Exploring Eyes-free Bezel-initiated Swipe on Round Smartwatches

Pui Chung Wong[1], **Kening Zhu**[1], **Xing-Dong Yang**[2], **Hongbo Fu**[1]
[1]City University of Hong Kong [2]Dartmouth College
tony.wong.pc, keninzhu, hongbofu@cityu.edu.hk
xing-dong.yang@dartmouth.edu

ABSTRACT
Bezel-based gestures expand the interaction space of touch-screen devices (e.g., smartphones and smartwatches). Existing works have mainly focused on bezel-initiated swipe (BIS) on square screens. To investigate the usability of BIS on round smartwatches, we design six different circular bezel layouts, by dividing the bezel into 6, 8, 12, 16, 24, and 32 segments. We evaluate the user performance of BIS on these layouts in an eyes-free situation. The results show that the performance of BIS is highly orientation dependent, and varies significantly among users. Using the Support-Vector-Machine (SVM) model significantly increases the accuracy on 6-, 8-, 12-, and 16-segment layouts. We then compare the performance of personal and general SVM models, and find that personal models significantly improve the accuracy for 8-, 12-, 16-, and 24-segment layouts. Lastly, we discuss the potential smartwatch applications enabled by the BIS.

Author Keywords
Bezel; Round smartwatches; Bezel-initiated gestures; Eyes-free

CCS Concepts
•**Human-centered computing** → **Gestural input;** *User studies; Mobile computing;*

INTRODUCTION
Wrist-worn devices are becoming increasingly popular in both research and commercial deployment. Smartwatch is commonly used for quick information access and handling simple daily tasks, such as viewing messages, making a phone call, controlling a music player. However, in many situations, interacting with such devices is inaccurate and often requires extra sequential operations due to its small form factor and limited input space. Voice input is an alternative input method, but it may suffer from inaccuracy in noisy environments and might be socially inappropriate when used in quiet situations. Therefore, input methods using other modalities need to be explored. Researchers have proposed various approaches to increase the input vocabulary of smartwatches, such as adding extra hardware or sensors [8, 17, 33] and enhancing interaction with built-in sensors (e.g., continuous pressure touch [37], target selection with wrist tilting [10]).

The limited interaction space of a smartwatch can also be enhanced by bezel-initiated gestures. The bezel refers to the physical touch insensitive frame surrounding the touch screen of a device. There are a variety of bezel-initiated gestures, such as, swiping from the bezel inwards [21, 22], pressing on bezel [37], bezel-to-bezel swipe [16], and sequential tapping started from bezel [32]. In this paper, we focus on *bezel-initiated swipe* (BIS) used in [21], instead of bezel tapping, since the detection of bezel tapping may require extra sensors installed on the watch. A BIS begins with a finger swiping from the bezel inwards to the touch screen and ends by lifting up the finger from the touch screen.

BIS is a fast, subtle, and natural action, and has a few unique benefits. First, BIS does not require any extra hardware or sensor, and is available in every smartwatch. Second, it can be performed without mode switching and co-exist with other touch gestures, e.g., for menu control and item selection. Third, BIS can be executed without looking at the smartwatch surface (referred as "eyes-free") due to the tactile feedback at the edges of the watch. Thus, BIS-based smartwatch interaction can be unobtrusive and is useful in various social scenarios, such as in a meeting or when talking with friends. Eyes-free gestures on smartwatch can also act as inputs to facilitate navigation with smart glasses and virtual reality [12, 14, 31].

While existing works on bezel-initiated gestures have largely focused on square mobile devices, we are more interested in exploring the use of BIS on round smartwatches in an eyes-free condition. On a square watch, its four edges and corners act as always-available tactile guides for users to locate their touch locations without looking at the watch. On a round watch, with users' natural spatial awareness of touch locations, users can still distinguish the rough locations, like top, bottom, left, and right, in an eyes-free condition. However, due to the lack of tactile guides available on square watches, it becomes more challenging for identifying precise touch locations, making eyes-free bezel-initiated gestures possibly more difficult.

We conducted a user study to investigate the user performance of eyes-free BIS on a round smartwatch with six different division layouts (6-, 8-, 12-, 16-, 24-, and 32-segments) of its circular bezel region. Our results reveal that the number of bezel segments significantly affects the performance of selecting correct bezel segments with BIS, for example, with the accuracy of 6- and 8-segment layouts being 93.31% and

CHI'20, April 25–30, 2020, Honolulu, HI, USA

ISBN 978-1-4503-6708-0/20/04. . . $15.00
DOI: https://doi.org/10.1145/3313831.3376393

81.80%, respectively. From the collected data, we further investigate the potential of improving the BIS accuracy with supervised machine learning. It is found that support vector machines (SVMs) achieve the best improvement compared with k-nearest neighbors, logistic regression, and random forest. The SVM has significant improvement on the accuracy of 6-, 8-, 12-, and 16-segment layouts. Finding varying performance among users, we conducted another six-day study to derive and evaluate the performance of a personal SVM model compared with a general SVM model. The personal model further yields significant improvement on 8-, 12-, 16-, 24-, and 32-segment layouts.

Our contributions are threefold:

- The first work studying the performance of eyes-free BIS on round smartwatches;
- Machine-learning-based analysis and prediction models for eyes-free BIS on round smartwatches;
- Discussions on potential smartwatch applications enabled by SVM-enhanced BIS.

RELATED WORKS

In this section, we present existing literature in expanding the interaction space for smartwatch, marking menu, eyes-free smartwatch interaction, and bezel-initiated interaction for smartwatch.

Expanding Smartwatch Input Space

Researchers have proposed various input techniques to expand the input space on smartwatches. These approaches can be classified into two categories: using extra hardware and using the built-in sensors in watches. Skin Buttons [20] and SkinWatch [27] allow on-skin input around smartwatch with infrared proximity and distance sensors. WristWhirl [8] enables wrist whirling input with proximity sensors. WristOrigami [41] manipulates multiple touch panels around the watch. On the other hand, approaches using only built-in sensors expand the input space by software analysis on IMU data [37] , temporal tapping [26] and detecting finger orientation [35]. Similarly, BIS also does not require extra hardware. In addition, it is a familiar gesture to users, since it has already been used in daily mobile scenarios (e.g., for manipulating the smartphone notification bar).

Marking Menu

A marking menu allows users to perform a menu selection by making a straight mark in the direction of a target menu item on a radial or pie menu [18]. It is believed that the directions along the horizontal and vertical axes and the directions between them (e.g., compass directions), are easy to remember and distinguish [19]. Due to the limited interaction space on mobile devices, unistroke gestures in marking menu are difficult to perform. Multi-stroke marking menu [39] requires less space by using a series of simple straight marks for sub-selection. Machine learning enhanced BIS might support multi-stroke marking menu with more items in each level and thus improve its performance.

