A Survey of Text Entry Techniques for Smartwatches

Ahmed Sabbir Arif[✉] and Ali Mazalek

Synaesthetic Media Laboratory, Ryerson University, Toronto, ON, Canada
{asarif,mazalek}@ryerson.ca

Abstract. The growing interest in wearable devices has resulted in a wave of novel and improved text entry techniques for smartwatches and other ultra-small devices. These techniques are not only diverse in nature but also evaluated in different experimental conditions. This makes it difficult for designers and researchers to compare the techniques, and their performances in terms of speed and accuracy. This paper reviews the most important text entry techniques for smartwatches and other ultra-small devices. It categorizes all techniques based on whether they use (a variant of) the standard QWERTY keyboard, a novel keypad or keyboard, or handwriting recognition, and discusses the design and evaluation of the techniques. It includes a table that displays the performances of these techniques in the most common text entry performance metrics.

Keywords: Input and interaction · Text entry · Text input · Smartwatches · Smaller devices · Ultra-small devices · Touchscreen · Gesture · Wearables

1 Introduction

Smartwatches and other wearable devices are becoming increasingly popular among mobile users [19]. Users nowadays can use several applications on their smartwatches, and can interact with the applications installed on a paired smartphone without ever touching the smartphone. Yet interaction with smartwatches is mostly limited to checking emails, texts, and social networking posts. This is primarily due to the unavailability of an effective text entry technique for smartwatches. Most other tasks (e.g., replying to a text message) require text entry in some capacity, thus to perform those tasks users are forced to use a more text entry friendly device (e.g. a smartphone or tablet).

Researchers from both academia and industry are attempting to address this issue by designing and developing novel and improved text entry techniques for smartwatches. Unfortunately, experimental data on smartwatch text entry performance reported in the literature varies widely due to the use of different development platforms, devices, and performance metrics. This makes it difficult to compare studies or to extract meaningful average performance data from this body of work. This makes it hard for designers and researchers to use and apply these results and works against the synthesis of a larger picture. This can cause re-exploration of design philosophies, slowing down the overall development process in the area.

To provide designers and researchers with a better understanding of the current developments in the area, this paper reviews the most important text entry techniques for smartwatches and other ultra-small devices. It categorizes all techniques based on whether they use (a variant of) the standard Qwerty keyboard, a novel keypad or keyboard, or handwriting recognition, and discusses the design and evaluation of these techniques. It excludes all techniques that, in theory, cannot function individually (i.e., require either additional devices or external sensors to function). It also excludes all speech recognition techniques.

Table 1 shows performances of the reviewed techniques from empirical evaluations, both in terms of speed and accuracy, when available. In most evaluations, participants were asked to transcribe short English phrases from the MacKenzie and Soukoreff set [24] using the examined technique(s). However, some studies used words [7] or phrases from a different set [20]. Most studies instructed participants to transcribe the phrases as fast and accurately as possible, and to correct their mistakes as they notice them. Table 1 displays entry speed in the standard words per minute (WPM), and accuracy in Error Rate (ER), Total Error Rate (TER), or Character Error Rate (CER) metrics [2]. For some techniques these metrics were derived from the other data reported in the literature.

2   The QWERTY Layout

The standard QWERTY is the most dominant keyboard layout in personal computers and handheld devices [3]. Therefore, many use a variant of QWERTY in text entry techniques for smartwatches with the hope that a familiar layout will encourage users to use it and will accommodate a faster transition from novice to expert. Although not optimized for ultra-small devices, many have also explored the possibility of using a miniature version of the standard QWERTY on smartwatches [16]. This section reviews all techniques that use the QWERTY layout in some capacity.

Virtual Sliding QWERTY (VSQ). This technique loads a smartphone QWERTY keyboard on smartwatches, but displays only a part of it on the screen (Fig. 1). To see an invisible region, the user has to change the view by dragging the keyboard to the intended direction [5]. To enter a character with this technique, the user first navigates to the region where it is located, providing it is not already in the visible area, then taps on the corresponding key. In addition to using the space and backspace keys, the user can also enter a space and backspace by performing left and right swipes on the input area, respectively.
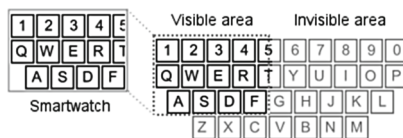

Fig. 1. The Virtual Sliding QWERTY (VSQ). To enter a character with this technique, the user drags the keyboard to a particular region and then taps on the intended key.

