Back Keyboard: A Physical Keyboard on Backside of Mobile Phone using QWERTY

Hwan Kim
Dept. of Industrial Design,
KAIST.
291 Daehak-ro, Yuseong-gu,
Daejeon 305-701, Korea
iamhwankim@gmail.com

Yeakyung Row
Dept. of Industrial Design,
KAIST.
291 Daehak-ro, Yuseong-gu,
Daejeon 305-701, Korea
applemint87@gmail.com

Geehyuk Lee
HCI@lab
KAIST
291 Daehak-ro, Yuseong-gu,
Daejeon 305-701, Korea
geehyuk@gmail.com

CHI'12, May 5–10, 2012, Austin, Texas, USA.
ACM 978-1-4503-1016-1/12/05.

Abstract
Since smart phones adopted touchscreen, users have been enjoying large displays. However, when using soft keyboard, the available size of the display becomes less than 50%. In this paper *Back Keyboard,* a physical keyboard installed backside of mobile phone, is presented. Also the design process with a prototype through a series of studies is described. User evaluation was conducted with the prototype; the average text entry rate was 15.3 WPM (SD: 3.6) and the error rate was 12.2% (SD: 9.0) after a 40-minute typing session. Moreover, the text entry rates of *Back Keyboard* and general keyboards for PCs did not have significant relations. This means that the prototype could be used smoothly regardless of one's ability of typing on a PC.

Keywords
Text entry, keyboard, backside interaction, handheld devices, mobile devices

ACM Classification Keywords
H.5.2 [Information interfaces and presentation] : User Interface - Input devices and strategies

General Terms
Design, Experimentation, Human Factors, Smart Phone, Mobile Phone

Figure 1. Concept image of *Back Keyboard*. A physical keyboard is on backside of mobile phone.

Introduction

Recently, mobile phones with soft keyboard on full touch screen have been popularly used. Although the soft keyboard allows larger display while reducing the size of the mobile devices, significant problems have come up: users cannot use full size of display since the soft keyboard occupies more than 50% of the screen. People make errors more frequently due to small size of key buttons, condensed layout, and no tactile feedback from the soft keyboard.

For similar problems described above, Frank Chun Yat Li et al. suggested the *1Line keyboard* which uses word disambiguration to reduce the size of *iPad*'s soft keyboard while keeping the QWERTY layout [2]. However Bi et al. studied that users usually tend to refrain from learning reduced or rearranged keyboards but wish to use QWERTY layout they are already familiar with [6]. Therefore, we made an attempt to resolve the soft keyboard problem in a different direction. Although the starting concept was similar to 1Line Keyboard, differentiation was made through size of device and interaction style.

Daniel Wigdor et al. and Patrick Baudisch et al. suggested the possibility of backside interactions using the back of devices [1, 5]. Although they used *pseudo-transparent* screen, it became our starting point of *Back Keyboard* (Figure 1). One of our challenges was to verify whether users are able to use *Back Keyboard,* which is a physical QWERTY keyboard installed on backside of devices as small as *iPhone*.

The physical keyboard on the backside of mobile devices is not a totally novel concept. *RearType* is a physical keyboard applied to the backside of *iPad*-size mobile devices [4]. However, a big difference exists from this project in terms of the size. All 10 fingers can be used in *RearType*, but only two or four fingers are applicable for *Back Keyboard*. Since users' view is occluded with *Back Keyboard*, users have to transfer their spatial memory of the QWERTY keyboard to the backside of mobile phone. Thus, a critical issue in designing *Back Keyboard* is to verify how smoothly the user's spatial memory of the QWERTY layout helps to build the mental model of *Back Keyboard*.

This paper illustrates the process of designing *Back Keyboard* and the results of user evaluation The research questions are defined as follows:

- What is an appropriate form (layout, size, position and etc.) of *Back Keyboard*?
- Does the user's spatial memory of QWERTY layout transfer smoothly to the suggested form?
- Is the user able to use *Back Keyboard* regardless of their proficiency in using QWERTY keyboard?

Design Process

1. Form factor design

The appropriate form factor was decided from the survey and the workshop. A survey was conducted to learn which mode is mainly used between portrait mode and landscape mode. This survey was done with 25 smart phone users (13 male, mean of age: 25.6). Above 90% of the replies show that the portrait mode is used more dominantly in applications related text entry such as SMS. Thus it was decided to design *Back Keyboard* in portrait mode.

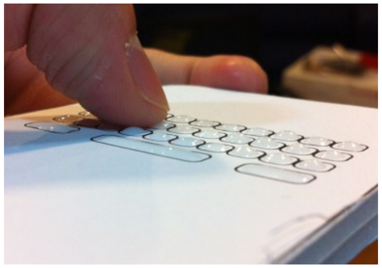

Figure 2. Each button can be distinguished by tactile sensation.