Eyes-free Smartwatch Interaction

Researchers have also started investigating eyes-free smartwatch interaction to reduce the attention load of users. Cheung et al. [6] suggest eyes-free input with a deformable wristband. Pasquero et al. [28] present a haptic wristwatch to facilitate eyes-free smartwatch interaction. The work of [8] suggests using wrist whirling as input on smartphones or smart glasses. Wong et al. [36] develop FingerT9, and show that users can efficiently perform eyes-free thumb-to-finger tapping for text entry on smartwatches. Meanwhile, smartwatch can facilitate user input in virtual reality environments (eyes-free usage), including 3D pointing for navigation[14], hand gestures for quick operations[31], and joystick-like control [12]. Blasko et al. [4] put tactile landmarks on the bezel of a round smartwatch and allow users to move the fingertip along the bezel to decrease the dependence of the GUI on visual display. This suggests that smartwatch as an always-carried device is potential and useful to support eyes-free bezel-based inputs in different scenarios.

Bezel-initiated gestures on mobile devices

Various touch gestures utilizing the device bezels have been proposed on smartphones [5, 13, 21], tablets [15, 32], and smartwatches [16, 11]. All these approaches suggest to use BIS on rectangular devices for target selection, quick command activation, and providing additional functions, like text editing. Among them, a few works have focused on smartwatches. For example, PageFlip [11] utilizes a continuous bezel swipe action from corners for quick text editing. B2B Swipe [16] leverages the tactile feedback obtained by touching a smartwatch to allow double-crossing touch gestures on bezels. These works show that BIS is useful to expand the input vocabulary and is potential to be used in an eyes-free condition. Although Ahn et al. [1] and Ashbrook [3] have investigated the interaction of *sliding along* bezels of square smartwatches and bezels of small round touch screens, BIS on round smartwatches is largely unexplored, and is the focus of this paper.

BEZEL-INITIATED GESTURES ON ROUND SMARTWATCHES

Existing studies [16, 11] of BIS on smartwatches have been focused on square smartwatches, and B2B-Swipe [16] supports eyes-free usage. Round smartwatches currently available in the market[1] also support bezel swipe, but the omnidirectional edge and lack of screen corners make it more challenging and error-prone to perform BIS than square smartwatches, especially in an eyes-free situation. In this paper, we study the user performance of selecting target bezel segments with eyes-free BIS on round smartwatches with different segment numbers and sizes. We define the bezel region on a smartwatch as one eighth of the diameter of its touch screen (50 pixels width in our implementation), the typically untouched display pixels [29]. We divide the bezel region into different numbers of segments to study the effect of different segment sizes. More specifically, we choose commonly used layouts and their subdivisions, i.e., 6, 8, 12, 16, 24, and 32 segments, resulting in six layouts (Figure 1).

[1] https://www.samsung.com/global/galaxy/galaxy-watch-active2/

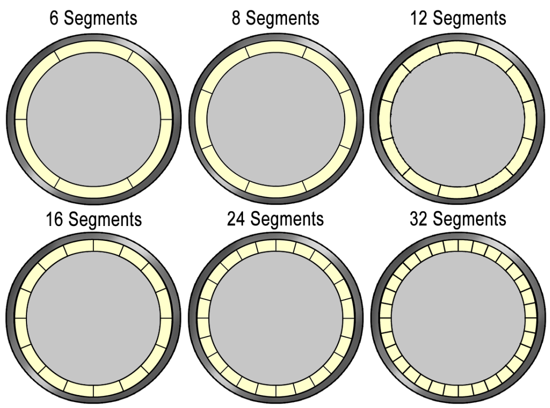

Figure 1. Layouts with 6, 8, 12, 16, 24, and 32 segments.

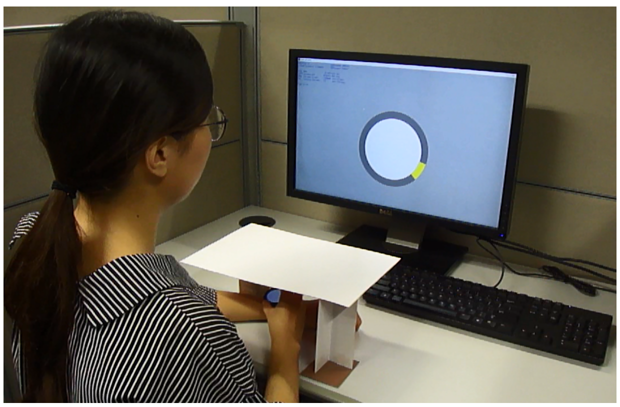

Figure 2. Study setup: each participant was performing a BIS on a round smartwatch in an eyes-free condition.

We determine the orientation of each layout as follows. For the 6-segment layout, we choose the orientation adopted in commercial smartwatches (e.g., Samsung Galaxy Watch 2). For 8- and 12-segment layouts, the orientations are the same as the eight compass directions and twelve directions of the analog clock, respectively. We also want to investigate the effect of small segment size on the performance and subdivide the layouts into smaller segments. For 16- and 24-segment layouts, we simply divide each segment from the 8- and the 12-segment layouts into halves, respectively. The 32-segment layout is a subdivision of the 16-segment layout, resulting in each segment with its size as small as 3.4mm, which is similar to the small target size in many mobile applications [30].

USER STUDY 1: USER PERFORMANCE OF EYES-FREE BIS IN DIFFERENCE LAYOUTS

The goal of this study is to investigate the user performance of eyes-free BIS on the six layouts mentioned above.

Participants

We recruited 12 right-handed participants (4 female; aged between 20 and 35), wearing their watches on their left wrists. Five participants were everyday watch users, and two of them used smartwatches in their daily lives.

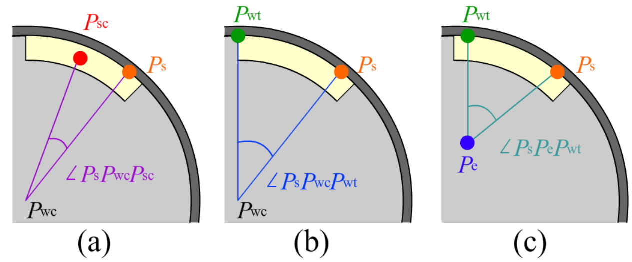

Figure 3. (a) Relative offset; (b) Angle from center; (c) Angle between start and end points.

Apparatus

The smartwatch used in the study was a 1.4-inch Ticwatch 2 [2] having a touch screen with resolution of 400 x 400 pixels. During the study, each participant sat in a chair and placed their hands comfortably and horizontally on a table (Figure 2). A monitor was placed in front of the participant to display the tasks. All participants wore the smartwatch on their left wrists and did the tasks with their right index fingers. A cardboard was placed above the participants' hands to ensure that the participants could not see the smartwatch screen.