SplitBoard. Similar to VSQ [5], SplitBoard displays a partial view of a larger keyboard [17]. It divides the standard QWERTY into two main sections and includes an extra section for the digits, symbol, caps, and enter keys that enable the entry of numbers, symbols, and uppercase characters (Fig. 2). To enter a character, the user first navigates to the section where the character is located by performing horizontal swipes, providing it is not already in the visible area, then taps on the corresponding key. The space and backspace keys are located at the bottom of the screen, and can be selected by touching the bezel.
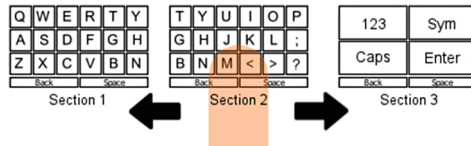

Fig. 2. The SplitBoard. To enter a character with this technique, the user navigates to a region by performing left and right swipes, and then taps on the intended key.

Swipeboard. Swipeboard [13] divides the standard QWERTY into nine regions (Fig. 3). Entering any character with this technique requires two swipes: the first swipe specifies the region where the character is located, and the second specifies the character within that region. These swipes can be performed anywhere on the screen, which eliminates the need for precise target selection. For space and backspace, the user has to perform a double-swipe diagonally down to the right and left, respectively. In addition, a double-swipe up switches to symbols and numbers.
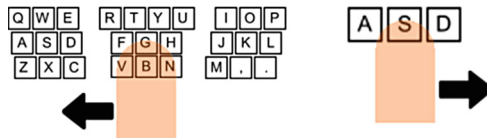

Fig. 3. The Swipeboard. In this picture, the user first performs a left swipe to select one of the nine regions, and then a right swipe to select the character 'D'.

A smart eyewear adaptation of Swipeboard, called **SwipeZone**, slightly modifies the QWERTY layout and divides it into three regions, each containing three rows of keys [14]. To enter a character, the user first swipes vertically on a region to select one of its three rows, and then swipes horizontally towards the direction where the intended character is located. In a user study with Google Glass, SwipeZone reached up to 8.73 WPM with on average 24.9 % ER, including hard and soft errors.

Fig. 4. The ZoomBoard. When the user taps on the miniature QWERTY keyboard, it iteratively magnifies the touched region. The user then can tap on a particular key to enter the corresponding character. The keyboard goes back to its original state immediately after that.

ZoomBoard. ZoomBoard displays a miniature standard QWERTY on the screen [26]. To enter a character, the user roughly taps on the region where it is located, and the system iteratively magnifies the region until the keys are large enough for the user to select (Fig. 4). The user then enters the character by tapping on the corresponding key. The keyboard transforms back to its original state immediately after entering a character. ZoomBoard also enables space and backspace entry through a left and a right swipe on the keyboard, respectively. In addition, an up swipe switches the keyboard to symbols and numbers.

Callout and ZShift. Both Callout and ZShift are inspired by the callout feature of many modern virtual keyboards for smartphones. When a user touches a region of a miniature QWERTY keyboard, the Callout technique [22] displays a callout containing the currently selected character above the keyboard (Fig. 5). The user then can refine the selection by slightly moving the finger, and when satisfied, enter the character by lifting up the finger. One disadvantage of this technique is that the user has to rely on his/her spatial memory when refining a selection, as the fingertip usually covers most of the keyboard. ZShift [22] addresses this issue by showing a magnified version of the occluded region in the callout. It provides the user with visual feedback on the currently selected character by highlighting it in the callout, illustrated in Fig. 5, right.
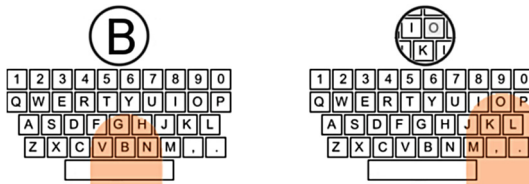

Fig. 5. The Callout and ZShift, respectively. The former displays the currently selected character in the callout, while the latter displays a magnified version of the occluded region and highlights the currently selected character. With both techniques, the user can refine the selection by slightly moving the finger, and enter a selected character by lifting the finger.

SlideBoard. SlideBoard [17] consists of fifteen keys laid out in a 5 × 3 grid, each containing two characters (Fig. 6). With this technique, the user swipes right on a key to enter the right character and swipes left to enter the left character. The enter, space, and backspace keys are located at the bottom of the screen, and can be selected by touching the bezel.

