# Braille Text Entry on Smartwatches

## An Evaluation of Methods for Composing the Braille Cell

Mateus M. Luna
Universidade Federal de Goiás, Goiânia, Brazil
mateus.m.luna@gmail.com

Fabrízzio A. A. de M. N. Soares
Universidade Federal de Goiás, Goiânia, Brazil
fabrizzio@inf.ufg.br

Hugo A. D. Nascimento
Universidade Federal de Goiás, Goiânia, Brazil
hadn@inf.ufg.br

Aaron Quigley
University of St Andrews, St Andrews, Scotland
aquigley@acm.org

## ABSTRACT

Smartwatches are gaining popularity on market with a set of features comparable to smartphones in a wearable device. This novice technology brings new interaction paradigms and challenges for blind users, who have difficulties dealing with touchscreens. Among a variety of tasks that must be studied, text entry is analyzed, considering that current existing solutions may be unsatisfactory (as voice input) or even unfeasible (as working with tiny QWERTY keyboards) for a blind user. More specifically, this paper presents a study on possible solutions for composing a Braille cell on smartwatches. Five prototypes were developed and different feedback features were proposed. These are confronted with seven specialists on an evaluation study that results in a qualitative analysis of which strategies can be more useful for blind users in a Braille text entry.

## CCS CONCEPTS

• **Human-centered computing** → **Accessibility design and evaluation methods**;

## KEYWORDS

blind people, braille, text entry, text input, smartwatches

## 1 INTRODUCTION

Wearable devices, specially smartwatches and smartbands, are starting to take consumers daily routine in the last years. These devices can be used as an extension of a smartphone or even as a standalone platform for many applications. They come with benefits of being constantly carried by their users, attached to their body, enabling easy and immediate interaction [33].

Their small screen has been the cause for proposing new layouts, in order to provide a better experience on tasks such as text entry. Currently many smartwatches rely only on voice recognition as

input method or even on small QWERTY keyboards, with strong word-model and auto completion.

If such a device is challenging for overall users, it can be even more complicated for blind and visually impaired people. Voice input is an alternative, but it comes with its issues of bad recognition and lack of privacy [23, 29]. The current solution for QWERTY keyboards can even be adapted to the use of screen readers, such as TalkBack and VoiceOver, but on a screen size so small it would be harder for blind people to use them on smartwatches.

A new solution would be to work on a method that provides Braille text entry for smartwatches. Braille is a system based on cells of up to six or eight dots per character, distributed over two columns. It is used by low vision and blind people, specially for reading information on paper, and has been adopted around the world not only for literacy, but for daily information such as public signs, medicine in drug stores, etc. Even with the arrival of technologies that improve accessibility to use them on smartphones, Braille education stills holds its importance for blind people autonomy [8]. As it will be discussed in the related work, some studies took steps in this direction, but none of them clearly solves the issues.

From the understanding that different solutions must be provided for different kinds of users [20], we here describe the development of a method for text entry on smartwatches focused on blind people who are literate in Braille. First, we propose five different prototypes for braille cell composition, using different approaches inspired on literacy. After, we aim to narrow down these methods through an evaluation study and gather opinions on our ideas and features. Up to our knowledge, this is the first project aiming specifically at this kind of device for Braille text entry.

The remainder of the paper is organized as follows: next section describes our review on literacy, followed by our method presentation. Prototypes and evaluation protocol are highlighted. Then comes the evaluation section with results, just before our discussion session. Finally, future work and conclusion encloses the paper.

## 2 RELATED WORK

### 2.1 Text entry on smartwatches

Text entry on smartwatches had been addressed by many authors as an open problem, specially due to small screen sizes. It is covered on a systematic review of the literature [15].

A strategy that had been often adopted is to propose new layouts to fit bigger buttons on screen, separating character selection into two stages, where the first one reduces the amount of elements on screen, to guarantee a precise click on the second one. This was introduced in ZoomBoard [22], an interactive zooming keyboard, and similar concepts appear on SplitBoard [11], SwipeBoard [5] VirtualSlidingQWERTY [4], DriftBoard [25] and UniWatch [24].

Other researches analyze different ways of interpreting a click action, for providing more than one value to a single button on screen. ForceBoard [12] and DualKey [10] are examples. On screen gestural input is also theme for studies such as Invisiboard [18] and the work by Nascimento *et al* [19]. COMPASS [34] is a particular proposal of circular keyboard, using a rotational bezel.

Finally, more recent studies try to solve the problem by providing statistical word models, without causing modifications on QWERTY layout. Examples of work with this strategy are WatchWriter [9], Velocitap [32] and the method analyzed in Turner *et al* [30]. These works seem to inspire current industry adopted methods, such as Google's WearOS default QWERTY keyboard, which relies heavily on tracing gestures for word composition and predictions.

