B2B-Swipe: Swipe Gesture for Rectangular Smartwatches from a Bezel to a Bezel

Yuki Kubo, Buntarou Shizuki, and Jiro Tanaka
University of Tsukuba
Tennodai 1-1-1, Tsukuba, Ibaraki 305-8573, Japan
{kubo,shizuki,jiro}@iplab.cs.tsukuba.ac.jp

ABSTRACT
We present B2B-Swipe, a single-finger swipe gesture for a rectangular smartwatch that starts at a bezel and ends at a bezel to enrich input vocabulary. There are 16 possible B2B-Swipes because a rectangular smartwatch has four bezels. Moreover, B2B-Swipe can be implemented with a single-touch screen with no additional hardware. Our study shows that B2B-Swipe can co-exist with Bezel Swipe and Flick, with an error rate of 3.7% under the sighted condition and 8.0% under the eyes-free condition. Furthermore, B2B-Swipe is potentially accurate (i.e., the error rates were 0% and 0.6% under the sighted and eyes-free conditions) if the system uses only B2B-Swipes for touch gestures.

Author Keywords
Ultra-small devices; Bezel Swipe; eyes-free input; crossing; double-crossing; touch screen; touch gesture; wearable.

ACM Classification Keywords
H.5.2 [Information Interfaces and Presentation]: User Interfaces – Input devices and strategies, Interaction styles.

INTRODUCTION
Touch interactions on smartwatches are severely limited compared with those on tablets and smartphones because the touch screen is so small (e.g., 1.6 inches on SONY SmartWatch 3), making it difficult for the user to perform touch gestures with multiple fingers. Therefore, touch gestures on a smartwatch are essentially limited to gestures with a single finger.

To solve this problem, recent products provide input methods using other modalities, such as wrist shake gestures and voice input, although these methods cannot be used in some environments (e.g., libraries and crowded places) because they are obtrusive. In the HCI field, a trend in increasing the input vocabulary of smartwatches has been to add sensors, such as infrared sensors [11, 14, 9], touch sensors [19, 16], a magnetometer [6], joysticks [20], and a camera [5]. For example, Xiao et al. [20] used joystick sensors to use a watch face as a mechanical interface. iSkin by Weigel et al. [19] is a flexible and stretchable touch sensor that can be used to form a keyboard for smartwatches.

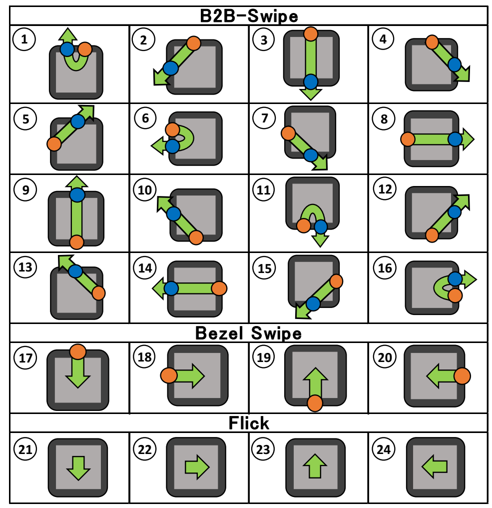

Table 1. B2B-Swipes (#1 – #16) and the touch gestures that we used in our experiment (#17 – #24). The orange and blue circles show the start and end bezels for the gestures, respectively.

In this paper, we present B2B-Swipe, a single-finger swipe gesture for a rectangular smartwatch that starts at a bezel and ends at a bezel. There are 16 possible B2B-Swipes because a rectangular smartwatch has four bezels. B2B-Swipe can be implemented using a single-touch screen with no additional hardware.

B2B-SWIPE
A B2B-Swipe is a swipe gesture that starts from a bezel (start bezel) and ends at a bezel (end bezel) of a smartwatch with a rectangular touch screen, as shown in Table 1. To perform a B2B-Swipe, the user lets a finger cross the start bezel, i.e., move from outside the touchscreen on to the touch screen. The user then lets the finger move across the touchscreen and cross the end bezel moving towards the outside.