DualKey. DualKey [15] uses a very similar keyboard template as SlideBoard [17], see Fig. 6, but leverages the distinction between the index and middle fingers to enable single tap character entry. In DualKey, the first fourteen keys contain two characters,
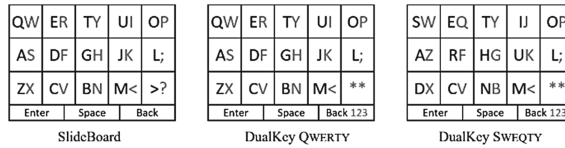
Fig. 6. The SlideBoard, DualKey QWERTY, and SWEQTY. With SlideBoard, the user swipes left or right on a key to enter the left or right character, respectively. In DualKey, the left and right characters associate with the index and middle fingers, respectively. Therefore, the user uses the index finger to enter the left character and middle finger to enter the right. The '**' key enables swapping a character with its same-key counterpart. SWEQTY is an optimized layout for DualKey that reduces the time required between two taps and the total finger switching instances.

where the right character associates with the middle and the left associates with the index finger. The '**' key enables swapping a character with its same-key counterpart, e.g., tapping on the key immediately after entering a 'Q' will replace it with a 'W'. The enter, space, and backspace keys are located at the bottom of the screen. A middle finger tap on the backspace switches the keyboard to symbols and numbers.

The SWEQTY layout attempts to increase the performance of DualKey by reducing the time and finger switching instances between subsequent taps. It deliberately maintains a closeness to QWERTY to accommodate faster learning (Fig. 6).

Fig. 7. Screenshots of Fleksy and Minuum, respectively. Both are predictive techniques that correct and disambiguate input at both character and word levels based on the sequence of keys pressed.

Fleksy. Fleksy is a commercial predictive keyboard, available for several touchscreen-based devices [10], including smartwatches [21]. Its predictive system autocorrects the entry at character-level as the user types based on the previous input and context. Fleksy also enables word prediction and autocorrection. In case of an incorrect autocorrection, the user can swipe down anywhere on the screen to see alternative suggestions. A long press on the screen enables symbols and number entry. The user can also delete one word at a time by swiping left on the screen. Figure 7 illustrates the technique.

Minuum. Minuum is a commercial predictive keyboard, originally designed for tablets [23], but can be used on various touchscreen devices, including smartwatches [33]. It condenses the three rows of keys in the standard QWERTY layout into a single line. The

system disambiguates the input based on the sequence of keys pressed. It also includes an extra line for symbols and numbers (Fig. 7). Minuum also supports gestures. A right swipe on the screen enters a space, a left swipe deletes a full word, and two right swipes changes to symbols. There is no empirical evaluation available for Minuum on smartwatches.

Fig. 8. Screenshots of Swype and WatchWriter [13], respectively. Both are predictive techniques that enable gesture typing. The user enters one character per tap or one word per gesture. The traces in the picture indicate the gestures drawn to enter 'Swype' and 'please', respectively.

Swype. Swype is a commercial predictive keyboard, designed mainly for smartphones, that supports both touch and gesture typing [31]. With Swype, the user enters either one character per tap or a word per gesture (Fig. 8). It features a suggestion bar that displays the best predictions based on the preceding input and context. The user accepts a prediction by either tapping on the prediction bar or the space key. When a prediction is selected, the system automatically enters a space following the word. Further, Swype automatically corrects all likely incorrect words. Although not optimized for ultra-small devices, Swype has been evaluated on a smartwatch [6].

WatchWriter. WatchWriter supports both touch and gesture typing on smartwatches [13]. Similar to modern gesture keyboards, the user enters either a character per tap or a word per gesture. It also features a suggestion bar (Fig. 8) that displays the two best predictions based on a language model during gesture typing. During tap typing, the bold suggestion on the left displays the best prediction and the right suggestion displays the literal string. If the most likely prediction matches the literal string, the left suggestion displays the second most likely prediction. The user can accept a prediction by tapping on it, which also enters a space following the word. The backspace key is located beside the prediction bar. It operates at a word-by-word level, that is, deletes one word per tap.