Task and Procedure

The task required each participant to perform a BIS from the highlighted bezel segment toward the center, i.e., to select the highlighted segment to start a BIS gesture. The participants were instructed to perform the BIS task as naturally and as accurately as possible. Upon the end of a trial, the next trial started automatically. No feedback was given to the participants during the trials, except the next trial appeared on the monitor to indicate the end of each trial. This avoided slowing down the swipe of BIS due to visual or auditory feedback and influencing the accuracy of the next BIS due to the result of the current BIS. In order to capture their actions as natural as possible, the participants were told that their actions would be always correctly detected by the system.

There were six sessions which tested the six layouts. The order of the six sessions tested was counter-balanced among the participants and the bezel-segment targets appeared in a random order within each session. Prior to the study, each participant was given 1 to 3 minutes to get familiar with BIS and the system, but without practicing the locations of targets. After each session, the participants could take a 3-minute break if needed. The study can be summarized as: 12 participants $\times$ 98 segments (from 6 layouts) $\times$ 20 repetitions = 23,520 trials.

During the study, segment selection was made once a BIS was performed: even if a target segment was not selected correctly, the next trial would still appear. In a few cases, however, the participants accidentally touched the bezel and swiped before they were ready to input. These accidentally-completed trials covered less than 2% of the total number of trials. For an accidentally-completed trial, the participants redid the task by pressing the left arrow key on a physical keyboard under the monitor.

[2]Ticwatch 2: https://www.mobvoi.com/hk/types/wearable

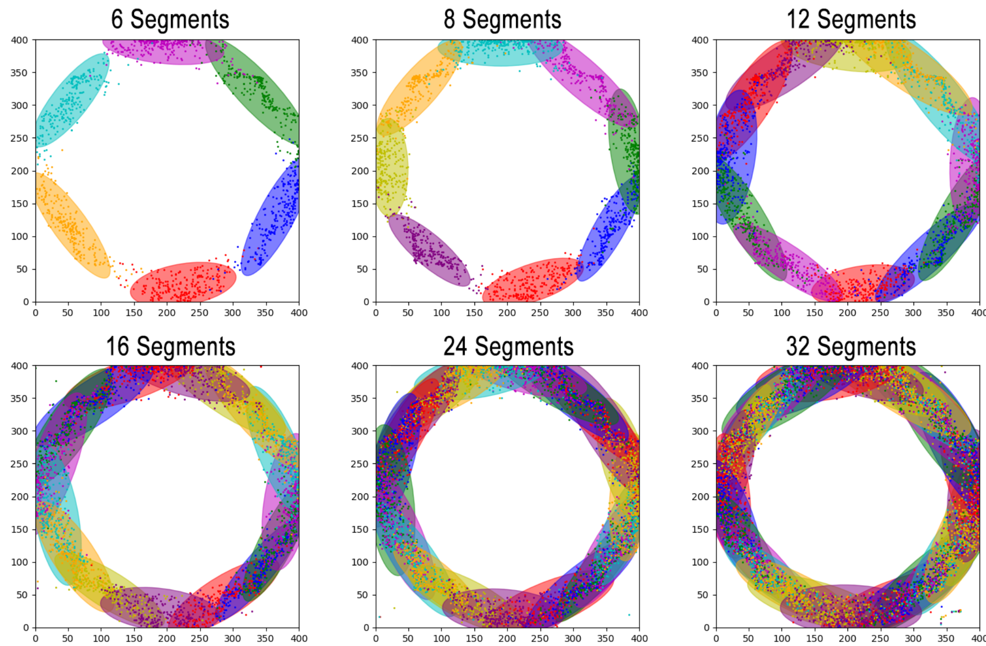

Figure 4. Touch point scatter plots with 95% confidence ellipses in all the tested layouts.

Data Collection

We collected the following data: start and end positions of BIS, and time stamps of the start and end of BIS. With these data, absolute offset, relative offset, and task completion time were calculated. The start position, denoted as p_s, refers to the finger-down position on the touch screen where the user starts the BIS, and the end position refers to the finger-up position where the user leaves the index finger from the touch screen. Let p_{sc}, p_{wc}, and p_{wt} be the target segment center, smartwatch center position, and smartwatch topmost position, respectively. The relative offset ($\angle p_s p_{wc} p_{sc}$ in Figure 3a) refers to the angle between p_s and p_{sc} with p_{wc} as the origin. A positive value means that the start position is on the right of the target segment center. The absolute offset is the absolute value of the relative offset. The task completion time refers to the time taken to perform a BIS, from finger down to finger up.

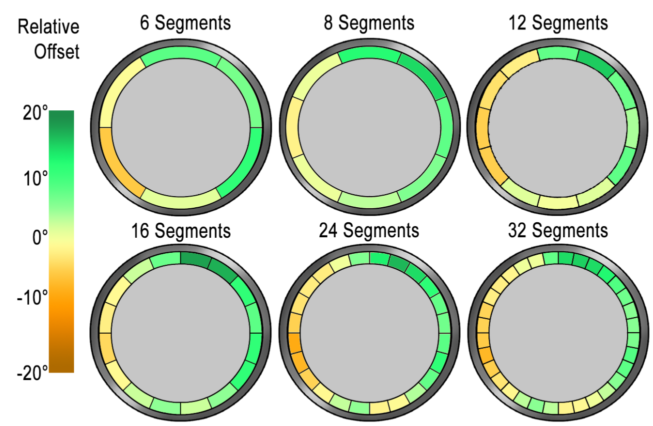

Figure 5. Relative offset per segment of the six layouts.

Results

The touch points recorded in this study refer to the finger-down positions, i.e., the first detected points by the touch screen. Figure 4 shows the distribution of the touch points from all the participants with 95% confidence ellipses among all the layouts. For the 6-segment layout, the ellipses are separated nicely with only a small overlap between the top and the top-right segments. For the 8-segment layout, there are more overlaps at the leftmost and rightmost areas of each segment except the bottom-left one. For the rest of the layouts, it is observed that the finger-down locations are noisy with considerable overlaps among different ellipses. Increasing in the number of segments, the size of overlapping areas among segments increases. This suggests that the participants were able to identify the segment locations for small segment numbers while it became more difficult with more segments since the segment size became smaller.

