Published in: IEEE TRANSACTIONS ON HUMAN-MACHINE SYSTEMS

The Version of Record is available online at: http://dx.doi.org/10.1109/THMS.2017.2754938

Handwriting on Smartwatches: An Empirical Investigation

Gennaro Costagliola, *Member, IEEE,* Mattia De Rosa, and Vittorio Fuccella

***Abstract*—The results of an experimental study on handwriting on smartwatches are described in this study. This study aims to gain knowledge on the current practice and possible future developments on smartwatches. This includes two experiments: the first experiment aims to compare a state-of-art commercial handwriting system to other competing text entry methods based on soft keyboards, while the other experiment allows us to gather some useful information regarding users' performance and preference in an ideal setting (without character recognition), so as to direct the design of future handwriting applications on small devices. In the first experiment, we obtained an initial entry speed of 7.5 wpm, lower than that of the other compared methods. The performance obtained in the second experiment, in the range of 15-19 wpm, suggests that, by appropriately changing some settings (e.g. improved recognition and feedback, preferred writing style and spare of space characters), the entry speed can be significantly increased. The results also showed that users prefer to enter upper case characters and that, although this style of writing requires a significantly greater number of gestures per character, it does not adversely affect the writing speed.**

***Index Terms*—handwriting; smartwatch; wristwatch; wearable devices; text entry; soft keyboard;**

I. INTRODUCTION

Efficient text entry on small touchscreens is challenging. This is especially true on wearable devices as smartwatches. In fact, some models do not even include any soft keyboard among the pre-installed system programs, but only a voice recognition application as a text input method. The latter method, however, has numerous and imaginable disadvantages, e.g. related to the privacy of the users. For this reason, the recent HCI literature contains several attempts to bring adapted versions of soft keyboards on smartwatches [1], [2], [3].

Handwriting is a very natural method for entering text on electronic devices. Furthermore, it has the advantage of exploiting a knowledge acquired by most users in primary school. Another advantage is that it does not require a total commitment of vision. In fact, experiments [4] showed that it is possible to write with good accuracy even without vision. This can make handwriting preferable to soft keyboards in many contexts. Nevertheless, it has been traditionally considered slow, although recent studies [5] revised upwards its performance (up to about 25 wpm after some hours of practice). The above speed may be considered acceptable if obtained on devices where size does not allow users to do much better, e.g. smartwatches.

The study presented in this article is aimed at understanding the feasibility of handwriting on smartwatches. On these devices, the small display size only allows the user to input a single character at a time, superimposed on the previous one. Nevertheless, unlike some similar techniques, which were based on unistroke alphabets, used in the past [6], we let the participants use their preferred number of gestures per character in our tests.

We performed two different experiments in which we tested the character by character handwriting technique in two different settings. The first experiment is intended to know whether the current handwriting systems are appreciated by the users and can compete in performance with other existing text entry methods. Thus, we compared a state-of-art commercial handwriting system to other methods, which can fit well in a very small screen. In particular, we choose two different types of soft keyboard-based methods: T9 and *Splitboard* [3]. These two methods were compared using two different display dimensions in order to test them with the most common sizes adopted in commercial devices.

Aware of the limitations of the real system tested in the first experiment, we decided to complete our study on handwriting with a second, exploratory experiment, which aims to investigate the feasibility and potentiality of handwriting in ideal settings. In this experiment, we did not choose a specific text recognizer and gave no feedback to the users, to avoid linking the results of the experiment to the performance of a particular software system. Thus, the speed detected in this experiment should be intended as an upper bound to the one obtainable through a real system performing character recognition, i.e., the speed obtainable by a *perfect* recognizer (which also forgives small writing errors) and the best feedback system. Another objective of our second experiment is the understanding of users' preferences of writing. We adopted the writing style (uppercase, lowercase, or unconstrained) as an independent variable. Furthermore, given the well-established use of dictionaries to improve text entry recognition and the recent trend to tolerate the lack of spaces in text input (on soft keyboards) [7], we tested the possibility of omitting the input of the space.

What we measure in both experiments is the initial *pickup-and-use* speed [8]. That is, we do not investigate the learning of the method or the speed reached after some practice. The importance of the initial performance on smartwatches is stressed by Hong et al. [9]: "smartwatch users are not expected to do massive text entry on a smartwatch, and therefore, it is difficult to expect that they will invest time to learn a new text entry skill".

The authors are with the Department of Informatics, University of Salerno, Fisciano (SA), Italy.

E-mail: {gencos, matderosa, vfuccella}@unisa.it.

The next section describes some work related to ours. Sections III and IV describe our two experiments and their results. A discussion on the limitation of this study and some comments on future work conclude the paper.

II. RELATED WORK

In this section, we include two short surveys on handwriting on touchscreens and text entry methods on small devices. Our surveys are not exhaustive. For a survey on handwriting recognition, the reader may refer to [10]. For more extensive surveys on text entry, the reader may refer to [11], [12].

A. Handwriting on Touchscreens

Handwriting has been traditionally considered slow. As reported by Kristensson and Denby [5], many sources came to this conclusion based on the results reported by an experiment conducted by Devoe in the 60's [13] that attributed handwriting an average speed of about 16 wpm. Recognition difficulties, together with the belief of its inefficiency, determined the limited use of handwriting on various devices.

In the 90's, with the spread of the first handheld devices equipped with touchscreen, unistroke alphabets were introduced. They allowed the entry of a character with a single stroke, so to get around the problem of segmentation. We can cite *Unistrokes* [6] and *Graffiti* [14] among them. A comparison between them was performed by Castellucci and MacKenzie [15]. Their results showed very low initial rates, around 4 wpm, for both methods. After some hours of learning, Unistrokes prove to be faster, reaching an average speed of about 16 wpm. Another example of unistroke writing, designed to improve writing accuracy for people with motor impairments, is *Edgewrite* [16]. To improve stability, a stroke is performed by traversing the edges and diagonals of a square hole imposed over the text input area

Discrete printing also simplifies segmentation and relaxes the constraint of having to write each character with a single stroke. The text is entered in a grid and each character occupies a single cell. MacKenzie and Chang [17], comparing two commercial handwriting recognizers with discrete hand-printed characters, recorded entry speed ranging from 16.7 wpm to 17.6 wpm.