3   Novel Keyboard Layouts

Many have also proposed novel text entry techniques for smartwatches. Most of these techniques map multiple characters onto a single key to account for the smaller screens and use different strategies to disambiguate an ambiguous entry. This section reviews all these techniques.
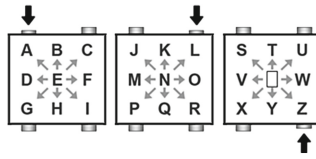
Fig. 9. The TiltType. To enter a character, the user tilts the device and presses one or more physical buttons. A character is entered based on the button(s) pressed and the direction and angle of the tilt.

TiltType. TiltType is a novel text entry technique for wristwatches [27] that utilizes four physical buttons (two above and two below the device) and eight compass directions for text entry (Fig. 9). It assigns all letters alphabetically and the space character to three different views that the user selects by pressing the top two and the bottom right physical buttons, respectively. To enter a character, the user tilts the device towards the direction where the character is located, and then presses the button respecting the view containing the character. Leveling the device selects the character in the center position. The user can refine the selection when holding the button by changing the tilt direction and angle. The fourth button (bottom left) is used for backspace and other special features. Pressing it without tilting enters a backspace, while tilting the device in different directions and angles enters uppercase characters, numbers, and symbols. TiltType requires two hands to operate, thus devices using this method must be easily removable from its wrist-strap. Unfortunately, there is no empirical evaluation available for the technique.

DragKeys. DragKeys [8], also known as Tipckle [32], consists of an array of circularly arranged keys that continuously follows the text cursor. It has two levels of key arrays, where each array contains multiple keys, and each key contains multiple characters (Fig. 10). To enter a character, first the user drags an ambiguous key containing multiple letters to the cursor. This loads the second-level non-ambiguous keys. The user then drags a non-ambiguous key to the cursor to enter the corresponding character. Skipping the second step enters the character in the center of the first-level key. The most frequently used characters are placed in that position, so that they can be entered with a single stroke. Unfortunately, there is no empirical evaluation available for the technique.
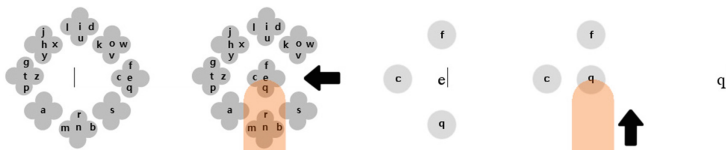

Fig. 10. The DragKeys. To input the letter 'Q', the user drags the right most key to the cursor to see the second-level keys containing the letters from the dragged key, and then drags 'Q' key to the cursor to input it. Skipping the second step enters the letter in the center 'E'.

QWERTY-like Keypad (QLKP). QLKP consists of nine keys laid out in a 3 × 3 grid, each containing multiple characters [18]. The keypad places the left characters of the QWERTY on the left column and the right characters on the right column to maintain a resemblance to the standard QWERTY layout (Fig. 11). Similar to Multi-tap [3], to enter a character with this technique the user taps on a key repeatedly until he/she gets the intended character. Although primarily designed for feature phones, this technique has been evaluated on a smartwatch [17]. The smartwatch version includes the enter, space, and backspace keys at the bottom of the screen that can be selected by touching the bezel.

AB CD	EFG HIJ	KL MN
Entry|		
OP QRS	T UV	WX YZ

EWQ	TFY	OP
ADZ	RGV	ILJ
SCX	HUB	NMK
Enter	Space	Back

Fig. 11. The Optimized Alphabetic Layout (OAL) and QWERTY-like Keypad (QLKP), respectively. To enter a character with OAL, the user taps on the ambiguous keys and the system disambiguates the input based on the tap sequence. With QLKP, similar to Multi-tap on a standard 12-key keypad, the user taps on a key repeatedly until he/she gets the intended character.

Optimized Alphabetic Layout (OAL). OAL consists of six large ambiguous keys, three above and three below the input area [20]. The layout maps the letters onto the keys in alphabetic order (Fig. 11), but methodically splits them to reduce ambiguity errors and subsequent target distances. The keyboard uses a predictive system to disambiguate the input, that is, predicts the intended character based on tap sequences. It also suggests word completion that the user can accept by swiping right on the screen. A first tap on the central area enters a space, while the subsequent taps rotate through alternative suggestions that match the ambiguous entry. Similarly, a left swipe enters a backspace and a down swipe switches the layout to symbols and numbers.