A rectangular touch screen has four bezels, so there are 16 possible B2B-Swipes (i.e., the gestures #1 – #16 in Table 1). Note that B2B-Swipes can co-exist with other touch gestures, such as Bezel Swipes, Flicks, and Taps, because the system can detect a B2B-Swipe simply by examining whether a

CHI'16, May 07-12, 2016, San Jose, CA, USA
© 2016 ACM. ISBN 978-1-4503-3362-7/16/05$15.00
DOI: http://dx.doi.org/10.1145/2858036.2858216

stroke starts by crossing a bezel and ends by crossing a bezel. Therefore, B2B-Swipes could greatly contribute to enriching the input vocabulary of smartwatches.

The design of B2B-Swipes leverages the physical characteristics of a rectangular smartwatch in two ways. First, it leverages the tactile feedback that the user obtains by touching the smartwatch: the user can specify one of the four bezels as the start bezel without looking at the smartwatch. This design allows the user to perform a B2B-Swipe in an eyes-free manner. Second, the design takes advantage of the ultra-small displays of smartwatches: the distance between two bezels is short; thus the user can perform the gestures quickly.

In addition, a B2B-Swipe is a kind of double-crossing gesture because it consists of two consecutive crossing gestures. Therefore, the user can perform B2B-Swipes accurately because a double-crossing gesture decreases errors, as shown by Nakamura et al. [12].

RELATED WORK

Our B2B-Swipe can be implemented using only the single-touch screen of smartwatches with no additional sensors. Similarly, Ashbrook et al. [1] explored variously sized buttons placed on the bezel of a circular touch screen smartwatch, which could be implemented with only the single-touch screen. Moreover, Beats [13] used only a touch screen to increase the input vocabulary: they devised touch gestures by combining tapping patterns and release patterns with two fingers on a multi-touch screen. While the Beats' implementation requires a multi-touch screen to recognize tapping with two fingers and the release pattern, B2B-Swipe requires only a single-touch screen, because it is a swipe with a single finger.

Some eyes-free input methods for smartwatches have been proposed [15, 3, 4, 16]. Among them, the most similar to B2B-Swipe were developed by Gábor and Fiener [4], who proposed an input method using a bezel as a tactile landmark for interaction that allows the user to perform eyes-free input. Similarly, B2B-Swipe uses the bezels of a rectangular smartwatch as tactile landmarks so that the user can specify the start bezel without looking at the smartwatch. Some work has used sensors other than the touch screen to realize eyes-free input methods. For example, WatchIt [16] embedded touch sensors within a watchband and allowed the user to perform swiping and tapping on the band; Pasquero et al. [15] used Hall-effect sensors and a force sensor that enable the user to perform various eyes-free gestures, including covering the watch's face and turning the watch's bezel. In contrast, B2B-Swipe uses only a single-touch screen.

Touch gestures utilizing the bezels of touch screens of mobile devices other than smartwatches have been proposed [2, 18, 7, 10, 17, 8]. Among them, Bezel Swipe [17] stands out. It is a touch gesture that starts by crossing a bezel. In contrast, a B2B-Swipe is a double-crossing touch gesture, which starts by crossing a bezel and ends by crossing a bezel. Double-crossing touch gestures are not new; Kurosawa et al. [10] proposed Bezel Check, which starts by crossing a bezel and ends by crossing the same bezel, as a touch gesture to store data in a clipboard placed at the bezel where the gesture was performed. Bezel Check crosses the same bezel of tablets twice. In contrast, B2B-Swipe utilizes the short distances among the four bezels of smartwatches. Bezel Menus [8] explored different bezel-initiated marking menu layouts for eyes-free interaction on small mobile devices. The menus are initiated with Bezel Swipe; thus, B2B-Swipe could co-exist with Bezel Menus.