We measured the performance of the six layouts with accuracy, absolute offset, relative offset, and task completion time, as summarized in Table 1. The accuracy is calculated by the number of tasks with correct selection divided by the total number of tasks. When a BIS starts at the position within a target segment, a selection is considered correct. The accuracy achieves 95.34% for the 6-segment layout and decreases gradually to 25% for the 32-segment layout. In contrast, the absolute offset, relative offset, and task completion time among the six layouts do not have large differences. Figure 5 shows the relative offset of each segment. It is observed that the participants tended to have positive offsets on the right side and negative offsets on the left side of the smartwatch bezel. The participants started

Segment Number	Accuracy	Absolute Offset	Relative Offset	Completion Time
6	93.34% (SD = 0.07)	13.42° (SD = 4.31)	3.74° (SD = 9.47)	238.7ms (SD = 100.0)
8	81.80% (SD = 0.13)	13.02° (SD = 3.63)	6.03° (SD = 8.42)	248.0ms (SD = 108.2)
12	59.27% (SD = 0.17)	14.53° (SD = 4.53)	2.04° (SD = 11.39)	295.2ms (SD = 162.7)
16	43.88% (SD = 0.13)	15.19° (SD = 4.22)	6.11° (SD = 9.59)	262.7ms (SD = 136.1)
24	32.80% (SD = 0.11)	14.38° (SD = 4.73)	3.12° (SD = 10.16)	299.7ms (SD = 165.5)
32	25.00% (SD = 0.10)	14.57° (SD = 4.89)	2.86° (SD = 10.59)	276.3ms (SD = 135.6)

Table 1. Average accuracy, absolute offset, relative offset, completion time of all the layouts from 12 participants.

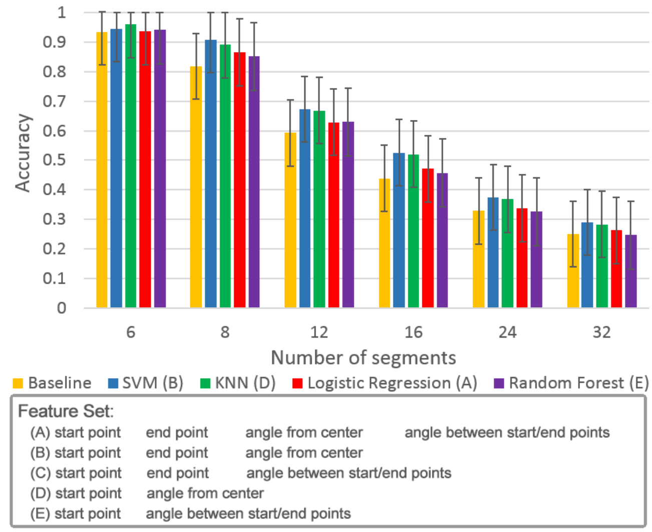

Figure 6. The accuracy among the baseline, and the four classification methods trained with their best feature sets (shown in the brackets).

BIS with shifting on right especially on the top-right regions and with shifting on left especially on the left regions. This behavior could be mainly due to the relative angle between the smartwatch and the interacting index finger, and the results for left-handedness and right-handedness may be mirrored and are subject to further investigation.

The repeated-measures ANOVA revealed that there are significant effects among the layouts on accuracy ($F(5,66) = 58.354$, $p < 0.001$, $\eta^2 = 0.999$), but no significant difference in term of absolute offset, relative offset, and task completion time. For accuracy, post-hoc pairwise comparison showed that there are significant difference ($p < 0.001$) between pairs of layouts except the pairs of 6 and 8 segments, 12 and 16 segments, 16 and 24 segments, and 24 and 32 segments. For each layout, repeated-measures ANOVA showed that there are significant between-users differences on accuracy, absolute offset, relative offset, and task completion time (all $p < 0.001$). This reflects that users' behaviors on performing BIS vary even for the same layouts. Among the segments within each layout, there are also significant effects on accuracy, absolute offset, relative offset, and task completion time (all $p < 0.001$). This suggests that BIS is orientation dependent: different segments located in different orientations affect the performance.

All participants were unconfident to locate the targets for 24-, and 32-segment layouts. 4, 6, and 2 participants preferred 8-, 12-, and 16-segment layouts, respectively. One participant told us *"It is impossible to distinguish among such small targets (in 32-segment layout) correctly. I can only locate their positions roughly."* Another participant commented that *"I am confident and can quickly locate large targets (in 6-, 8-, and 12-segment layout) without thinking much. I need more time to locate the small targets in the layouts with many segments (24 and 32 segments) as they require high precision."* Two participants said, *"Adding physical landmarks on the bezel may help me find the targets"* and *"I may locate the targets better if there is visual feedback."* This reflected that further guidance may be useful for locating desired segments, though such guidance might slow down the performance of BIS.

IMPROVING BIS ACCURACY WITH SUPERVISED MACHINE LEARNING

From Study 1, we found that BIS was orientation dependent, and the error rate significantly increased with the increasing number of segments. Observing that the detection of BIS from different bezel segments is essentially a classification problem, next we explore the possibility of improving BIS detection through machine-learning techniques. Each segment of a layout is regarded as a class. For example, BIS on a 6-segment layout is essentially a 6-class classification problem. We first randomly divided the data collected in Study 1 from the 12 participants into two sets: 30% for testing and 70% for training (i.e., to randomly keep 70% of each participant's data as training set). The probabilistic models were trained and validated using the training data to improve the accuracy of BIS detection. The trained models were then evaluated with the testing set.

First, we determine the classification methods and the features to be used. We consider four commonly used classification methods, including support-vector machine (SVM), k-nearest neighbors algorithm (KNN), logistic regression, and random forest. We derive four individual features for a BIS gesture, including its start (finger-down) position, end (finger-up) position, angle of start position from center, and angle between start and end positions.Let p_e and p_{wt} be the end position and smartwatch topmost position, respectively. The angle from center ($\angle p_s p_{wc} p_{wt}$ in Figure 3b) is the angle of the start position p_s with the smartwatch center p_{wt} as the origin. The angle between the start and end points ($\angle p_s p_e p_{wt}$ in Figure 3c) is the angle of the start position with the end position as the origin with respect to the smartwatch topmost direction. We define five feature sets (feature sets A to E in Figure 6) with different combinations of these four individual features. The start position is probably the most important feature determining which bezel region is selected, so it is included in every feature set. The performance of each classification method may vary

Segment Number	Baseline Accuracy	SVM Accuracy
6	93.03% (SD = 14.96%)	95.60% (SD = 14.07%)
8	81.94% (SD = 22.72%)	93.75% (SD = 12.60%)
12	57.41% (SD = 32.76%)	67.25% (SD = 39.13%)
16	44.79% (SD = 29.90%)	54.08% (SD = 27.38%)
24	33.51% (SD = 24.54%)	39.70% (SD = 25.62%)
32	24.61% (SD = 23.10%)	28.91% (SD = 22.24%)

Table 2. The average accuracy of the baseline and SVM of the six layouts with significant difference highlighted in blue.

with different training features used. To obtain the optimal feature set for each method, we compare the performance of each feature set on all the four methods. Validating the four methods with the five feature sets, it is found that feature set B, D, A, and E gave the best performance for SVM, KNN, logistic regression, and random forest, respectively.