Recently, Kristensson and Denby's experiments [5] demonstrated that the efficiency of unconstrained handwriting on mobile devices had been underestimated. *Unconstrained* means that the recognizer simultaneously accepts hand-printed characters, cursive script, and a combination of both. After 250 minutes of practice, their participants had a mean text entry rate of about 24 wpm.

B. Text Entry on Small Devices

The reduced size of mobile devices only allows their keyboards to contain a reduced number of keys, which are often lower than the number of existing alphabetic characters. Therefore, ambiguous keyboards were introduced. In such keyboards, multiple characters are assigned to the same key. A very common arrangement groups the characters into 3 or 4 and assigns them in alphabetical order. Other arrangements, such as Less-tap [18] and the QWERTY-like keypad [19] tried to provide users with more familiar layouts.

The ambiguous keyboards need a disambiguation method to select the desired character upon a key press. A baseline method is multi-tap, in which the character is selected based on the number of consecutive key presses. This method is not very efficient, since it requires an average number of more than two key presses per character [20]. The most widespread method was undoubtedly the T9 [21], where the keys can be pressed once and the disambiguation is performed based on a dictionary with words sorted by their frequency in the target language. A key is dedicated to the selection of candidate words. An alternative method, which disambiguates the characters based on those previously entered is LetterWise [22], which requires a smaller amount of memory.

Soft keyboards spread with the adoption of touchscreens on mobile devices. Their "malleability" enabled the test of optimized layouts [23], [24], [25], [26] alternative to QWERTY. Despite their better performance, some reluctance to abandon the QWERTY was observed probably because the users tend to minimize their learning effort [8]. In parallel with layout optimization, some methods, which enabled the entry of a word through a single stroke, were proposed. The first examples, e.g., *Cirrin* [27] and *Quikwriting* [28], required the use of a special layout. Later on, starting from the proposal of Kristensson and Zhai [29], it was possible to use gestural input directly on the QWERTY keyboard to enter whole words [30] or parts of them [31].

The adaptation of soft keyboards to very small displays, as those of smartwatches, at first led researchers to design methods showing only a part of the keyboard at a time, letting the user access parts mostly through zoom and pan operations. An example of this sort of adaptation is *Zoomboard* [1], which requires two taps for a character entry: the first one for zooming on the keyboard area where the character is located, the second one to enter it; Leiva et al. [32] proposed and compared to Zoomboard alternative designs based on callouts and shift techniques. In *Splitboard* [3], the keyboard is horizontally split in two halves displayed one at a time. The user can switch between them by using a right or left flick. The experiments performed with these methods, however, showed initial speeds of writing less than or equal to 15 wpm [3], [9].

However, in more recent times, with the improvement of the interpretation of user typing through the optimization of language models, it was possible to effectively use the full layout even on small displays [7]. Thanks to the use of a large sized display, Gordon et al. [33] were able to successfully test both tapping and gesture writing on a smartwatch with a 1.3 in circular display.

Among the layouts alternative to QWERTY tested on wristwatches, we can mention the method described in [2], which uses an ambiguous keyboard composed of six keys and a central region to enter a space and to show the currently entered text.

III. EXPERIMENT I: HANDWRITING VS SOFT KEYBOARDS

In this section, we analyze the performance of a state-of-art handwriting system in comparison to two different text entry methods based on soft keyboards:

- an ambiguous keyboard with a reduced number of keys enhanced with a dictionary (the well-known *T9* method);
- an adaptation to small screens of a keyboard with the QWERTY layout, i.e. the *SplitBoard* method described in [3].

The latter is one of the best performing [3] of the recently introduced soft-keyboards with QWERTY layout adapted to very small touchscreens. Ambiguous keyboards were used in the past with mobile phones. Although these methods became outdated with the spread of the latest smartphones models, we included one of them in the experiment because a reduced keyboard is suitable for a wristwatch-sized display. T9 was selected in a preliminary experiment where we compared it to other methods based on 12 keys keyboards (i.e., *Multitap* and *TiltText* [34]) adapted to the smartwatch.

A. Methods

1) Participants: We recruited 24 participants (8 female). All of them were university students who agreed to participate for free. All were habitual computer and touchscreen users. Their age was in the range 22 - 34 ($M = 26.1$, $SD = 2.7$). All participants had a good experience with T9 on old phone models, four of them had a little experience with handwriting on touchscreen.

2) Apparatus: The experiment was carried out on a *Sony SmartWatch 3 SWR50* equipped with a *Quad ARM A7, 1.2 Ghz* processor and running the *Android Wear* operating system. The device weighs 45 grams and has a square display of 1.6 in (28.7 mm wide) with 320×320 pixel resolution.

The experimental software was developed in Java and was divided into four modules, one front-end module for each input method and one back-end module for gathering text entry metrics and producing logs. The back-end library was implemented as a Java library. Each of the three modules for text entry was an independent Android Activity having its own canvas for presenting and transcribing text. They were the following:

- *Handwriting Module*: module implementing the handwriting system. The interface, shown in Figure 1a, is divided into two parts. The upper part, with a black background, shows the presented text (in gray) and the transcribed text (in white). The rest of the screen is in lighter gray. The whole surface of the device can be used to enter characters. This module is based on the *MyScript* API[1]. The API enables character recognition in real time. Furthermore, it supports a form of dictionary-based spell-check and correction. We adopted the convention of using two consecutive 'enter' characters to submit a phrase and load the next one.
- *T9 Module*: our implementation for the T9 method. The method disambiguates the words by proposing the word with highest frequency among possible candidates (words matching the sequence of pressed keys) in the dictionary. Our dictionary contains about 30K word-frequency pairs. The sharp ('#') key is used to show the next candidate word. The interface is shown in Figure 1b. The upper part of the view shows the presented and the transcribed text. A phrase is submitted by pressing the 'enter' key twice.
- *SplitBoard module*: implementation (courtesy of Jonggi Hong) of the *Splitboard* method [3]. Its interface is shown in Figure 1c. It is worth noting that this prototype does not use a dictionary nor a language model to improve input accuracy.

The interfaces of the methods were displayed at two different display sizes, one using the full screen at 320×320 resolution, and the other using a 280×280 central area of the screen. The above resolutions were chosen to represent very common medium (1.6 in) and small (1.4 in) sized smartwatch displays.

3) Procedure: Before starting the experiment, participants had an induction phase where the objective and the procedure of the experiment were briefly explained. Then, they were asked to fill out a pre-experiment questionnaire with the following information: personal data (age, gender); handedness (right-handed, left-handed); previous experience with touchscreen devices (tablets, smartphones, smartwatches, etc.) and with the three text entry methods; level of proficiency in English.