IMPLEMENTATION

Because the smartwatch that we used in our implementation could not detect touch events on its bezels, we implemented a gesture detector to estimate the start and end bezels, based on the stroke velocity and the position when a stroke is made. To estimate the start bezel, the detector used the timed coordinates of the first and sixth touch events (p_1 and p_6) to calculate the stroke velocity. Based on the velocity, the detector estimated the coordinate of the finger (p_0) 30 ms before p_1 and used the segment between p_0 and p_1 for this estimation. The end bezel was estimated in a similar manner. The detector used the last and second-last touch events (p_n and p_{n-1}) to estimate the position of the finger (p_{n+1}) 40 ms after p_n. The detector used the segment between p_n and p_{n+1} for this estimation. If the detector found both the start and end bezels, the stroke was classified as a B2B-Swipe; if the detector found only the start bezel, the stroke was classified as a Bezel Swipe. Otherwise, the detector used Fling() of the GestureDetector class to examine whether a Flick was performed.

EXPERIMENTAL EVALUATION

We evaluated B2B-Swipe together with Bezel Swipe and Flick to explore its performance and to assess whether it can co-exist with other touch gestures. We chose Bezel Swipe and Flick because both are used widely on smartwatches and are touch gestures that use finger movement on the touch screen, as does B2B-Swipe. Moreover, to investigate users' ability to perform B2B-Swipe without looking at the smartwatch, we evaluated these touch gestures under sighted and eyes-free conditions.

We recruited eight participants (one female; all right handed) ranging in age from 21 to 24 years. For participating, they were paid 1,640 JPY (approximately 13.6 USD). Their experience with mobile devices with touch screens ranged from 29 to 72 months (M = 51.75, SD = 14.8). No participant had ever used a smartwatch.

Design

The participants performed the 24 gestures described in Table 1 under two pose conditions: sighted (Figure 1a) and eyes-free (Figure 1b). The participants were randomly assigned to one of two equal groups to counterbalance the order effect: one group performed these gestures under the sighted condition first and then under the eyes-free condition; the other group performed these gestures in reverse order. Each participant was asked to complete one training session and then four sessions under one condition. In each session, he/she performed the 24 gestures once in a randomized order. In summary, the experiment design involved

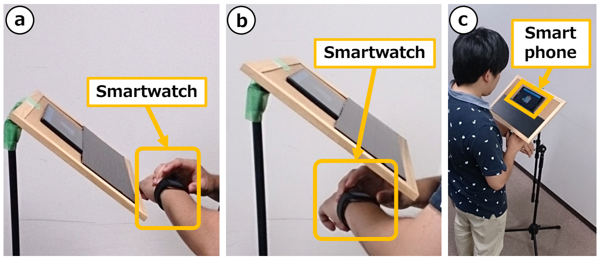

Figure 1. Two conditions used in our experiment and their environments. (a) Under the sighted condition, participants placed their forearms with the smartwatch within their sight. (b) Under the eyes-free condition, they placed their forearms under a board. (c) Participants stood in front of a smartphone.

$2\,conditions \times 8\,participants \times 4\,sessions \times 24\,gestures = 1,536\,trials.$

We asked participants to stand in front of a smartphone, which was used to present instructions, as shown in Figure 1c. We adjusted the height of the smartphone so that each participant could see the screen easily. We then asked the participant to wear the smartwatch (SONY SmartWatch 3 with a 1.6-inch touch screen) on their preferred wrist (all participants chose their left wrists). The participants then read the document with the instructions for the experimental procedure and the set of gestures (Table 1) that the participant had to perform during the experiment. The document emphasized that the gestures had to be performed as accurately as possible. We also asked the participants to think-aloud their errors. Then, the sessions began.

As a trial, each participant performed the gesture specified on the smartphone, on the touch screen of the smartwatch. We asked the participant to do this with the index finger of their dominant hand, with the thumb touching the bottom-left corner of the smartwatch to stabilize their fingers. After a gesture was performed, the smartphone displayed the result as visual feedback to notify the participants whether the gesture had been performed correctly. There was an interval of 1.5 s between trials. Participants took a break of at least 30 s between each of two sessions. We also interviewed the participants about their impressions regarding these gestures after every session. When the participants finished the task, we asked them to complete a questionnaire with six questions that used a five-point Likert scale (1-Strongly disagree, 5-Strongly agree). Each participant completed all of the tasks in approximately 70 min.