Figure 6 shows the baseline accuracy and the 10-fold cross validation accuracy using the training set data on the four classification methods trained with their optimal feature sets. The baseline accuracy is calculated with the same way in Study 1 result analysis. All the four classification methods improved the accuracy over the baseline except for random forest with the 32-segment layout. Among the six layouts, KNN achieved the highest accuracy of 94.52% in the 6-segment layout and SVM achieved the highest accuracy of 90.74%, 67.16%, 52.62%, 37.44%, 28.95% in 8-, 12-, 16-, 24-, 32-segment layouts, respectively. Thus, we chose SVM with its optimal feature set B as our classification method for the subsequent analysis, and applied it as a general classification model to predict the testing set for each participant in Study 1. Table 2 showed the average baseline accuracy and SVM accuracy on the testing set. In overall, SVM improved the BIS-detection accuracy for all the six layouts. Paired T-test revealed that there is a significant improvement by SVM on 6-segment ($t(11) = -4.749$, $p < 0.001$), 8-segment ($t(11) = -3.252$, $p < 0.01$), 12-segment ($t(11) = -2.405$, $p<0.05$), and 16-segment ($t(11) = -2.246$, $p<0.05$) layouts.

USER STUDY 2: GENERAL MODEL VS PERSONAL MODEL

In the previous section, we found that using an SVM classification model significantly improved the BIS detection accuracy. On the other hand, the SVM model derived above was a general model trained with the data from all the participants in the Study 1. Since we observed there are significant differences in performance among users in Study 1, this suggested that a personal model might make further improvement and could yield better performance over a general model. Therefore, we conducted a study to collect more data in order to investigate the performance difference between the general and the personal SVM-based classification models.

Participants

3 right-handed participants (all male) of aged between 21 and 24 were recruited. All of them did not use smartwatches in their daily lives. All of them did not participate in Study 1.

Apparatus

We used the same apparatus as in Study 1.

Study Design

The study lasted six days for all the participants, with six sessions per day, one session for each layout. For each session, one of the six layouts was tested. The order of layouts was counter-balanced among the six days. Prior to the study in Day 1, the participants were asked to practice for several minutes to get familiar with BIS and our system. The participants then started the study and performed the tasks following the same procedures in Study 1. They were encouraged to take short breaks between sessions. The six experimental sessions of a day lasted on average around 45 minutes. We conducted the study in six days to prevent the participants becoming fatigued, thus affecting the data collected. In total, we collected 3 participants $\times$ 98 segments (from 6 layouts) $\times$ 20 repetitions $\times$ 6 days = 35,280 trials. Each participant performed 11,760 trials.

Results and Discussions

To investigate the performance of a personal classification model, the data collected from this study was used to train a personal model for each user. The collected data was divided into the training and testing sets by the number of days for training. For example, in the case of two-day training, we used the data of the first two days as the training set and the data of the remaining four days as the testing set. We compared the accuracy of the baseline, general SVM (derived from Study 1), and personal SVM with the same testing sets. Figure 7 showed the average accuracy of the baseline, the general SVM, and the personal SVM trained with the data using different numbers of day as training set. For 6-segment layout, the baseline, general SVM, and personal SVM achieved 94.38%, 96.11%, and 95.10% average accuracies among all training days, respectively, with variance of 0.0075% under different training days. For 8-, 12-, 16-, 24-, and 32-segment layouts, the overall accuracies of the three methods do not have much differences with one-day training. With the increase in the training data (i.e., taking more days of user data for training), the accuracy of personal SVM improved gradually over the baseline and the general SVM. With two or more days training, the accuracy of personal SVM became the highest among the three methods and it kept growing with the increasing number of training data. For 8-segment layout, the accuracy of the personal SVM increased significantly ($F(2,24) = 6.186$, $p < 0.01$) with two-day training comparing with general SVM and increased with three- to five-day training data but not significantly. For 12-, 16-, 24-, and 32-segment layouts, the accuracy of the personal SVM increased gradually when the user data for training increased and it kept growing with five-day training. It is possible that the accuracy of the personal SVM did not reach their peaks and would keep growing if trained for more days.

Repeated-measured ANOVA showed significant improvement in accuracy in 8-segment ($F(2,15) = 8.329$, $p < 0.01$), 12-segment ($F(2,15) = 4.013$, $p < 0.05$), 16-segment ($F(2,15) = 4.755$, $p < 0.05$), and 24-segment ($F(2,15) = 7.310$, $p < 0.01$) layouts with four-day user data as the training sets. Post-hoc

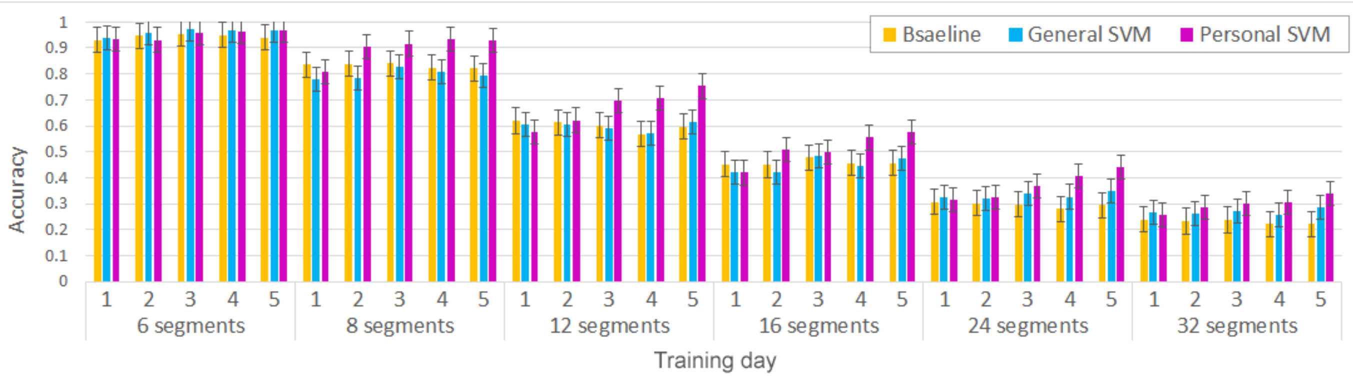

Figure 7. The comparison in the average accuracy between the baseline, general SVM model, and personal SVM models with data from different numbers of days as training data.

pairwise comparison showed that the personal SVM yielded significant increase in accuracy over the baseline and general SVM (both $p < 0.05$) for the 8-segment layout, and achieved 93.34%. In the 8-segment layout, the personal SVM made over 10% accuracy improvement and turned an originally less accurate layout to a more precise and usable layout. Besides, personal SVM improved significantly over general SVM in the 16-segment layout ($p < 0.05$) and improved significantly over the baseline in the 24-segment layout ($p < 0.001$). Although the improved accuracy for the 16-segment and 24-segment layouts was 70.72% and 55.49%, respectively, which was not high, the improvement might be useful with the refinement or other applications combining with other probability models (e.g., statistical decoder for text entry [9, 38, 43, 23, 40]).