The experiment was conducted in a well-lit laboratory. Participants were asked to wear the watch on the non-dominant arm, adjusting the strap comfortably and possibly resting their arm on a table. They remained seated during the execution of the tasks. A picture of a participant performing the experiment is shown in Figure 2. Then, they had a short practice in which the use of the three tested text entry methods were explained. The participants were given all the recommendations related to the experiment, including:

1) reading and memorizing the phrase before starting to copy it;
2) balancing speed and accuracy while writing;
3) correcting possible errors noticed while entering text. Corrections were only possible by using backspace (left flick in MyScript).

The measured trials started after the operator ascertained that the participant had well understood the procedure. The task was to transcribe short sentences of text, as in almost all of the recent text entry experiments. Each participant had to insert 6 phrases in each of the 6 test condition. The first one was for training and not measured. Phrases were chosen at random from MacKenzie and Soukoreff set [35], which do not include punctuation or numbers. At the end of each test condition, participants were allowed to have a rest of a few minutes.

At the end of the task, each participant was asked to complete the post-questionnaire to collect opinions. To record subjective impressions on the input methods we used the System Usability Scale (SUS) questionnaire [36]. SUS includes

---

[1] http://myscript.com


the world_
the world is a stage

please_
please try to be home before midnig

important_
important news always seems to be

(a) A system based on MyScript.

(b) Our implementation of T9.

(c) SplitBoard.

Fig. 1. The three text entry methods tested in the experiment.

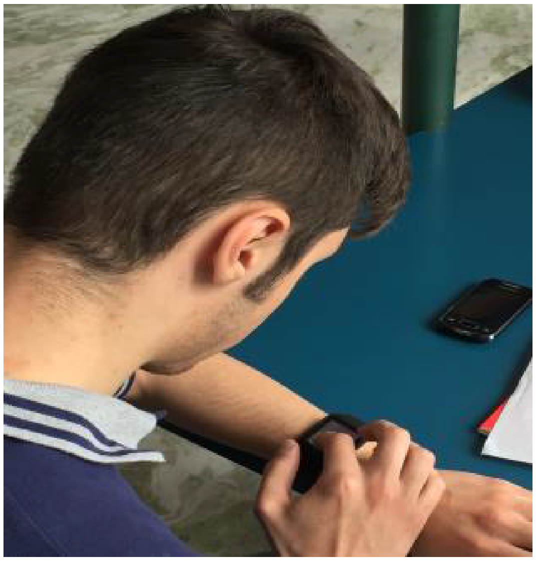

Fig. 2. A participant performing the experiment.

ten statements, to which respondents had to specify their level of agreement using a 5-point Likert scale. Questions alternate between positive and negative wording and are in a somewhat standard form. For this reason, we do not report the whole set of questions here. Each user questionnaire has a score in the range 0-100. Scores are then averaged across participants. Besides SUS, we gathered further opinions and preferences through structured questions and freeform comments. The questionnaire included five questions aimed at comparing the feelings of the users on the three methods: the participants had to declare their preference for one of the three methods with regard to speed, ease of use, accuracy, comfort, and overall impression.

4) Design: The experiment was a two-factor within-subjects design. The factors were the *Input method* and the *Display size*. The Input method included the following three levels:

- *MyScript* - the character by character handwriting system;
- *SplitBoard* - the soft keyboard with panning functions with the QWERTY layout described in [3];
- *T9* - the 12 keys soft keyboard implementing the T9 disambiguation method.

The display size included two levels:

- *Medium* - the full screen mode at 320×320 resolution;
- *Small* - the reduced screen mode at 280×280 resolution.

As dependent variables we included the following:

- Speed - the text entry speed measured in wpm, calculated as specified in [37]: the total number of entered characters (the first entered character is excluded) is divided by the time to enter them. In this metric, a *word* is conventionally composed of five characters.
- Accuracy - the text entry accuracy using both the total error rate (TER) and the not corrected error rate (NCER), calculated as follows

  $$TER = \frac{INF + IF}{C + INF + IF} \times 100\%$$

  $$NCER = \frac{INF}{C + INF + IF} \times 100\%$$

  As specified in [38], the keystrokes are subdivided in *Correct* (C), *Incorrect and Not Fixed* (INF), *Incorrect but Fixed* (IF) and *Fixes* (F).
- GPC - The number of gestures per character. Gestures include key presses, flicks, and any stroke used to enter (or delete) a character or a part of it.

We counterbalanced the order for the two factors, method (MyScript, SplitBoard and T9) and display size (medium, small). We obtained 6 different permutations for the former factor (method) and 2 for the latter (display size). Test conditions with the same method were always consecutive, not to confuse participants. This led to 12 different orders for 24 participants.

Aside from training, the amount of entry was 24 participants × 6 test conditions × 5 phrases = 720 phrases for the whole experiment.

B. Results

All participants completed the experiment. Except for induction and questionnaires, the trials lasted about 45 minutes per participant. This time also includes the rest time between test conditions (5 pauses of about 3 minutes each). We tested for significance using a repeated measures analysis of variance (ANOVA). For significant main effects, we used Scheffé post-hoc tests. The alpha level was set to 0.05.

1) Speed: The text entry speeds are reported in Figure 3. The grand mean for writing speed was 9.8 wpm. T9 was the fastest input method at 11.7 wpm. SplitBoard followed with 10.1 wpm, while the slowest was MyScript with 7.5 wpm. From the ANOVA resulted that the main effect of the input method on the speed was highly significant ($F_{2,46} =$

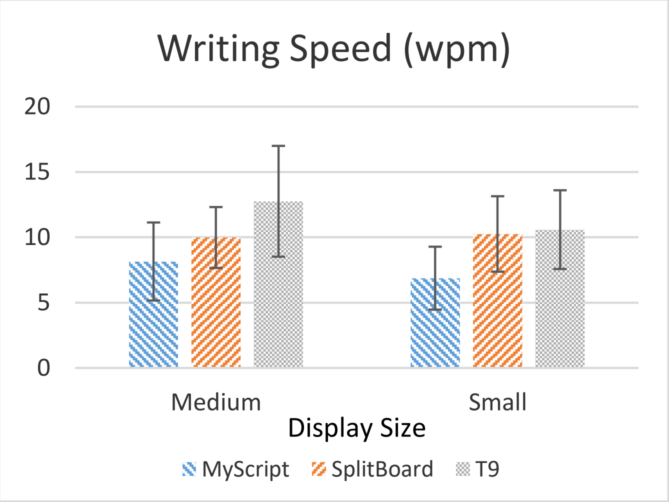

Fig. 3. Speeds (in wpm) measured in Experiment I. Error bars show the standard deviation.