Results

We calculated the error rate for each gesture. The rates under the sighted condition were 3.7% (SD = 2.8), 10.2% (SD = 10.0), and 18.8% (SD = 23.1), for B2B-Swipe, Bezel Swipe, and Flick, respectively; those under the eyes-free condition were 8.0% (SD = 4.5) 9.4% (SD = 12.5), and 12.5% (SD = 12.5). The Kruskal-Wallis tests showed that there was no significant difference in the error rates for these gestures under the sighted ($p = 0.232 > 0.05$) and eyes-free ($p = 0.688 > 0.05$) conditions. We also compared the error rates of the three gestures between the sighted and eyes-free conditions. A dependent t-test showed no significant difference for B2B-Swipe ($p = 0.066 > 0.05$). For Bezel Swipe and Flick, we used Wilcoxon signed-rank test because Kolmogorov-Smirnov tests showed non-normality of the Bezel Swipe data under the eyes-free condition ($p = 0.021 < 0.05$) and the Flick data under the sighted condition ($p = 0.032 < 0.05$). The tests showed no significant difference for Bezel Swipe ($p = 0.731 > 0.05$) and Flick ($p = 0.674 > 0.05$).

We calculated the trial time of each gesture. Under the sighted condition, times were 336.3 ms (SD = 193.6), 317.1 ms (SD = 176.8), and 150.7 ms (SD = 66.9) for B2B-Swipe, Bezel Swipe, and Flick, respectively; under the eyes-free condition, they were 316.3 ms (SD = 163.2), 274.7 ms (SD = 155.9), and 158.6 ms (SD = 57.3). Kruskal-Wallis tests showed significant differences under the sighted condition ($p = 0.030 < 0.05$) and no significant difference under the eyes-free condition ($p = 0.088 > 0.05$). Under the sighted condition, post-hoc tests showed a significant difference between B2B-Swipe and Flick ($p = 0.033 < 0.05$): Flick was 185.5 ms faster than B2B-Swipe. We also compared the trial times of the three gestures between the sighted and eyes-free conditions. Dependent t-tests showed a significant difference: Bezel Swipe was 42.4 ms faster under the sighted condition than under the eyes-free condition ($p = 0.019 < 0.05$), with no significant difference for B2B-Swipe ($p = 0.618 > 0.05$) and Flick ($p = 0.252 > 0.05$).

Figure 2 summarizes the questionnaire results. The Kruskal-Wallis test showed no significant differences among the three gestures for Q1 ($p = 0.813 > 0.05$), Q2 ($p = 0.801 > 0.05$), Q3 ($p = 0.392 > 0.05$), Q4 ($p = 0.461 > 0.05$), Q5 ($p = 0.418 > 0.05$), and Q6 ($p = 0.878 > 0.05$).

DISCUSSION

Overall, the results suggest that the performance of B2B-Swipe is competitive with that of Bezel Swipe, as the error rates of B2B-Swipe were low and B2B-Swipe did not differ significantly from Bezel Swipe in either error rates or trial times under both conditions. Furthermore, the performance of B2B-Swipe was roughly the same under the eyes-free condition as under the sighted condition, with no significant differences in the error rates and trial times between the two conditions.

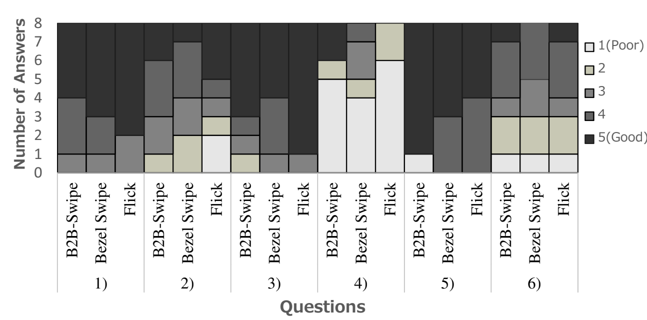

Figure 2. Responses to the following six questionnaire items: 1) the gesture is easy to perform; 2) the gesture requires accuracy; 3) I got used to the gesture quickly; 4) I grew tired of performing the gesture; 5) I would like to have the gesture on my smartwatch; and 6) the gesture is more difficult under the eyes-free gesture condition than under the sighted condition.