All the participants were unconfident to locate the targets for 24-, and 32-segment layouts throughout the six-day study period. On the third day, one participant said, *"I felt I could locate the targets of the 16-segment layout faster and more accurately after practicing for a few days. But it is still difficult to precisely locate smaller targets in 24- and 32-segment layouts."*

Although a live study with the derived personal classification models was not conducted, it is believed that the investigation on the performance of a personal model through the simulation is close to such a live study. The average recognition time of a BIS with SVM is 1.984e-06s (on a PC with Intel Core i5-3450 3.1GHz CPU, 4GB RAM), which is largely shorter than the suggested real-time response time of 0.1s [25]. We thus believe that the recognition time would not affect participants' behavior of BIS in real-time.

APPLICATIONS

To illustrate the feasibility and potential smartwatch applications with SVM-enhanced BIS, we suggested three applications: bezel-initiated text entry, bezel shortcuts, and bezel menu control. These applications support an eyes-free usage, and do not require users to look at the smartwatch touch screen.

Bezel-initiated Text Entry

Text entry and editing is one of the most common activities in mobile and wearable devices [36, 42]. To facilitate text

Figure 8. A user performs an eyes-free BIS gesture to select a multi-letter key in a circular keyboard when experienced in a VR environment.

entry on smartwatches, several techniques [2, 34, 7, 38] have been proposed to use a circular multi-letter keyboard with a statistical keyboard decoder. A statistical keyboard decoder consists of a spatial model, using a probability distribution to handle noisy multiple-key inputs, and a language model, determining the probability of word candidates from the dictionary. Even for noisy input on a tiny screen, a language model is greatly helpful to improve the typing performance. However, the challenge is to strike a balance between key size and word disambiguation. It is even more difficult in an eyes-free condition, in which larger keys are preferable. WrisText [7] achieved 95.3% of corpus words in top three candidate with a six-key circular layout. Our Study 2 results showed that personal SVM model significantly improved the accuracy of 8- and 12-segment layouts in an eyes-free condition, and might be used as a spatial model to improve typing performance for 8- and 12-key circular keyboards. Increasing the number of keys, the number of letters per key is reduced, thus potentially decreasing the word disambiguation. For example, using a personal SVM model may make eight-key TouchOne [34] keyboard feasible for eyes-free typing in a secondary display (Figure 8). BIS is used for selecting multi-letter keys, and without mode switching touch gestures, such as, tap, swipe, and hold can be used for candidate confirmation and selection, delete, space, and switching to symbols.

Bezel Shortcuts

By associating different functions with different bezel segments, BIS allows users to control with other devices (e.g., smartphones, tablets, laptops) by a single swiping action start-

Figure 9. Perform BIS on different bezel segments to pause a video on a laptop and to reject a phone call.

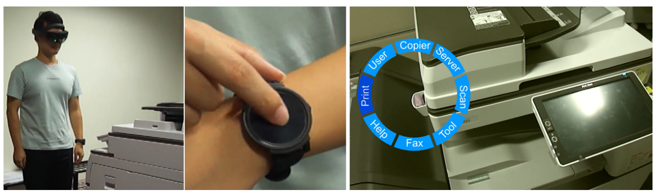

Figure 10. A user navigates with a circular menu using BIS to control a printer when wearing a HoloLens.

ing at different bezel positions of a smartwatch. This is fast, natural, and unobtrusive. For example, a user can quickly pause a video on a laptop or reject a phone call when talking to others with BIS started from different bezel segments (Figure 9). BIS can also make a smartwatch act as a remote controller of a smart home. The users can set their own shortcuts with their preferable BIS segments and interact with the household devices, such as, adjusting home lighting and turning on the air conditioner.

Bezel Menu Control

BIS provides an alternative input method for menu navigation in VR or AR environment. The user can interact with a printer and perform BIS on different segments to control the menu (Figure 10). BIS facilitates the use of existing VR or AR headsets and allows navigation in hierarchical menu without looking at the smartwatches. Besides, BIS can be used in VR environments for quick mode switching or input beyond the controller. One may argue that input in VR could be achieved by head/gaze tracking, but these two techniques may cause more fatigue than BIS.

DISCUSSIONS, LIMITATIONS AND FUTURE WORK

In this section, we discuss the insights gained from this work, the limitations, and the directions for future work.

Continuous Input with BIS

BIS supports discrete input and can potentially support continuous input. Similar to NotifEye [24] which uses three different swiping speeds and two browsing directions to support six different discrete inputs for navigation with smart glasses. Discrete and continuous inputs can be distinguished with short and long BIS swipe in the time domain and by the swiping distance. The results in Study 1 showed that there was no significant difference in swiping time among the six layouts and among users. However, how swiping length may vary with different bezel regions and the threshold to distinguish between long and short BIS swipe still need further investigation. Apart from the swiping time and length, BIS can be performed in a curved trajectory instead of a straight swipe. For example, a BIS gesture toward left can be slightly upward or downward toward the top or bottom of the smartwatch bezel. This may further expands the input space with BIS and is potentially useful for secondary functions or selection control.

Additional Feedback

BIS leverages physical feedback from a smartwatch bezel, while distinguishing different bezel locations on round smartwatches becomes more difficult with an increasing number of segments. In Study 1, two participants suggested that visual or audio feedback may be helpful to identify which target was selected, especially in layouts with many segments. Additional feedback, such as audio, vibration, physical landmark may provide further guidance to help users locate desired segments. For example, a 12-segment layout can be divided into four quarter, each consisting of three segments. Three vibration patterns can then be applied onto each quarter and provide tactile cues helping users identify among segments. With such feedback, users can know if touching a wrong segment, make a correction by locating a desired segment again before swiping inward to the touch screen for BIS.

User Study

We found that BIS is useful to enable eyes-free interaction in different mobile scenarios. Our two user studies were conducted in a sitting condition and other scenarios also deserve careful investigation for further study. For example, we plan to evaluate the usability of BIS in walking, standing with hand hanging down, and other mobile scenarios. Besides, large finger, long nails, wrist orientation, handedness, and input with thumb may affect the performance of BIS, deserving further investigation.

Sample Size

The performance of personal SVM model might vary with users and our work is the first step to understand the performance of BIS on round smartwatches. Increasing the sample size in the user studies, particularly for study 2, would provide more generalized results. In Study 2, we aimed to compare the performance between personal SVM and general SVM models. The general SVM model was derived from Study 1 and trained with the data from 12 participants. To train personal SVM models, sufficient data were needed from an individual user (11,760 trial per participant), and thus we focused on studying the performance of personal SVM over days with respect to the increasing amount of training data. In the future, we plan to study the performance of BIS with larger samples for more in-depth investigation.

Adaptive Learning

The performance of BIS varies among different users and the results in Study 2 showed that personal models outperformed the general model and improved individual's performance. In the future work we will investigate the effectiveness of an adaptive algorithm that can dynamically update the model from general to personal gradually. Users use a general classification model at the beginning and they perform BIS gestures. The BIS data is collected and used as the training data to update the classification model gradually and automatically. In this work, we did not consider the deep-learning techniques due to the limited training data, which may cause the issue of overfitting.