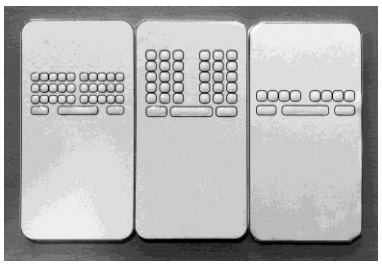

Figure 3. Paper prototypes for the second workshop.
Split but not rotated (left), Split and rotated (center), Reduced like *1Line Keyboard* [2] (right)

Figure 4. A participant is using *Invisible keyboard* with *Nexus One* downside up

After the survey, 6 participants were recruited for a short workshop to determine the appropriate form factor of *Back Keyboard*. The workshop was conducted for 30 minutes with paper prototypes (Figure 2, 3). The paper prototypes were made in the same size of *iPhone* with various types of keyboard layouts. Participants tried these paper prototypes and gave feedback during and after the workshop.

Important findings of these workshops are listed below:

- Vertical position of keyboard should be placed on the upper half the mobile phone.
- The QWERTY keyboard should be split in half as spatial cue for easy recognition of the location of the keys
- The most preferred type was 'Split but not rotated QWERTY layout keyboard' (left of Figure 3. 'Split and rotated keyboard' was considered more difficult to search the location of the keys with fingers.
-

2. *Back Keyboard* layout design

How well the user's spatial memory of positions of all alphabet keys in QWERTY layout transfer from a normal keyboard to backside of mobile phone was examined.

Participant and Method

10 participants who have type speed over 300 WPM and are familiar with QWERTY layout (All right-handed and 7 male, mean of age: 23.1) were recruited.

An application called *Invisible Keyboard* was made for this study. It was developed on Android OS ver.2.1 and installed on *Nexus One*. The *Invisible keyboard* generates twenty phrases randomly selected from MacKenzie & Soukoreff set in advance [3]. Participants were asked to type the phrases as accurately as possible with the *Nexus One* downside up (Figure 4). The application logged the coordinates of X-Y positions where the fingers touched.

Results and Analysis

All participants were able to type the test phrases on the backside of device without extra practice. They used their own spatial memory of QWERTY layout. The average location of each alphabet represented by 10 participants is shown in Figure 5.

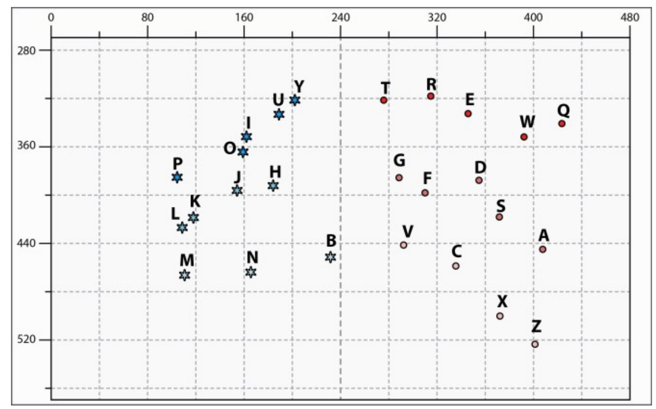

Figure 5 Participants' mental layout of a QWERTY keyboard on backside of mobile device. (View from the back of device) Outlier with value of more than 3 times of SD was removed from the calculation.

Significance test between two alphabets was conducted via post-hoc (Turkey HSD) of MANOVA. Confusing pairs were defined which X-positions and Y-positions were not significantly different from post-hoc test ($p>.05$). The two alphabets of confusing pair are hard to distinguish the positions of each other. There were 28 confusing pairs. This was 8.6% of all possible pairs.

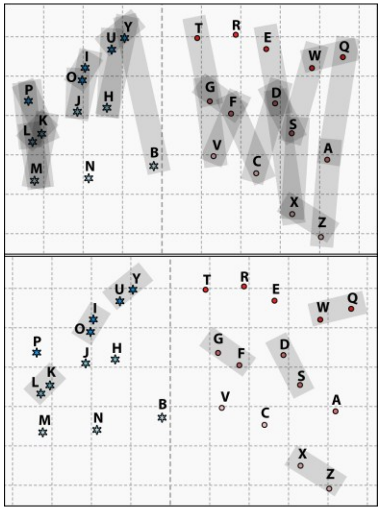

Figure 6. The highlighted alphabets in grey are confusing pairs. Upper image presents all confusing alphabets and lower image contains confusing alphabets in a same row.