(a) Classfied as

Instruction	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	0
1	100	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	90.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1	0.0	6.3
3	0.0	0.0	100	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	96.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1
5	0.0	0.0	0.0	0.0	96.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0
6	0.0	0.0	0.0	0.0	0.0	100	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7	0.0	0.0	0.0	0.0	0.0	0.0	96.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1
8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	96.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1
9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	84.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	12.5	0.0	3.1	0.0	0.0	0.0	0.0
10	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	93.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1	0.0	0.0	0.0	0.0	0.0	3.1
11	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
12	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	93.8	0.0	0.0	0.0	0.0	0.0	0.0	3.1	0.0	0.0	0.0	0.0	3.1	0.0
13	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	96.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1
14	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	96.9	0.0	0.0	0.0	0.0	0.0	3.1	0.0	0.0	0.0	0.0	0.0
15	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
16	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	96.9	0.0	0.0	0.0	3.1	0.0	0.0	0.0	0.0	0.0
17	0.0	0.0	6.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	81.3	0.0	0.0	3.1	0.0	0.0	3.1	0.0	6.3
18	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	87.5	0.0	0.0	0.0	0.0	0.0	3.1	6.3
19	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	90.6	0.0	0.0	0.0	0.0	0.0	6.3
20	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	100	0.0	0.0	0.0	0.0	0.0
21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1	0.0	93.8	0.0	0.0	0.0	0.0
22	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	6.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	93.8	0.0	0.0	0.0
23	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	6.3	0.0	0.0	0.0	0.0	0.0	93.8	0.0	0.0
24	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100	0.0

(b) Classfied as

Instruction	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	0
1	100	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	100	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	87.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	9.4	0.0	0.0	0.0	0.0	0.0	3.1	0.0	0.0
4	0.0	0.0	0.0	93.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1	0.0	0.0	0.0	0.0	0.0	0.0	3.1	0.0
5	0.0	0.0	0.0	0.0	90.6	0.0	0.0	6.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1	0.0
6	0.0	0.0	0.0	0.0	0.0	96.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7	0.0	0.0	0.0	0.0	0.0	0.0	96.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1	0.0	0.0
8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	93.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	6.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0
9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	96.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1
10	0.0	0.0	0.0	0.0	0.0	3.1	0.0	0.0	0.0	96.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
11	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	96.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1	0.0	0.0
12	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	75.0	0.0	0.0	0.0	0.0	0.0	0.0	12.5	0.0	12.5	0.0	0.0	0.0	0.0
13	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	93.8	0.0	0.0	0.0	0.0	0.0	0.0	3.1	3.1	0.0	0.0	0.0	0.0
14	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	87.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	6.3	0.0	0.0	6.3
15	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	84.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	6.3	0.0	6.3
16	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	81.3	0.0	0.0	0.0	9.4	0.0	0.0	0.0	6.3	3.1
17	0.0	0.0	3.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	96.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
18	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	93.8	0.0	0.0	0.0	0.0	0.0	0.0	3.1
19	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	93.8	0.0	3.1	0.0	0.0	0.0	3.1
20	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1	0.0	3.1	0.0	0.0	0.0	78.1	0.0	6.3	0.0	0.0	9.4
21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	96.9	0.0	0.0	0.0	0.0
22	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	9.4	0.0	90.6	0.0	0.0	0.0
23	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100	0.0	0.0
24	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100	0.0

Table 2. Confusion matrices for the (a) sighted and (b) eyes-free conditions. In this table, the gesture identifications are the same as those shown in Table 1, with the exception of #0, when the system classified the participant's gesture as not associated with B2B-Swipe, Bezel Swipe, or Flick.

The results also suggest that B2B-Swipe gestures are potentially accurate if a system uses only B2B-Swipe as touch gestures, because no B2B-Swipe was classified as another B2B-Swipe (i.e., the error rate was 0%) under the sighted condition; three B2B-Swipes were classified as other B2B-Swipes (i.e., the error rate was 0.6% $= 3/(8\times4\times16)$) under the eyes-free condition. Note that the results support the same claim for Bezel Swipe and Flick: one Bezel Swipe was classified as another Bezel Swipe under the sighted condition, and no Bezel Swipe was classified as another Bezel Swipe under the eyes-free condition; no Flick was classified as another Flick under either condition. Therefore, these three gestures would be equally usable touch gestures on smartwatches if they were only used.