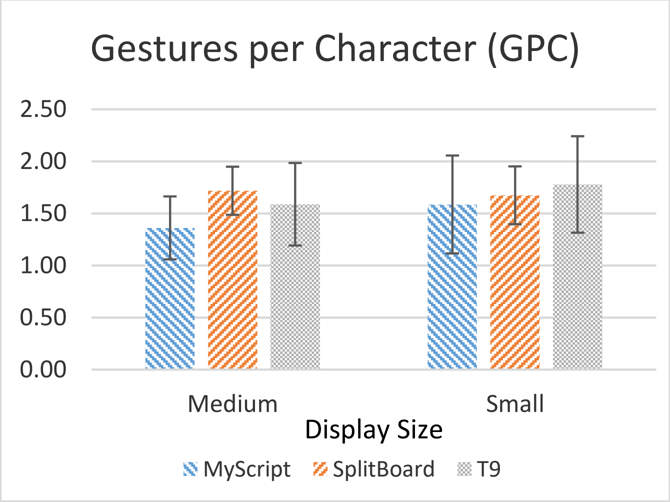

Fig. 4. Gestures per Character (GPC) measured in Experiment I. Error bars show the standard deviation.

30.305, $p < .0001$). There was also a significant effect for the display size ($F_{1,23} = 9.527$, $p = .0052$). The interaction effect between the two factors was also statistically significant ($F_{2,46} = 4.677$, $p = .0142$). A Sheffé post-hoc test revealed significant differences between T9-medium and MyScript-medium and between MyScript-small and all conditions with SplitBoard/T9.

2) Accuracy: Average values for TER and NCER are summarized in Table I (standard deviations are reported in parentheses). Values are reported by input methods and display sizes. The grand mean for TER was 14.02%. MyScript had the smallest TER with 9.90%. From the ANOVA we observed a significant effect of the input method ($F_{2,46} = 13.627$, $p < .0001$). There was also a significant effect of display size ($F_{1,23} = 4.803$, $p = .0388$) and a significant interaction between the two independent variables ($F_{2,46} = 6.050$, $p = .0047$).

The grand mean for NCER was 1.21%. Also in this case, MyScript proved to be the most accurate method with 0.64%, followed by SplitBoard with 1.39% and T9 with 1.59%. The main effect of the input method was not statistically significant ($F_{2,46} = 1,961$, $p = .1523$).The main effect of the display size was not statistically significant($F_{1,23} = 1.358$, $p = .2559$). The effect of the interaction between the two independent variables was marginally significant ($F_{2,46} = 3.245$, $p = .0480$).

3) Gestures per Character: The number of gestures per character (GPC) is reported in Figure 4. The grand mean for GPC was 1.618. The method with the lowest GPC was MyScript, with 1.474, followed by T9 with 1.683 and SplitBoard with 1.696. The main effect of the input method was significant ($F_{2,46} = 30.305$, $p < .0001$). There was a significant effect of the display size ($F_{1,23} = 9.527$, $p = .0052$). The effect of the interaction between the two independent variables ($F_{2,46} = 4.677$, $p = .0142$) was also significant. A Sheffé post-hoc test revealed significant differences between MyScript-medium and SplitBoard-medium, SplitBoard-small, T9-small.

4) User Satisfaction: The input method with the highest SUS mean score was SplitBoard, with 77.1 ($SD = 22.1$), followed by T9 with 70.5 ($SD = 17.2$), and MyScript with 65.9 ($SD = 18.2$). A Friedman test showed that the difference was not significant ($H = 3.65, p = .1612$).

The trend on such scores was confirmed in the choice of the preferred input method. When asked to choose, 12 participants preferred SplitBoard, 6 preferred T9 and 6 MyScript (see “Overall” in Figure 5). Participants’ preference seems to be due mainly to their perceived accuracy (20 out of 24 preferred SplitBoard). Their perception contrasts with our experimental results, which revealed a greater accuracy for MyScript. Freeform comments on this aspect revealed a difficulty for some participants in recognizing specific characters and inefficiency in correcting errors for MyScript. Conversely, MyScript was preferred for its ease of use (11 out of 24) and comfort (11 out of 24). T9 turns out to be the fastest method and is perceived as such by the majority of participants (13 out of 24). Furthermore, freeform comments about the three tested techniques were generally positive. Isolated criticisms, besides MyScript’s accuracy, was voiced against SplitBoard: some participants complained that flicks to switch between layouts are too frequent.

One participant found entering text difficult since he regarded the smartwatch as a “not sufficiently stable” device. Complaints regarding the size of the display were directed only towards the keyboard-based techniques. Five participants out of 24 complained about the size of both the small and medium keyboards, while additional 10 complained about the size of the small variant. Almost all of them complained about the general size of the keys. One participant would have preferred to have a largest backspace key.

IV. EXPERIMENT II: IDEAL HANDWRITING PERFORMANCE

We performed an experiment to evaluate the performance of handwriting and user satisfaction in ideal settings. We tested the writing performance with different writing styles

TABLE I
EXPERIMENT I: ERROR RATES OBTAINED BY THE THREE INPUT METHODS AND THE TWO DISPLAY SIZES.

	MyScript		**SplitBoard**		**T9**	
	medium	**small**	**medium**	**small**	**medium**	**small**
TER ***(SD)***	7.82% *(4.68%)*	11.98% *(8.09%)*	15.29% *(6.69%)*	12.72% *(7.97%)*	15.57% *(7.77%)*	20.76% *(9.73%)*
NCER ***(SD)***	0.69% *(1.22%)*	0.58% *(0.85%)*	1.51% *(2.68)*	1.27% *(1.60%)*	0.74% *(1.03%)*	2.44% *(4.40%)*

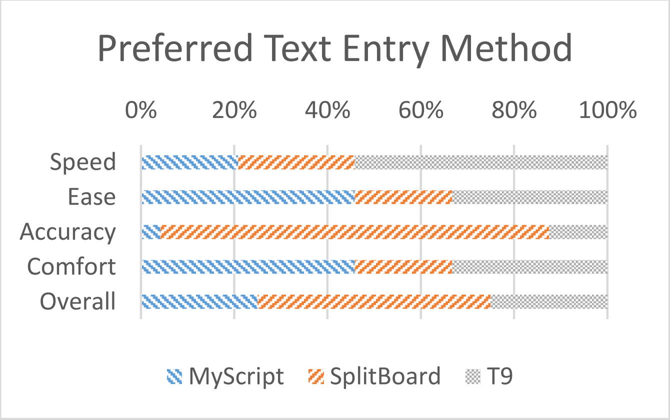

Fig. 5. Experiment I participants' preferences about the three text entry methods.