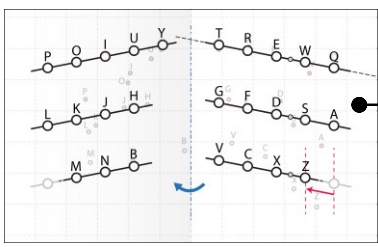

Figure 8. Layout of *Back Keyboard*

Referring to Figure 6, it was hard to distinguish alphabets in columns than rows. The number of confusing alphabet pairs in columns was 21, but only 7 in rows.

In summary, the transfer from the QWERTY layout in human's mental memory to backside of a mobile device was smoothly carried out except for 28 pairs. If spatial cue of the layouts are provided, the number of confusing pairs will decrease.

Prototype of *Back Keyboard*

A prototype of *Back Keyboard* (Figure 7) was designed taking the above findings in consideration. Size of the prototype was similar to *iPhone 4*.

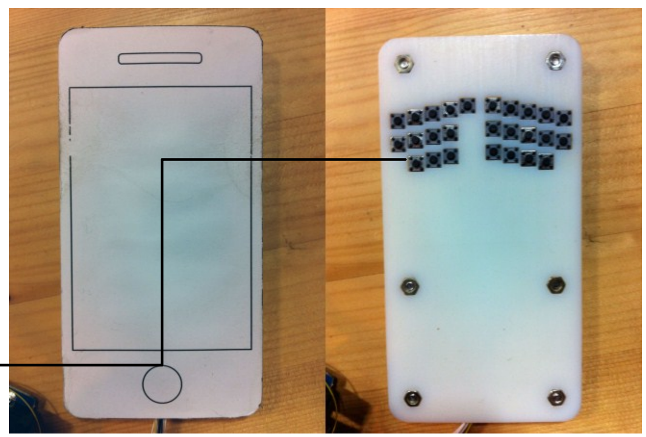

Figure 7. The prototype of *Back Keyboard*.

Prototype of Back Keyboard

The size of the designed prototype is: width 62mm, height 120mm, and depth 10mm. The prototype has only alphabet buttons, no space bar or backspace. The buttons are made with tact switches.

Layout of QWERTY

The QWERTY layout (Figure 8) was arranged based on Figure 5. After drawing a trend line of 'QWERT' row, positions of the 5 alphabets were set on the trend line in equal intervals. This was repeated in the middle and lower rows. However, the last column in the lower row was left out in order to help the users easily distinguish between the three rows. The layout is mirrored on the opposite side. There is a gap between the left and right sides as another spatial cue.

User Evaluation

User Evaluation was conducted with 10 participants to measure WPM and Error rate of the prototype and to verify whether the prototype requires users' typing abilities to general keyboard for a PC.

Participants

10 smart phone users were recruited (All right-handed, 4 male, mean of age: 20.9).

Materials

The prototype was connected to a PC through an I/O board. Phrases appeared on a LCD monitor of the PC, which the participants were asked to type with the prototype (Figure 9).

Study process

At first, text entry rate of each participant was logged with a general keyboard for a PC. Then participants were given time to practice *Back Keyboard* for 5 minutes. The main test consisted of 5 repeated typing sessions with 15 test phrases prepared for each session. All phrases were randomly selected from MacKenzie & Soukoreff set [3]. There were short breaking times between typing sessions.

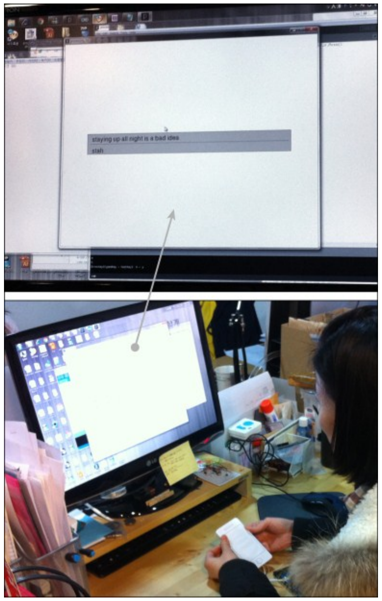

Figure 9. A participant types the phrase presented on LCD monitor with our prototype

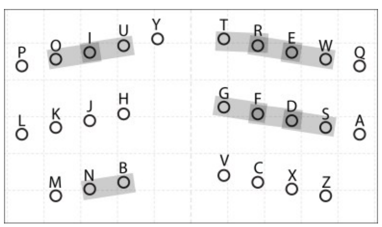

Figure 11. Top 60% errors were generated in a same row.

After all the typing sessions were completed, interviews with participants were conducted in order to get their feedbacks.

Results

1. Overall Performance

The average WPM and Error rate in session 5 were 15.3 WPM (SD: 3.6) and 12.2% (SD: 9.0).