UniWatch. UniWatch [29] is the smartwatch variant of a mobile text entry technique called UniGlyph [28]. It categorizes all characters into three groups based on the primary shape they are composed of. All characters that contain diagonal strokes are categorized as 'diagonal' characters, all other characters that contain loops or curves are categorized as 'curve' characters, and the remaining characters are categorized as 'line' characters. Accordingly, the UniWatch template consists of three keys, representing the three shapes (Fig. 12). To enter a character, the user taps on the key that represents its primary shape. As these shapes are shared between multiple characters, the technique disambiguates the input based on the sequence of keys pressed. Currently there is no empirical evaluation available for this technique. Relevantly, a text entry technique for smartphones, called UOIT, also exploits the shapes of the characters [1].

Table 1. Performances of text entry techniques for smartwatches from the literature. A "*" signifies results from a simulation.

Technique	Reference	Participant	Device	Diagonal Display mm	Keyboard Size mm	Key Size mm	Entry Speed WPM	Error Rate %	KSPC
NON-PREDICTIVE QWERTY									
QWERTY	16	18	Samsung Galaxy Gear	41.44	29.3×21.3	1.6×3.0	9.74	28.3^TER	-
					22.9×16.6	2.3×4.1	12.12	19.2^TER	-
					16.5×11.2	2.9×5.3	14.42	19.9^TER	-
	17	12	Samsung Galaxy Gear	41.44	-	2.9×5.3	12.90	21.4^TER	-
Virtual Sliding QWERTY (VSQ)	5	20	Microsoft Surface Pro 3	40.0	40.0×21.0	3.0×3.0	10.76	-	1.05
					50.0×26.5	4.0×4.0	11.66	-	1.03
					60.0×31.0	5.0×5.0	11.26	-	1.03
					70.0×36.5	6.0×6.0	10.64	-	1.02
					80.0×41.0	7.0×7.0	10.04	-	1.02
SplitBoard	17	24	Samsung Galaxy Gear	41.44	-	4.8×6.5	14.75	7.5^TER	-
	16	18	Samsung Galaxy Gear	41.44	29.3×21.3	2.7×3.7	10.66	13.9^TER	-
					22.9×16.6	3.8×5.1	11.67	11.1^TER	-
					16.5×11.2	4.8×6.5	15.07	7.9^TER	-
Swipeboard	7	8	Apple iPad 3	-	12.0×12.0	1.5×1.5	19.58	17.5^ER	-
ZoomBoard	26	6	Apple iPad 3	-	16.5×6.1	1.5×1.5	9.30	-	2.15
	16	18	Samsung Galaxy Gear	41.44	29.3×21.3	1.6×1.6	8.02	10.1^TER	-
					22.9×16.6	2.3×2.3	9.09	6.7^TER	-
					16.5×11.2	2.9×2.9	9.26	7.3^TER	-
	17	12	Samsung Galaxy Gear	41.44	-	2.9×2.9	9.20	7.1^TER	1.85*
	22	20	Samsung Nexus S	25.46	16.0×6.5	1.5×1.5	6.00	15.3^TER	2.7
				33.94	21.3×8.6	2.0×2.0	7.80	8.0^TER	2.2
				45.25	28.4×11.4	2.6×2.6	8.20	7.3^TER	2.1
Callout	7	8	Apple iPad 3	-	16.5×6.1	1.5×1.5	17.08	19.6^ER	-
	22	20	Samsung Nexus S	25.46	16.0×6.5	1.5×1.5	4.30	19.9^TER	1.8
				33.94	21.3×8.6	2.0×2.0	7.10	14.8^TER	1.5
				45.25	28.4×11.4	2.6×2.6	8.30	12.4^TER	1.4
ZShift	22	20	Samsung Nexus S	25.46	16.0×6.5	1.5×1.5	5.40	15.4^TER	1.5
				33.94	21.3×8.6	2.0×2.0	7.20	13.9^TER	1.4
				45.25	28.4×11.4	2.6×2.6	9.10	12.3^TER	1.3
SlideBoard	17	12	Samsung Galaxy Gear	41.44	-	5.6×6.5	12.08	7.9^TER	1.85*
DualKey^QWERTY	15	10	LG G Watch	41.86	-	5.6×6.5	19.61	5.3^TER	-
PREDICTIVE QWERTY									
Fleksy	6	18	Samsung Galaxy Gear	41.44	-	5.66×-	20.3	16.0^TER	-
Swype	6	18	Samsung Galaxy Gear	41.44	-	4.10×-	29.3	9.0^TER	-
WatchWriter^Gesture	13	18	LG G Watch R	33.02	-	-	24.0	3.7^CER	0.4
WatchWriter^Tap	13	18	LG G Watch R	33.02	-	-	22.0	1.5^CER	1.5
NOVEL NON-PREDICTIVE									
QLKP	17	12	Samsung Galaxy Gear	41.44	-	9.5×6.4	9.20	4.3^TER	1.53*
DualKey^SWEQTY	15	8	LG G Watch	41.86	-	5.6×6.5	21.59	3.27^TER	-
NOVEL PREDICTIVE									
OAL	20	20	Sony Smart-Watch 2	39.05	-	-	8.08	~5.7^CER	-
HANDWRITING RECOGNITION									
EdgeWrite	9	5	Breadboard	58.6	-	-	3.9	6.4^TER	-