Fig. 6. Experiment II user interface.

(uppercase, lowercase, and unconstrained). Moreover, given that recent research [7] highlighted the potential for recognizing phrases without words separators with the use of powerful language models, we tested the possibility to omit separators between words while writing. As an alternative, we tested one of the classic ways to separate words, i.e. by performing a swipe gesture to the right.

A. Methods

1) Participants: We recruited 24 participants (8 females) between 21 and 30 years old (M=24.3, SD=2.3) with no overlapping with those of the previous experiment. They were all university students who agreed to participate for free. All of them had experience with touchscreens while only five of them had some experience (two of them were very experienced) with smartwatches. Only 6 participants had some experience with handwriting on touch devices, and only one of them was highly experienced.

2) Apparatus: The experiment was carried out on the same device of the previous experiment.

We developed and installed on the device an experimental software whose duty was to show the text to be transcribed to the user, record user input and calculate text entry metrics. The interface of this system is shown in Figure 6. It is rather simple and divides the screen into three sections:

- The upper line shows the text to transcribe.
- The lower left corner contains a button to load the next phrase. This location was regarded as the best to avoid accidental pressures while writing. Furthermore, we performed an additional check based on stroke length and endpoint positions to distinguish a click on the button from a gesture.
- The rest of the screen is used to draw characters.

To help participants not to get confused, at the beginning of each condition the system popups a message indicating the start of the new condition and its parameters. Furthermore, the text is presented differently in the various conditions: in those requiring the writing of capital letters, the text is displayed in upper case; in those requiring the use of the separator, the words are separated by a hyphen ('-') character.

Character input is performed seamlessly, without any timeout between two characters: the current character is superimposed on the previous one. The traces of gestures are slowly fading, using a grayscale until complete disappearance. Fading helps users in drawing multistroke characters by showing them the previous stroke. At the same time the user is not distracted by the trace of the previous character while drawing the current one.

3) Procedure: The procedure of the experiment was very similar to that described in section III-A3 for the previous experiment, i.e. including the induction, questionnaires, location, posture of the participants, task and pauses. Regarding the handwriting style, participants were clarified that a character could be entered with an unconstrained number of gestures but they were not allowed to enter more than one character through a single gesture.

The recommendations given to participants were the same, except for correction. Participants were instructed to correct the previously entered character through a *left flick* gesture when they felt they had entered a wrong character. Since our software did not perform character recognition, the search for input errors was made after the experiment on the logged data, with the goal of eliminating data affected by a significant number of errors (e.g., incomplete phrases or with missing words, words added by mistake, etc.). In particular, with the aid of a visualization software, we inspected all the gestures traces produced by the participants during the execution of the task. Phrases with small input errors (e.g., simple misspellings) were not removed.

Besides SUS, the post-questionnaire included two questions regarding the preferred method (against a miniaturized keyboard and other methods) and writing style and freeform comments.

4) Design: The experiment was a two-factor within-subjects design. The factors were the *writing style* and the *separator*. The writing style included the following three levels:

1) *Uppercase* - participants entered all characters uppercase;
2) *Lowercase* - participants entered all characters lowercase;
3) *Unconstrained* - participants were left free to choose the case for any input letter.

The separator included two levels:

1) *NoSpace* - participants entered all the characters of a sentence with no explicit separator between words;
2) *Swipe* - participants made a right swipe gesture to separate words;

The orders of the three writing styles and two separators were counterbalanced across participants with a schema similar to that of the previous experiment. In this experiment, test conditions with the same writing style were always consecutive.

As dependent variables we included two variables related to performance (speed in wpm and the number of gestures per character), and the SUS score. The speed and the number of gestures per character were calculated on the basis of the number of characters in the presented text, including spaces.

The amount of entry was 720 phrases (24 participants × 6 test conditions × 5 phrases), as in the previous experiment.

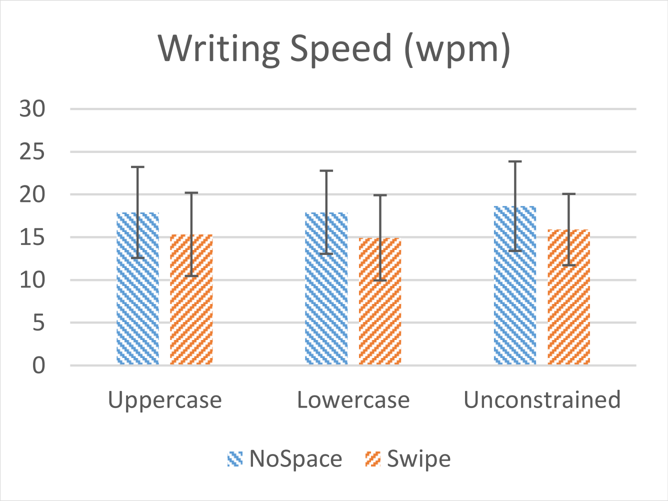

Fig. 7. Speed (in wpm) measured in Experiment II. Error bars show the standard deviation.

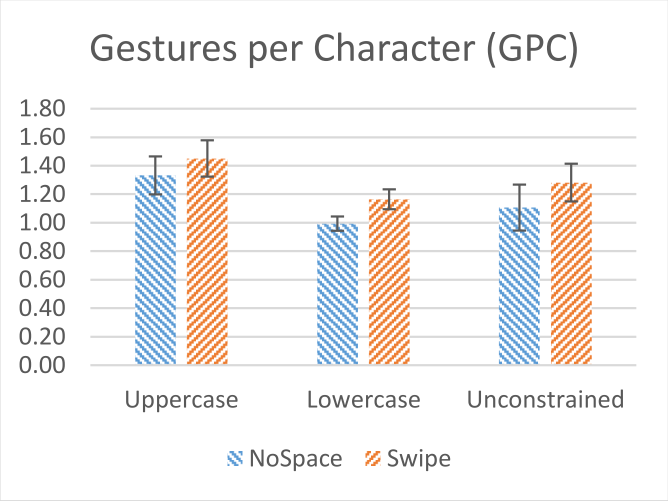

Fig. 8. Gestures per Character (GPC) measured in Experiment II. Error bars show the standard deviation.