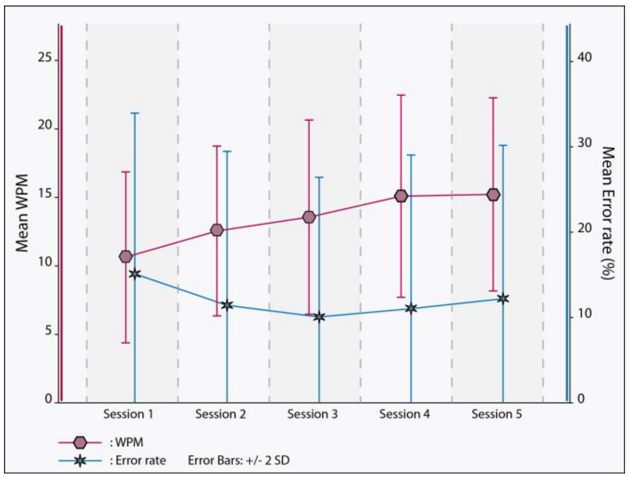

Figure 10. Upper line shows the overall text entry rate (WPM) and lower line shows the overall error rate (%).

The average typing speed was 15.3 WPM (SD: 3.6) in the 5th session. The mean WPM values of 4th session and 5th session were not significantly different ($p<.05$), which means that the learning effect was diminished at 4th session.

Average error rate of session 3 was 10.1%, which was the lowest among five sessions. From session 2 to 5, the mean error rates were not significantly different ($p<.05$).

2. Pattern of Errors

Figure 11 shows the top 60% errors. The top 60% of errors were all pairs in the same row. The spatial cues designed for distinguishing alphabet keys between different rows worked well. Still there was a necessity to find a method to help users to distinguish the different keys in the same row.

3. Comparison among Participants

The participants' typing speeds on a general keyboard for PC were logged, and simple regression analyses were conducted twice; (i) typing speeds of a general keyboard and WPMs, (ii) typing speeds of a general keyboard and error rates. The result of the analyses indicated that the text entry performance of the QWERTY keyboard for the general PC had no significant impact on the WPMs and error rates ($p>.05$). According to this result, user's ability of QWERTY keyboard typing is not necessarily need for using *Back Keyboard*. Consequently, this proves that users can easily acquaint with *Back Keyboard* in a short experience period regardless of their familiarity with keyboard typing.

Conclusion and Further work

The purpose of this study was to introduce the concept of Back Keyboard as a method to keep full screen available when typing text. The prototype was designed to investigate the learning curve and verify how well it can be used regardless of proficiency in using QWERTY keyboards of PCs. It is shown that all participants typed easily after 5 short sessions.

However this research leaves room and necessity for improvements of the performance of the prototype, according to the user evaluation and feedbacks of participants. Five participants said, “I couldn’t type text more quickly because the key buttons (tact switch) are a little bit stiff”. Moreover a *Black Berry* user commented, “It is stiffer to press a button than my phone.” Furthermore, the ways to help users distinguish the key buttons in a same row need to be studied.

In user evaluation session, it is necessary as future work to compare the performance between Back Keyboard and a soft keyboard of a mobile phone of the same size in order to clarify usefulness of Back Keyboard.

The main concept of Back Keyboard is using the backside of mobile phone as another input space, especially text entry. The full size of the touch screen of mobile phone becomes available for input and output when a user types text with Back Keyboard. It is envisioned that this can create a novel interaction between the front and back of device via Back Keyboard. The possible interaction techniques with touch screen of mobile phone and Back Keyboard will be studied in future work.

Acknowledgements
We thank Ju-Whan Kim and Han-Seung Ryu who helped us to implement the prototype.

Reference
[1] Daniel Wigdor, Clifton Forlines, Patrick Baudisch, John Barnwell, and Chia Shen. Lucid touch: a see-through mobile device. In *Proc.* UIST '07. ACM Press, 269-278.

[2] Frank Chun Yat Li, Richard T. Guy, Koji Yatani, and Khai N. Truong. The 1Line keyboard: a QWERTY layout in a single line. In *Proc.* UIST '11. ACM Press (2011), 461-470.

[3] I. Scott MacKenzie and R. William Soukoreff. 2003. Phrase sets for evaluating text entry techniques. Ext. Abstracts CHI '03. ACM Press, 754-755.

[4] James Scott, Shahram Izadi, Leila Sadat Rezai, Dominika Ruszkowski, Xiaojun Bi, and Ravin Balakrishnan. 2010. RearType: text entry using keys on the back of a device. In *Proc.* MobileHCI '10. ACM Press, 171-180.

[5] Patrick Baudisch and Gerry Chu. Back-of-device interaction allows creating very small touch devices. In *Proc.* CHI '09. ACM Press (2009), 1923-1932.

[6] Xiaojun Bi, Barton A. Smith, and Shumin Zhai. Quasi-qwerty soft keyboard optimization. In *Proc.* CHI '10. ACM Press (2010), 283-286.