Because the error rate of Flick was as high as that of simple gestures, we also examined the log of the think-aloud protocol, along with the system log. This indicated that 4 of the 121 errors were user errors. We also found that many of the 121 errors were system errors resulting from the detector implementation described above; if the speed of a Flick was slow, then the Android operating system, did not call Fling(); consequently, the detector failed to detect 30 Flicks. Therefore, the corrected error rates of Flick under the sighted and eyes-free conditions were 4.7% (SD = 8.7) and 3.1% (SD = 3.3), respectively. The corrected results are shown in Figure 3 and Table 2. The high error rates of Bezel Swipe were also due to the detector: 8 of 25 Bezel Swipe trials were misclassified as B2B-Swipes. As a result, the corrected error rates of Bezel Swipe under the sighted and eyes-free conditions were 7.0% (SD = 9.1) and 6.3% (SD = 8.2), respectively.

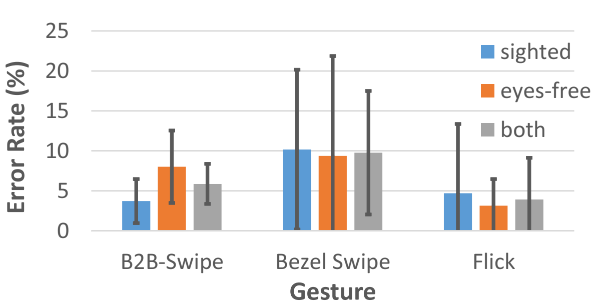

Figure 3. Error rate for each gesture. The grand mean of each gesture is denoted as 'both' in this figure.

Analysis of the confusion matrices described in Table 2 reveals how these three gestures can co-exist. The table shows that a B2B-Swipe classified as a Bezel Swipe tends to share the same start bezel and the same direction, as does a Bezel Swipe that is classified as a B2B-Swipe. For example, 12.5% (4/32) of #9 (upward B2B-Swipe) were classified as #19 (upward Bezel Swipe); and 3.1% (1/32) of #19 were classified as #9 under the sighted condition. Therefore, if a system uses both B2B-Swipe and Bezel Swipe, the system should be designed to assign the same function to B2B-Swipes and Bezel Swipes with the same shapes. Similarly, this design principle should be applied when the system uses both B2B-Swipe and Flick.

In addition, due to the small standard deviations in the error rates, although the participants had never used a smartwatch, B2B-Swipe may be the most stable gesture among the three gestures. Therefore, B2B-Swipe is a touch gesture that anybody can perform.

CONCLUSION & FUTURE WORK

We presented a single-finger swipe gesture for smartwatches that starts from a bezel and ends at a bezel to enrich the input vocabulary of smartwatches. It can be implemented with no additional hardware. Our experiment showed that B2B-Swipe can co-exist Bezel Swipe and Flick, with an error rate of 3.7% under the sighted condition and 8.0% under the eyes-free condition. Moreover, B2B-Swipe is potentially accurate if a system uses only B2B-Swipe for touch gestures.

In the future, we plan to explore a version of B2B-Swipe for circular smartwatches. In addition, memorizing 16 B2B-Swipes is a design issue. Memorization would be facilitated if the four start bezels were assigned to four groups of similar commands. For example, in our music player, we assigned the right start bezel to commands related to volume (e.g., we assigned right→top B2B-Swipe to increase volume and right→down B2B-Swipe to lower volume). Therefore, we plan to implement real B2B-Swipe applications such as map, alarm, and menu control to evaluate its usability, including the memorability of B2B-Swipes in real applications.