B. Results

Except for induction and questionnaires, the trials lasted about 30 minutes per participant, including breaks. All participants completed the experiment. The visual inspection of the data resulted in the elimination of just one phrase out of 720.

We tested for significance using a repeated measures analysis of variance (ANOVA). For significant main effects, we used Sheffé post-hoc tests. The alpha level was set to 0.05.

1) Writing Speed: The writing speeds are reported in Figure 7. The grand mean for writing speed was 16.8 wpm. We sought a small difference among the different writing styles. Unconstrained writing was the fastest at 17.3 wpm. From the ANOVA resulted that the main effect of writing style on the speed was not statistically significant. The effect of the separator, instead, was highly significant ($F_{1,23} = 122.881$, $p < .0001$). Sparing the swipe to insert a space allowed an increase of writing speed of about 18%, from 15.4 wpm to 18.2 wpm. The interaction effect between the two factors was not significant.

2) Gestures per Character: The number of gestures per character (GPC) is reported in Figure 8. The grand mean for GPC was 1.22. The differences due to the writing styles were highly significant ($F_{2,46} = 68.545$, $p < .0001$). A Sheffé post-hoc test revealed significant differences among all pairs of writing styles except between Uppercase-Swipe and Unconstrained-NoSpace, Lowercase-NoSpace and Unconstrained-Swipe, Lowercase-NoSpace and Unconstrained-Swipe. We expected this result, as most users prefer to write uppercase characters in *printed* style. It is well known that this style requires more (pen) strokes than *cursive* style. The effect of the separator was also statistically significant ($F_{1,23} = 113.841$, $p < .001$). Sparing the swipe to insert a space allowed a decrease of GPC of about 16%. The interaction effect between the two factors was not significant.

3) User satisfaction and Preferences: From the SUS questionnaire resulted that, except for two participants with very low scores of 42.5 and 50, all of our participants appreciated the usability of handwriting on the smartwatch, with scores

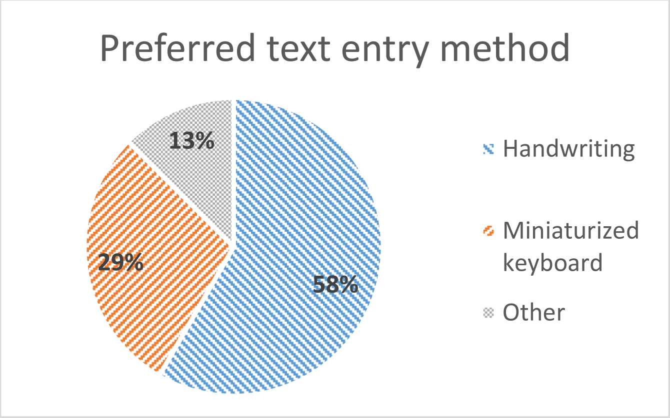

Fig. 9. Experiment II participants' preferred text entry method.

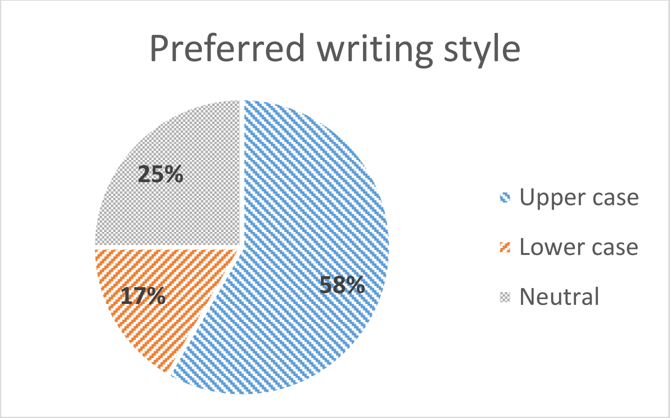

Fig. 10. Our participants' preferred writing style.

ranging from 62.5 to 97.5. The average score was 77.5, which can be regarded as a good result [39].

The technique was preferred over other possible text entry methods, with the results shown in Figure 9. Participants' preference for uppercase writing resulted from the answer given to a specific question. It was chosen by twelve of them, as shown in Figure 10.

Free form comments also indicated a general acceptance of the technique and a willingness of having it installed on their own devices. General criticisms regarding the technique concerned the writing on a small device, which for three participants "may be tiring in the long run", and the entry of a single character at a time, judged uncomfortable by one of the participants. Writing without spaces was particularly appreciated by two of the participants, who found it "useful because it saves time". However, for one participant it was "more difficult".

We received some complaints regarding the implementation of the experimental software. In particular, the fading of the finger gesture did not always achieve the desired objective, and four participants were sometimes confused by this. A good system design should pay special attention to the implementation of this feature.

V. DISCUSSION

In the first experiment we observed a fairly limited speed for the current method of handwriting. This proved to be significantly lower than the other two tested methods in most test conditions. The differences observed on accuracy, although not statistically significant, showed a better accuracy for handwriting w.r.t. the other two methods. In particular, it was the only method with less than 1% error rate, traditionally regarded as a limit value for acceptability of NCER [40]. Nevertheless, difficulties in recognizing specific characters and in performing corrections negatively impacted on its perceived accuracy. Despite the low speed values, we recorded a fairly good acceptance by the users for handwriting, due to its appreciated ease and comfort of use. Handwriting remains one of the most natural ways of entering text and despite the limitations of the currently available implementation, some users prefer it over the use of keyboards.

As for the second experiment, our judgement on the experience is generally positive, since we recorded some rather good initial writing speeds (15-19 wpm), much faster than all the speeds reached in the first experiment. The SUS questionnaire also showed good levels of satisfaction. Furthermore, our participants expressed a preference for handwriting rather than another method based on a soft keyboards. It is worth noting that this result is based on participants' perception and not on a real experience. It is interesting to note that the speeds recorded in our second experiment are similar to those achieved in previous experiments with unconstrained handwriting [13]. Those experiments were obtained with very different settings. Comparing results, we can perceive by intuition that the use of a small wearable device and the character by character input do not significantly affect writing speed.