CONCLUSION

We investigated the usability of bezel-initiated swipe (BIS) on round smartwatches with six circular bezel layouts, evenly dividing the bezel into 6, 8, 12, 16, 24, and 32 segments. The performance of eyes-free BIS on round smartwatch was evaluated. The within-subject user study showed that BIS has a significant effect on accuracy among the six layouts, is orientation dependent, and varies among users. By analyzing machine-learning-based prediction models, we found that the SVM model has significant improvement in accuracy on 6-, 8-, 12-, and 16-segment layouts. We then conducted another six-day study to derive and evaluate the performance of personal SVM models. Compared with the general SVM model, the personal SVM models further yielded significant improvement in accuracy on the 8-, 12-, 16- and 24-segment layouts. BIS on round smartwatches potentially benefit various applications, ranging from mobile wearable to VR. Enabling BIS on round smartwatches and in an eyes-free condition serve as an important groundwork for future investigation on the interaction of round smartwatches.

Acknowledgement

We thank the anonymous reviewers for their insightful suggestions, and the user-study participants for their great input to the project. This research was supported by the Open Project Program of State Key Laboratory of Virtual Reality Technology and Systems, Beihang University (Project No. VRLAB2018C11), the Young Scientists Scheme of the National Natural Science Foundation of China (Project No. 61907037), the Centre for Applied Computing and Interactive Media (ACIM) of School of Creative Media, and Shenzhen Research Institute, City University of Hong Kong.

REFERENCES

[1] Sunggeun Ahn, Jaeyeon Lee, Keunwoo Park, and Geehyuk Lee. 2018. Evaluation of Edge-based Interaction on a Square Smartwatch. *Int. J. Hum.-Comput. Stud.* 109, C (Jan. 2018), 68–78. DOI: http://dx.doi.org/10.1016/j.ijhcs.2017.08.004

[2] Kohei Akita, Toshimitsu Tanaka, and Yuji Sagawa. 2018. *SliT: Character Input System Using Slide-in and Tap for Smartwatches*. 3–16. DOI: http://dx.doi.org/10.1007/978-3-319-91250-9_1

[3] 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, New York, NY, USA, 311–314. DOI: http://dx.doi.org/10.1145/1409240.1409276

[4] Gabor Blasko and Steven Feiner. 2004. An Interaction System for Watch Computers Using Tactile Guidance and Bidirectional Segmented Strokes. In *Proceedings of the Eighth International Symposium on Wearable Computers (ISWC '04)*. IEEE Computer Society, Washington, DC, USA, 120–123. DOI: http://dx.doi.org/10.1109/ISWC.2004.6

[5] Chen Chen, Simon T. Perrault, Shengdong Zhao, and Wei Tsang Ooi. 2014. 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, New York, NY, USA, 185–192. DOI: http://dx.doi.org/10.1145/2598153.2598162

[6] Victor Cheung, Alex Keith Eady, and Audrey Girouard. 2017. Exploring Eyes-free Interaction with Wrist-Worn Deformable Materials. In *Proceedings of the Eleventh International Conference on Tangible, Embedded, and Embodied Interaction (TEI '17)*. ACM, New York, NY, USA, 521–528. DOI: http://dx.doi.org/10.1145/3024969.3025087

[7] Jun Gong, Zheer Xu, Qifan Guo, Teddy Seyed, Xiang 'Anthony' Chen, Xiaojun Bi, and Xing-Dong Yang. 2018. WrisText: One-handed Text Entry on Smartwatch Using Wrist Gestures. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 181, 14 pages. DOI: http://dx.doi.org/10.1145/3173574.3173755

[8] Jun Gong, Xing-Dong Yang, and Pourang Irani. 2016. WristWhirl: One-handed Continuous Smartwatch Input Using Wrist Gestures. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. ACM, New York, NY, USA, 861–872. DOI: http://dx.doi.org/10.1145/2984511.2984563

[9] Mitchell Gordon, Tom Ouyang, and Shumin Zhai. 2016. WatchWriter: Tap and Gesture Typing on a Smartwatch Miniature Keyboard with Statistical Decoding. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 3817–3821. DOI: http://dx.doi.org/10.1145/2858036.2858242

[10] Anhong Guo and Tim Paek. 2016. Exploring Tilt for No-touch, Wrist-only Interactions on Smartwatches. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '16)*. ACM, New York, NY, USA, 17–28. DOI:http://dx.doi.org/10.1145/2935334.2935345

[11] Teng Han, Jiannan Li, Khalad Hasan, Keisuke Nakamura, Randy Gomez, Ravin Balakrishnan, and Pourang Irani. 2018. PageFlip: Leveraging

Page-Flipping Gestures for Efficient Command and Value Selection on Smartwatches. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 529, 12 pages. DOI: http://dx.doi.org/10.1145/3173574.3174103

[12] Teresa Hirzle, Jan Rixen, Jan Gugenheimer, and Enrico Rukzio. 2018. WatchVR: Exploring the Usage of a Smartwatch for Interaction in Mobile Virtual Reality. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems (CHI EA '18)*. ACM, New York, NY, USA, Article LBW634, 6 pages. DOI:http://dx.doi.org/10.1145/3170427.3188629

[13] 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, New York, NY, USA, 2221–2230. DOI: http://dx.doi.org/10.1145/2207676.2208376

[14] Daniel Kharlamov, Brandon Woodard, Liudmila Tahai, and Krzysztof Pietroszek. 2016. TickTockRay: Smartwatch-based 3D Pointing for Smartphone-based Virtual Reality. In *Proceedings of the 22Nd ACM Conference on Virtual Reality Software and Technology (VRST '16)*. ACM, New York, NY, USA, 365–366. DOI: http://dx.doi.org/10.1145/2993369.2996311

[15] Sangtae Kim, Jaejeung Kim, and Soobin Lee. 2013. Bezel-flipper: Design of a Light-weight Flipping Interface for e-Books. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems (CHI EA '13)*. ACM, New York, NY, USA, 1719–1724. DOI: http://dx.doi.org/10.1145/2468356.2468664

[16] Yuki Kubo, Buntarou Shizuki, and Jiro Tanaka. 2016. B2B-Swipe: Swipe Gesture for Rectangular Smartwatches from a Bezel to a Bezel. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 3852–3856. DOI: http://dx.doi.org/10.1145/2858036.2858216

[17] Hiroki Kurosawa, Daisuke Sakamoto, and Tetsuo Ono. 2018. MyoTilt: A Target Selection Method for Smartwatches Using the Tilting Operation and Electromyography. In *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '18)*. ACM, New York, NY, USA, Article 43, 11 pages. DOI: http://dx.doi.org/10.1145/3229434.3229457

[18] Gordon Kurtenbach and William Buxton. 1993. The Limits of Expert Performance Using Hierarchic Marking Menus. In *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems (CHI '93)*. ACM, New York, NY, USA, 482–487. DOI: http://dx.doi.org/10.1145/169059.169426