ABCDEFGHIJKLMNO
PQRSTUVWXYZ
$\boxed{\lor}\ \boxed{()}\ \boxed{-|}$

Fig. 12. UniWatch consists of three keys, representing the 'diagonal', 'curve', and 'line' shapes. The dark parts of the letters (above) signify their primary shapes. To enter a letter, the user taps on the key that represents its primary shape, the technique disambiguates the input.

4   Handwriting Recognition

Researchers are also exploring handwriting recognition for text entry on smartwatches. Unlike virtual keyboards, where many keys share a small screen, handwriting can offer most of the screen for each character, allowing much more comfortable character entry [25]. In addition, prior investigations showed that some handwriting systems can be used without looking at the screen [11], enabling eyes-free text entry.

EdgeWrite. EdgeWrite is a unistroke-based technique for users with motor impairments [34]. Unlike natural handwriting, unistroke-based techniques limit user behaviors by allowing only a single way of drawing each character to avoid segmentation and other handwriting recognition related problems [4]. The EdgeWrite alphabet maintains a resemblance to its printed counterpart to maximize the user's ability to guess (Fig. 13). It requires the user to input characters by traversing the edges and diagonals of a square screen. Then a gesture is recognized not through patterns, but based on the sequence of corners that are hit. Recently, this technique has been evaluated in the context of a smartwatch [9].
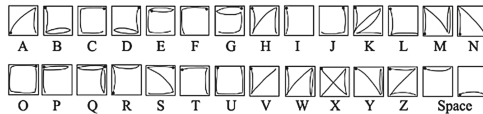
A B C D E F G H I J K L M N
O P Q R S T U V W X Y Z Space

Fig. 13. EdgeWrite unistroke gesture alphabet. Here, a dot represents the start point of a stroke.

Analog Keyboard. Analog Keyboard [25] enables natural handwriting on smartwatches. With this technique, the user writes one character at a time on the screen using a finger. The system then recognizes the character, including digits and symbols, and inputs it. The keyboard also includes two narrow buttons in the left and right sides of the screen for backspace and space, respectively. Unfortunately, there is no empirical evaluation available for the technique.

5   Results and Discussion

Results suggest that predictive techniques perform relatively better than non-predictive techniques, both in terms of speed and accuracy. This is not surprising, considering users usually make more mistakes when typing on smaller screens, most of which

predictive techniques can automatically correct. This improves the overall performance by reducing errors and error correction efforts. Most predictive techniques yielded over 20 WPM in empirical evaluations, while entry speeds for non-predictive techniques ranged from 4 to 22 WPM. Similarly, the lowest reported error rate was about 2 % with WatchWriter, a predictive technique. Error rates for non-predictive techniques ranged from 4 to 28 %. EdgeWrite yielded the lowest entry speed (4 WPM), which is not surprising considering it was designed for users with motor impairments.

To assist precise selection of smaller keys, most QWERTY-based techniques break up each key selection into a multi-step operation. However, the results do not indicate an immediate benefit of this approach. In studies, these techniques yielded on average 10 WPM, ranging from 4 to 20 WPM, while miniature QWERTY keyboards yielded on average 12 WPM. Both Swipeboard and ZoomBoard yielded noticeably better entry speed than QWERTY, roughly 20 and 17 WPM, respectively. However, the fact that these techniques were evaluated in longitudinal studies and on an Apple iPad may have contributed towards this. In other studies, ZoomBoard yielded on average 9 WPM. Hence, performances with these techniques may improve with practice. Error rates were mostly comparable between all techniques (~ 12 %). Some predictive techniques were relatively more accurate, but did not account for in-vocabulary errors in the studies, which would have increased their error rates. This suggests that these techniques are error prone, therefore demand extra correction efforts. This highlights the need for effective error correction methods for smartwatches.