REFERENCES

1. Daniel Ashbrook, Kent Lyons, and Thad Starner. 2008. An Investigation into Round Touchscreen Wristwatch Interaction. In *Proceedings of the 10th International Conference on Human Computer Interaction with Mobile Devices and Services (MobileHCI '08)*. ACM, 311–314.
2. Chen Chen, Simon T. Perrault, Shengdong Zhao, and Wei Tsang Ooi. 2014b. BezelCopy: An Efficient Cross-application Copy-paste Technique for Touchscreen Smartphones. In *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces (AVI '14)*. ACM, 185–192.
3. Xiang 'Anthony' Chen, Tovi Grossman, and George Fitzmaurice. 2014a. Swipeboard: A Text Entry Technique for Ultra-small Interfaces That Supports Novice to Expert Transitions. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, 615–620.
4. Blaskó Gábor and Steven Feiner. 2004. An interaction system for watch computers using tactile guidance and bidirectional segmented strokes. In *Proceedings of the 8th IEEE International Symposium on Wearable Computers (ISWC '04)*. IEEE, 120–123.
5. Jaehyun Han, Sunggeun Ahn, and Geehyuk Lee. 2015. Transture: Continuing a Touch Gesture on a Small Screen into the Air. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '15)*. ACM, 1295–1300.
6. Chris Harrison and Scott E. Hudson. 2009. Abracadabra: Wireless, High-precision, and Unpowered Finger Input for Very Small Mobile Devices. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology (UIST '09)*. ACM, 121–124.
7. Ken Hinckley, Jeff Pierce, Mike Sinclair, and Eric Horvitz. 2000. Sensing Techniques for Mobile Interaction. In *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology (UIST '00)*. ACM, 91–100.
8. Mohit Jain and Ravin Balakrishnan. 2012. User Learning and Performance with Bezel Menus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, 2221–2230.
9. Jungsoo Kim, Jiasheng He, Kent Lyons, and Thad Starner. 2007. The Gesture Watch: A Wireless Contact-free Gesture Based Wrist Interface. In *Proceedings of the 11th IEEE International Symposium on Wearable Computers (ISWC '07)*. IEEE, 1–8.
10. Toshifumi Kurosawa, Buntarou Shizuki, and Jiro Tanaka. 2015. Spatial Arrangement of Data and Commands at Bezels of Mobile Touchscreen Devices. In *Proceedings of the 17th International Conference on Human-Computer Interaction (HCI International 2015)*, Vol. 2. Springer, 227–237.
11. Gierad Laput, Robert Xiao, Xiang 'Anthony' Chen, Scott E. Hudson, and Chris Harrison. 2014. Skin Buttons: Cheap, Small, Low-powered and Clickable Fixed-icon Laser Projectors. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, 389–394.
12. Takashi Nakamura, Shin Takahashi, and Jiro Tanaka. 2013. An object selection tecnique using hand gesture in large display -proposing double-crossing and comparing with other techniques. *IEICE Transactions* J96-D, 4 (2013), 978–988. (In Japanese).
13. Ian Oakley, DoYoung Lee, MD. Rasel Islam, and Augusto Esteves. 2015. Beats: Tapping Gestures for Smart Watches. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, 1237–1246.
14. Masa Ogata and Michita Imai. 2015. SkinWatch: Skin Gesture Interaction for Smart Watch. In *Proceedings of the 6th Augmented Human International Conference (AH '15)*. ACM, 21–24.
15. Jerome Pasquero, Scott J. Stobbe, and Noel Stonehouse. 2011. A Haptic Wristwatch for Eyes-free Interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, 3257–3266.
16. Simon T. Perrault, Eric Lecolinet, James Eagan, and Yves Guiard. 2013. WatchIt: Simple Gestures and Eyes-free Interaction for Wristwatches and Bracelets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, 1451–1460.
17. Volker Roth and Thea Turner. 2009. Bezel Swipe: Conflict-free Scrolling and Multiple Selection on Mobile Touch Screen Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. ACM, 1523–1526.
18. Marcos Serrano, Eric Lecolinet, and Yves Guiard. 2013. Bezel-Tap Gestures: Quick Activation of Commands from Sleep Mode on Tablets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, 3027–3036.
19. Martin Weigel, Tong Lu, Gilles Bailly, Antti Oulasvirta, Carmel Majidi, and Jürgen Steimle. 2015. iSkin: Flexible, Stretchable and Visually Customizable On-Body Touch Sensors for Mobile Computing. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, 2991–3000.
20. Robert Xiao, Gierad Laput, and Chris Harrison. 2014. Expanding the Input Expressivity of Smartwatches with Mechanical Pan, Twist, Tilt and Click. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, 193–196.