachi, Buntarou Shizuki, and Jiro Tanaka. 2016. Investigating Accuracy of Tilting Operation on Wrist-worn Devices with Touchscreens. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '16)*. ACM, New York, NY, USA, 2705–2711. https://doi.org/10.1145/2851581.2892377

[14] Kenji Suzuki, Kazumasa Okabe, Ryuuki Sakamoto, and Daisuke Sakamoto. 2016. Fix and Slide: Caret Navigation with Movable Background. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '16)*. ACM, New York, NY, USA, 478–482. https://doi.org/10.1145/2935334.2935357

[15] Robert J. Teather and I. Scott MacKenzie. 2014. Position vs. Velocity Control for Tilt-based Interaction. In *Proceedings of Graphics Interface 2014 (GI '14)*. Canadian Information Processing Society, 51–58. http://dl.acm.org/citation.cfm?id=2619648.2619658

[16] Theophanis Tsandilas, Caroline Appert, Anastasia Bezerianos, and David Bonnet. 2014. Coordination of Tilt and Touch in One- and Two-handed Use. In *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 2001–2004. https://doi.org/10.1145/2556288.2557088

[17] Daniel Wigdor and Ravin Balakrishnan. 2003. TiltText: Using Tilt for Text Input to Mobile Phones. In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology (UIST '03)*. ACM, New York, NY, USA, 81–90. https://doi.org/10.1145/964696.964705

[18] Hui-Shyong Yeo, Xiao-Shen Phang, Steven J. Castellucci, Per Ola Kristensson, and Aaron Quigley. 2017. Investigating Tilt-based Gesture Keyboard Entry for Single-Handed Text Entry on Large Devices. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 4194–4202. https://doi.org/10.1145/3025453.3025520