Although some studies found significant effects of keyboard and key sizes on text entry performance [16], this survey failed to find a clear indication of this. This is most likely because performances do not differ substantially when keyboard and key sizes are within a certain range. Results of a prior study also support this assumption [22].

Interestingly, many are exploring techniques for feature phones on smartwatches, as these techniques also attempt to map all letters, digits, and symbols onto a smaller area [34]. Prior work in mobile text entry left a rich body of work, thus further investigations are necessary to fully understand whether (and how) these techniques, or modified versions of them, can be used on smartwatches. Further, most current techniques are designed for square-faced smartwatches. The support for round-faced devices is also important, as they are becoming increasingly popular among users [30]. Although modified versions of these techniques may function on round devices, thorough investigation is necessary to determine how that would impact their performances. Moreover, none of the current systems, apart from handwriting, explore eyes-free text entry. With eyes-free text entry users can reach their maximum entry speed, hence considered as the final step of the novice-expert transition [3]. This can also increase the usability of smartwatches. Thus, further investigation is necessary to design and develop methods that enable touch typing.

6 Conclusion

This document reviewed the most important text entry techniques for smartwatches. It categorized all techniques based on whether they use (a variant of) QWERTY, a novel keypad or keyboard, or a handwriting system. It discussed the design and motivation

for all current techniques and presented their performances from the literature, both in terms of speed and accuracy, in a table. Finally, it concluded with a discussion of the remaining challenges and future possibilities in the area.

References

1. AbuHmed, T., Lee, K., Nyang, D.: UOIT keyboard: a constructive keyboard for small touchscreen devices. IEEE Trans. Hum.-Mach. Syst. **45**(6), 782–789 (2015)
2. Arif, A.S., Sturzlinger, W.: Analysis of text entry performance metrics. In: IEEE Toronto International Conference—Science and Technology for Humanity, pp. 100–105. IEEE Press, Washington (2009)
3. Arif, A.S.: Predicting and Reducing the Impact of Errors in Character-Based Text Entry. York University, Toronto (2013)
4. Buxton, W.A.S.: Touch, gesture, and marking. In: Baecker, R.M., Grudin, J., Buxton, W., Greenberg, S. (eds.) Readings in Human Computer Interaction: Toward the Year 2000, pp. 469–524. Morgan Kaufmann, San Francisco (1995)
5. Cha, J.M., Choi, E., Lim, J.: Virtual sliding QWERTY: a new text entry method for smartwatches using Tap-N-Drag. Appl. Ergon. **51**, 263–272 (2015)
6. Chaparro, B.S., He, J., Turner, C., Turner, K.: Is touch-based text input practical for a smartwatch? In: Stephanidis, C. (ed.) 17th International Conference on Human-Computer Interaction. Posters' Extended Abstracts, pp. 3–8. Springer, Heidelberg (2015)
7. Chen, X.A., Grossman, T., Fitzmaurice, G.: Swipeboard: a text entry technique for ultra-small interfaces that supports novice to expert transitions. In: 27th Annual ACM Symposium on User Interface Software and Technology, pp. 615–620. ACM Press, New York (2014)
8. Cho, H., Kim, M., Seo, K.: A text entry technique for wrist-worn watches with tiny touchscreens. In: Adjunct Publication of the 27th Annual ACM Symposium on User Interface Software and Technology, pp. 79–80. ACM Press, New York (2014)
9. Darbar, R., Sen, P.K., Dash, P., Samanta, D.: Using hall effect sensors for 3D space text entry on smartwatches. In: 7th International Conference on Intelligent Human Computer Interaction. Elsevier, Philadelphia (2015)
10. Fleksy Keyboard. https://fleksy.com
11. Goldberg, D., Richardson, C.: Touch-typing with a stylus. In: INTERACT 1993 and CHI 1993 Conference on Human Factors in Computing Systems, pp. 80–87. ACM Press, New York (1993)
12. Gong, J., Tarasewich, P.: Alphabetically constrained keypad designs for text entry on mobile devices. In: SIGCHI Conference on Human Factors in Computing Systems, pp. 211–220. ACM Press, New York (2005)
13. Gordon, M., Ouyang, T., Zhai, S.: WatchWriter: tap and gesture typing on a smartwatch miniature keyboard with statistical decoding. In: 34th Annual ACM Conference on Human Factors in Computing Systems. ACM Press, New York (2016)
14. Grossman, T., Chen, X.A., Fitzmaurice, G.: Typing on glasses: adapting text entry to smart eyewear. In: 17th International Conference on Human-Computer Interaction with Mobile Devices and Services, pp. 144–152. ACM Press, New York (2015)
15. Gupta, A., Balakrishnan, R.: DualKey: miniature screen text entry via finger identification. In: 34th Annual ACM Conference on Human Factors in Computing Systems. ACM Press, New York (2016)