In our study, we assumed that a good recognizer, based on dictionaries and language models could correctly interpret and forgive misspellings and malformed characters not explicitly corrected by the user. Our results must be interpreted as an upper bound to the initial user performance for handwriting on smartwatches. A useful result for the scientific community is that handwriting on very small screens can hardly get better performance than those reported here and any alternative system exceeding those performance and with a good user acceptance will be preferable.

An interesting information we can give to the designers of handwriting methods for small screens is the clear preference, resulted from the second experiment, of the users to write in capital letters. Despite the uppercase characters require a greater number of gestures, their writing is not slower. Lastly, avoiding the entry of space characters (possibly feasible with the use of dictionaries and language models) improves the writing speed of about 18%. A well-designed recognizer could use this feature to get a significant gain in terms of speed.

Taken together, the results of the two experiments showed that there is a large gap between the performance of the currently available implementations and those obtainable by an ideal handwriting system. This interpretation was also confirmed by participants' comments. Margins for improvement are in recognition, error correction and feedback methods and their combination. In our tests, MyScript performed a recognition both at a character level and at a word level (the latter based on the dictionary). Giving feedback after each entered character entails a considerable loss of time for the user to check the correctness of input. This scheme proved to be too slow. It is necessary to study more efficient feedback and correction models. We are convinced that much (certainly not all) of the gap between the current implementation and the ideal system can be filled with the best combination of them.

The experiments described in this study suffer from some limitations. As most of the experiments described in the literature, they are based on the input of short phrases and were mostly carried out with young participants. It is worth pointing out that prolonged writing on very small screens can cause fatigue and pain. This aspect was not investigated enough and deserves further investigation.

VI. CONCLUSION AND FUTURE WORK

In this paper, we described the results of an experimental study aimed at exploring the feasibility of handwriting on smartwatches and at obtaining some useful information regarding users' performance and preference.

We recorded initial writing speeds with an existing implementation of about 7.5 wpm. We also recorded the possibility of significantly improving this performance through the use of an optimal system. As mentioned in the previous section, handwriting speed on smartwatches may be improved by using more efficient feedback techniques. Such techniques must also include error correction. One option would be not to give any feedback after each character, but only a sort of segmentation as done in [41] and use only a word-level feedback. Future work will be aimed at experimenting this and other correction techniques combined together.

In this study, we tested handwriting on a watch with a square shaped display. Lately, watches with circular display are spreading. Those devices enable, due to the increased display width, the use methods which show a full QWERTY keyboard. Testing handwriting in comparison to a one of such methods (e.g., Watchwriter [33]) on large circular displays is also planned for future studies.

Lastly, it would be also interesting to test the use of wearable devices by elderly users, which constitute a growing proportion of the population in some countries and are more and more confronted with small electronic devices. This topic is particularly important in the context of text entry, as it is well known that handwriting is subject to deterioration with age and diseases [42]. It would also be interesting to investigate the possibility of operating the smartwatch through a stylus, whose use in some contexts is preferred or preferable over that of finger [43], [44].

ACKNOWLEDGMENT

The authors would like to thank Jonggi Hong for providing an implementation of Splitboard. They also thank Luigi Amitrano, Michele Angeloro, Domenico Corbisiero, Rita D'Ambrosio, Antonella Guerra, Raffaela Piscitelli and Angelo Ranieri for their support in carrying out the experiments.

REFERENCES

[1] S. Oney, C. Harrison, A. Ogan, and J. Wiese, "Zoomboard: a diminutive qwerty soft keyboard using iterative zooming for ultra-small devices," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2013, pp. 2799–2802.

[2] A. Komninos and M. Dunlop, "Text input on a smart watch," *Pervasive Computing, IEEE*, vol. 13, no. 4, pp. 50–58, 2014.

[3] J. Hong, S. Heo, P. Isokoski, and G. Lee, "Splitboard: A simple split soft keyboard for wristwatch-sized touch screens," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 2015, pp. 1233–1236.

[4] R. R. van Doorn and P. J. Keuss, "The role of vision in the temporal and spatial control of handwriting," *Acta Psychologica*, vol. 81, no. 3, pp. 269–286, 1992.

[5] P. O. Kristensson and L. C. Denby, "Text entry performance of state of the art unconstrained handwriting recognition: a longitudinal user study," in *Proc. of CHI '09*. New York, NY, USA: ACM, 2009, pp. 567–570.

[6] D. Goldberg and C. Richardson, "Touch-typing with a stylus," in *Proceedings of the INTERACT'93 and CHI'93 conference on Human factors in computing systems*. ACM, 1993, pp. 80–87.

[7] K. Vertanen, H. Memmi, J. Emge, S. Reyal, and P. O. Kristensson, "Velocitap: Investigating fast mobile text entry using sentence-based decoding of touchscreen keyboard input," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 2015, pp. 659–668.

[8] M. D. Dunlop and M. M. Masters, "Pickup usability dominates: a brief history of mobile text entry research and adoption," *International Journal of Mobile Human Computer Interaction (IJMHCI)*, vol. 1, no. 1, pp. 42–59, 2009.

[9] J. Hong, S. Heo, P. Isokoski, and G. Lee, "Comparison of three qwerty keyboards for a smartwatch," *Interacting with Computers*, p. iww003, 2016.

[10] R. Plamondon and S. N. Srihari, "On-line and off-line handwriting recognition: A comprehensive survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, pp. 63–84, January 2000.

[11] I. S. MacKenzie and R. W. Soukoreff, "Text entry for mobile computing: Models and methods, theory and practice," *Human–Computer Interaction*, vol. 17, no. 2-3, pp. 147–198, 2002.

[12] A. S. Arif and A. Mazalek, "A survey of text entry techniques for smartwatches," in *International Conference on Human-Computer Interaction*. Springer, 2016, pp. 255–267.

[13] D. B. Devoe, "Alternatives to handprinting in the manual entry of data," *IEEE Transactions on Human Factors in Electronics*, no. 1, pp. 21–32, 1967.

[14] C. H. Blickenstorfer, "Graffiti: Wow!!!!" *Pen Computing Magazine*, vol. 1, pp. 30–31, 1995.

[15] S. J. Castellucci and I. S. MacKenzie, "Graffiti vs. unistrokes: an empirical comparison," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2008, pp. 305–308.