[19] Gordon P. Kurtenbach, Abigail J. Sellen, and William A. S. Buxton. 1993. An Empirical Evaluation of Some Articulatory and Cognitive Aspects of Marking Menus. *Hum.-Comput. Interact.* 8, 1 (March 1993), 1–23. DOI: http://dx.doi.org/10.1207/s15327051hci0801_1

[20] 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, New York, NY, USA, 389–394. DOI: http://dx.doi.org/10.1145/2642918.2647356

[21] Wing Ho Andy Li, Hongbo Fu, and Kening Zhu. 2016. BezelCursor: Bezel-Initiated Cursor for One-Handed Target Acquisition on Mobile Touch Screens. *Int. J. Mob. Hum. Comput. Interact.* 8, 1 (Jan. 2016), 1–22. DOI: http://dx.doi.org/10.4018/IJMHCI.2016010101

[22] Wing Ho Andy Li, Kening Zhu, and Hongbo Fu. 2017. Exploring the Design Space of Bezel-Initiated Gestures for Mobile Interaction. *Int. J. Mob. Hum. Comput. Interact.* 9, 1 (Jan. 2017), 16–29. DOI: http://dx.doi.org/10.4018/IJMHCI.2017010102

[23] Yiqin Lu, Chun Yu, Xin Yi, Yuanchun Shi, and Shengdong Zhao. 2017. BlindType: Eyes-Free Text Entry on Handheld Touchpad by Leveraging Thumb's Muscle Memory. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 2, Article 18 (June 2017), 24 pages. DOI:http://dx.doi.org/10.1145/3090083

[24] Andrés Lucero and Akos Vetek. 2014. NotifEye: Using Interactive Glasses to Deal with Notifications While Walking in Public. In *Proceedings of the 11th Conference on Advances in Computer Entertainment Technology (ACE '14)*. ACM, New York, NY, USA, Article 17, 10 pages. DOI: http://dx.doi.org/10.1145/2663806.2663824

[25] Jakob Nielsen. 1994. *Usability engineering*. Elsevier.

[26] 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, New York, NY, USA, 1237–1246. DOI:http://dx.doi.org/10.1145/2702123.2702226

[27] Masa Ogata, Ryosuke Totsuka, and Michita Imai. 2015. SkinWatch: Adapting Skin As a Gesture Surface. In *SIGGRAPH Asia 2015 Emerging Technologies (SA '15)*. ACM, New York, NY, USA, Article 22, 2 pages. DOI: http://dx.doi.org/10.1145/2818466.2818496

[28] 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, New York, NY, USA, 1451–1460. DOI: http://dx.doi.org/10.1145/2470654.2466192

[29] 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, New York, NY, USA, 1523–1526. DOI:http://dx.doi.org/10.1145/1518701.1518933

[30] 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. DOI: http://dx.doi.org/10.1145/1385569.1385594

[31] Franca Alexandra Rupprecht, Achim Ebert, Andreas Schneider, and Bernd Hamann. 2017. Virtual Reality Meets Smartwatch: Intuitive, Natural, and Multi-Modal Interaction. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '17)*. ACM, New York, NY, USA, 2884–2890. DOI: http://dx.doi.org/10.1145/3027063.3053194

[32] 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, New York, NY, USA, 3027–3036. DOI: http://dx.doi.org/10.1145/2470654.2481421

[33] Ke Sun, Yuntao Wang, Chun Yu, Yukang Yan, Hongyi Wen, and Yuanchun Shi. 2017. Float: One-Handed and Touch-Free Target Selection on Smartwatches. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 692–704. DOI: http://dx.doi.org/10.1145/3025453.3026027

[34] Infiniti Technology. 2015. TouchOne Keyboard. (2015). https://play.google.com/store/apps/details?id=net.infiniti.touchone.nimbus&hl=en_US

[35] Feng Wang, Xiang Cao, Xiangshi Ren, and Pourang Irani. 2009. Detecting and Leveraging Finger Orientation for Interaction with Direct-touch Surfaces. In *Proceedings of the 22Nd Annual ACM Symposium on User Interface Software and Technology (UIST '09)*. ACM, New York, NY, USA, 23–32. DOI: http://dx.doi.org/10.1145/1622176.1622182

[36] Pui Chung Wong, Kening Zhu, and Hongbo Fu. 2018. FingerT9: Leveraging Thumb-to-finger Interaction for Same-side-hand Text Entry on Smartwatches. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 178, 10 pages. DOI: http://dx.doi.org/10.1145/3173574.3173752

[37] Hui-Shyong Yeo, Juyoung Lee, Andrea Bianchi, and Aaron Quigley. 2016. WatchMI: Pressure Touch, Twist and Pan Gesture Input on Unmodified Smartwatches. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '16)*. ACM, New York, NY, USA, 394–399. DOI: http://dx.doi.org/10.1145/2935334.2935375

[38] Xin Yi, Chun Yu, Weijie Xu, Xiaojun Bi, and Yuanchun Shi. 2017. COMPASS: Rotational Keyboard on Non-Touch Smartwatches. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 705–715. DOI: http://dx.doi.org/10.1145/3025453.3025454

[39] Shengdong Zhao and Ravin Balakrishnan. 2004. Simple vs. Compound Mark Hierarchical Marking Menus. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology (UIST '04)*. ACM, New York, NY, USA, 33–42. DOI: http://dx.doi.org/10.1145/1029632.1029639

[40] Jun Gong Te-Yen Wu Aditya Shekhar Nittala Xiaojun Bi Jürgen Steimle Hongbo Fu Kening Zhu Xing-Dong Yang Zheer Xu, Pui Chung Wong. 2019. TipText:Eyes Free Text Entry on a Fingertip Keyboard. In *Proceedings of the 32th Annual Symposium on User Interface Software and Technology (UIST '19)*. ACM, New York, NY, USA.

[41] Kening Zhu, Morten Fjeld, and Ayça Ünlüer. 2018a. WristOrigami: Exploring Foldable Design for Multi-Display Smartwatch. In *Proceedings of the 2018 Designing Interactive Systems Conference (DIS '18)*. ACM, New York, NY, USA, 1207–1218. DOI: http://dx.doi.org/10.1145/3196709.3196713

[42] Kening Zhu, Xiaojuan Ma, Haoyuan Chen, and Miaoyin Liang. 2017. Tripartite Effects: Exploring Users' Mental Model of Mobile Gestures under the Influence of Operation, Handheld Posture, and Interaction Space. *International Journal of Human–Computer Interaction* 33, 6 (2017), 443–459.

[43] Suwen Zhu, Tianyao Luo, Xiaojun Bi, and Shumin Zhai. 2018b. Typing on an Invisible Keyboard. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 439, 13 pages. DOI: http://dx.doi.org/10.1145/3173574.3174013