16. Hong, J., Heo, S., Isokoski, P., Lee, G.: Comparison of three QWERTY keyboards for a smartwatch. In: Interacting with Computers (2016)
17. Hong, J., Heo, S., Isokoski, P., Lee, G.: SplitBoard: a simple split soft keyboard for wristwatch-sized touch screens. In: 33rd Annual ACM Conference on Human Factors in Computing Systems, pp. 1233–1236. ACM Press, New York (2015)
18. Hwang, S., Lee, G.: Qwerty-like 3x4 keypad layouts for mobile phone. In: CHI 2005 Extended Abstracts on Human Factors in Computing Systems, pp. 1479–1482. ACM Press, New York (2005)
19. IC Insights.: Research Bulletin: Wearable Systems Give Major Boost to Total IoT Sales in 2015 (2015). http://www.icinsights.com/news/bulletins/Wearable-Systems-Give-Major-Boost-To-Total-IoT-Sales-In-2015
20. Komninos, A., Dunlop, M.: Text input on a smart watch. IEEE Pervasive Comput. **13**(4), 50–58 (2014)
21. Lee, N.: Fleksy Keyboard Brings Predictive Touch Typing to the Galaxy Gear. Engadget (2014). http://www.engadget.com/2014/01/09/fleksy-galaxy-gear
22. Leiva, L.A., Sahami, A., Catala, A., Henze, N., Schmidt, A.: Text entry on tiny QWERTY soft keyboards. In: 33rd Annual ACM Conference on Human Factors in Computing Systems, pp. 669–678. ACM Press, New York (2015)
23. Li, F.C.Y., Guy, R.T., Yatani, K., Truong, K.N.: The 1line keyboard: a QWERTY layout in a single line. In: 24th annual ACM Symposium on User Interface Software and Technology, pp. 461–470. ACM Press, New York (2011)
24. MacKenzie, I.S., Soukoreff, R.W.: Phrase sets for evaluating text entry techniques. In: CHI 2003 Extended Abstracts on Human Factors in Computing Systems, pp. 754–755. ACM Press, New York (2003)
25. Microsoft Research: The Analog Keyboard Project. http://research.microsoft.com/en-us/um/redmond/projects/analogkeyboard
26. Oney, S., Harrison, C., Ogan, A., Wiese, J.: ZoomBoard: a diminutive qwerty soft keyboard using iterative zooming for ultra-small devices. In: 31st Annual ACM Conference on Human Factors in Computing Systems, pp. 2799–2802. ACM Press, New York (2013)
27. Partridge, K., Chatterjee, S., Sazawal, V., Borriello, G., Want, R.: TiltType: accelerometer-supported text entry for very small devices. In: 15th Annual ACM Symposium on User Interface Software and Technology, pp. 201–204. ACM Press, New York (2002)
28. Poirier, F., Belatar, M.: UniGlyph: only one keystroke per character on a 4-button minimal keypad for key-based text entry. In: 9th International Conference on Human-Computer Interaction, pp. 479–483. Springer, Heidelberg (2007)
29. Poirier, F., Belatar, M.: UniWatch - some approaches derived from UniGlyph to allow text input on tiny devices such as connected watches. In: Kurosu, M. (ed.) Human-Computer Interaction. LNCS, vol. 9170, pp. 554–562. Springer, Heidelberg (2015)
30. Sung, D.: Round v Square Faced Smartwatches: We Ask the Experts which is Best. Wareable (2015). http://www.wareable.com/smartwatches/round-v-square-smartwatches-which-is-best
31. Swype. http://www.swype.com
32. Tipckle. https://play.google.com/store/apps/details?id=esrc.korea.ac.kr.tipckle_ime
33. Typing On a Watch. http://minuum.com/minuum-on-smartwatch
34. Wobbrock, J.O., Myers, B.A., Kembel, J.A.: EdgeWrite: a stylus-based text entry method designed for high accuracy and stability of motion. In: 16th Annual ACM Symposium on User Interface Software and Technology, pp. 61–70. ACM Press, New York (2003)