[16] J. O. Wobbrock, B. A. Myers, and J. A. Kembel, "Edgewrite: A stylus-based text entry method designed for high accuracy and stability of motion," in *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology*, ser. UIST '03. New York, NY, USA: ACM, 2003, pp. 61–70. [Online]. Available: http://doi.acm.org/10.1145/964696.964703

[17] I. S. MacKenzie and L. Chang, "A performance comparison of two handwriting recognizers," *Interacting with Computers*, vol. 11, no. 3, pp. 283–297, 1999.

[18] A. Pavlovych and W. Stuerzlinger, "Less-tap: A fast and easy-to-learn text input technique for phones." in *Graphics Interface*, vol. 2003. Citeseer, 2003, pp. 97–104.

[19] S. Hwang and G. Lee, "Qwerty-like 3x4 keypad layouts for mobile phone," in *CHI '05 Extended Abstracts on Human Factors in Computing Systems*, ser. CHI EA '05. New York, NY, USA: ACM, 2005, pp. 1479–1482. [Online]. Available: http://doi.acm.org/10.1145/1056808.1056946

[20] I. S. MacKenzie, *KSPC (Keystrokes per Character) as a Characteristic of Text Entry Techniques*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 195–210. [Online]. Available: http://dx.doi.org/10.1007/3-540-45756-9_16

[21] C. L. James and K. M. Reischel, "Text input for mobile devices: Comparing model prediction to actual performance," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '01. New York, NY, USA: ACM, 2001, pp. 365–371. [Online]. Available: http://doi.acm.org/10.1145/365024.365300

[22] I. S. MacKenzie, H. Kober, D. Smith, T. Jones, and E. Skepner, "Letterwise: Prefix-based disambiguation for mobile text input," in *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology*, ser. UIST '01. New York, NY, USA: ACM, 2001, pp. 111–120. [Online]. Available: http://doi.acm.org/10.1145/502348.502365

[23] I. S. MacKenzie and S. X. Zhang, "The design and evaluation of a high-performance soft keyboard," in *Proc. of CHI '99*, 1999, pp. 25–31.

[24] S. Zhai, M. Hunter, and B. A. Smith, "The metropolis keyboard - an exploration of quantitative techniques for virtual keyboard design," in *Proc. of UIST '00*. New York, NY, USA: ACM, 2000, pp. 119–128.

[25] X. Bi, B. A. Smith, and S. Zhai, "Quasi-qwerty soft keyboard optimization," in *Proceedings of CHI '10*. ACM, 2010, pp. 283–286.

[26] ——, "Multilingual touchscreen keyboard design and optimization," *Human-Computer Interaction*, vol. 27, no. 4, pp. 352–382, 2012.

[27] J. Mankoff and G. D. Abowd, "Cirrin: a word-level unistroke keyboard for pen input," in *Proceedings of the 11th annual ACM symposium on User interface software and technology*. ACM, 1998, pp. 213–214.

[28] K. Perlin, "Quikwriting: continuous stylus-based text entry," in *Proceedings of the 11th annual ACM symposium on User interface software and technology*. ACM, 1998, pp. 215–216.

[29] P.-O. Kristensson and S. Zhai, "Shark 2: a large vocabulary shorthand writing system for pen-based computers," in *Proceedings of the 17th annual ACM symposium on User interface software and technology*. ACM, 2004, pp. 43–52.

[30] S. Zhai and P. O. Kristensson, "The word-gesture keyboard: reimagining keyboard interaction," *Communications of the ACM*, vol. 55, no. 9, pp. 91–101, 2012.

[31] V. Fuccella, M. De Rosa, and G. Costagliola, "Novice and expert performance of keyscretch: A gesture-based text entry method for touch-screens," *IEEE Transactions on Human-Machine Systems*, vol. 44, no. 4, pp. 511–523, 2014.

[32] L. A. Leiva, A. Sahami, A. Catalá, N. Henze, and A. Schmidt, "Text entry on tiny qwerty soft keyboards," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 2015, pp. 669–678.

[33] M. Gordon, T. Ouyang, and S. Zhai, "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*. ACM, 2016, pp. 3817–3821.

[34] D. Wigdor and R. Balakrishnan, "Tilttext: using tilt for text input to mobile phones," in *Proceedings of the 16th annual ACM symposium on User interface software and technology*. ACM, 2003, pp. 81–90.

[35] I. S. MacKenzie and R. W. Soukoreff, "Phrase sets for evaluating text entry techniques," in *CHI'03 extended abstracts on Human factors in computing systems*. ACM, 2003, pp. 754–755.

[36] J. Brooke *et al.*, "Sus-a quick and dirty usability scale," *Usability evaluation in industry*, vol. 189, no. 194, pp. 4–7, 1996.

[37] I. S. MacKenzie, "A note on calculating text entry speed," last visit 2014, http://www.yorku.ca/mack/RN-TextEntrySpeed.html.

[38] R. W. Soukoreff and I. S. MacKenzie, "Metrics for text entry research: An evaluation of msd and kspc, and a new unified error metric," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '03. New York, NY, USA: ACM, 2003, pp. 113–120. [Online]. Available: http://doi.acm.org/10.1145/642611.642632

[39] A. Bangor, P. Kortum, and J. Miller, "Determining what individual sus scores mean: Adding an adjective rating scale," *Journal of usability studies*, vol. 4, no. 3, pp. 114–123, 2009.

[40] M. LaLomia, "User acceptance of handwritten recognition accuracy," in *Conference companion on Human factors in computing systems*. ACM, 1994, pp. 107–108.

[41] W. Kienzle and K. Hinckley, "Writing handwritten messages on a small touchscreen," in *Proceedings of the 15th international conference on Human-computer interaction with mobile devices and services*. ACM, 2013, pp. 179–182.

[42] A. Holzinger, G. Searle, and M. Wernbacher, "The effect of previous exposure to technology on acceptance and its importance in usability and accessibility engineering," *Universal Access in the Information Society*, vol. 10, no. 3, pp. 245–260, 2011.

[43] A. Holzinger, M. Holler, M. Schedlbauer, and B. Urlesberger, "An investigation of finger versus stylus input in medical scenarios," in *Information Technology Interfaces, 2008. ITI 2008. 30th International Conference on*. IEEE, 2008, pp. 433–438.

[44] A. Holzinger, G. Searle, B. Peischl, and M. Debevc, "An answer to "who needs a stylus?" on handwriting recognition on mobile devices," in *International Conference on E-Business and Telecommunications*. Springer, 2011, pp. 156–167.