## 2.2 Braille text entry on touchscreen devices

Blind people interaction with touchscreen devices had been addressed by a variety of studies, focusing on many types of tasks, such as the work by Kane [14] on analyzing gestures onscreen. Text entry, is one that received particular attention, especially by those who stand that Braille make more sense for those literate on this system. The most important works from 2011 to 2015 about Braille text entry for smartphones are presented through a systematic review [26]. In addition we highlight a couple more recent articles.

BrailleType [21] presents three points on each side of screen: for each touch, a corresponding number of the Braille alphabet is read; in the center, the letter is confirmed and displayed. The BrailleTouch [28] accepted simultaneously six fingers on the screen. EdgeBraille [17] presents three points on each side, activated or disabled by finger gestures. The Perkinput [3] presents two approaches, with one or two smartphones, where text is typed using one hand in each device. On TypeInBraille [16], dots are composed two-by-two, sequentially. One finger selects the left or right dot, two fingers select both and three fingers don't select any dots. BrailleÉcran [27] has a tactile 3D printed film, which overlays the smartphone for braille text entry. SingleTapBraille [2] eliminates the need for finding specific locations on screen, allowing information to be entered, using a single finger. Some of these solutions inspire existing Braille keyboards for Android (SwiftBraille [1], SoftBrailleKeyboard[6]) and iOS (accessibility toolkit onscreen braille keyboard).

## 2.3 Smartwatches and Text Entry for the Blind

Smartwatches have been used in several studies regarding assistance to blind people, but most of these target in and outdoor displacement, sometimes using extra sensors. Text entry is less frequently debated, and Braille text entry is particularly more rare.

TacktBack [7] is as a vibration system to output braille cells for deaf-blind users. A sequence of up to three vibrations is emitted by a smartphone, paired with a smartwatch responsible for the other three vibrations. The method although, does not perform text input. In BrailleEasy [35], a single-handed braille keyboard for smartphones is introduced, and authors affirm that smartwatch input would work, but no study for such is made, which is needed for small screens considering that it requires three-fingers tap.

It is worth mention the work of Dot Inc. [13] on creation of Dot, a smartwatch effectively planned for blind users. Instead of a touchscreen, it provides a watch face with four eletromechanically controlled Braille cells. The watch can be paired with smartphones to receive notifications, translating to the braille cells. The product is interesting, but it still does not provide a method for text input.
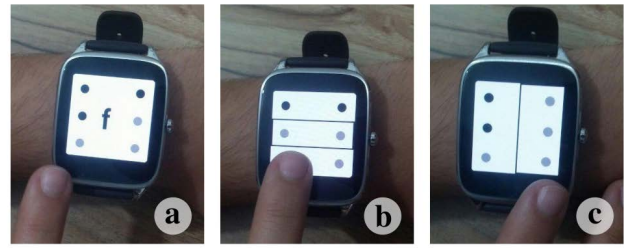


**Figure 1: Prototypes. a) Confirmation screen for Touch, Swipe and Connect methods. b) Line two composition for Serial method. c) Column two composition for Perkins.**

## 3 METHOD

In order to develop a text-entry method that can be useful for blind people in such a particular and novice environment as smartwatches, we access both our previous experience on Braille input for touchscreen devices and other researches publications. Based on related work, a set of different prototypes was implemented, strategies that must be evaluated by specialists in our study.

Instead of targeting all possible tasks related to text-entry, such as character navigation, erasing, number insertion and even using word-models for improving speed and accuracy of typing, we consider necessary to evaluate the most essential task in our context, which is to compose the Braille cell, thus choosing among the six dots that constitute a letter for character input. We also look into specific features that can improve how user interacts with device during this task, described in the extra features section.

The following are the variables in our user study, whose protocol is described in a separate subsection.

## 3.1 Proposed Prototypes

Every prototype provides a different method for composing the Braille cell. Upon confirmation of the dots selection, smartwatch speaker announces the chosen character and dots are reset to original state. Confirming a cell where all dots are deactivated inserts a white space. The generic layout can be seen in Figure 1.a.

**Touch**: The most basic interaction method proposed is based on BrailleType [21] and BrailleÉcran [27]. This works with the hypothesis that a watch small screen size could allow easy memorization of buttons position, especially after practicing. One tap on dot or its surrounding area toggles its activation. Confirmation of the desired Braille cell is performed by a double tap on the middle of the screen, mimicking the rationale of TalkBack and VoiceOver confirmation.

**Swipe**: This method attempts to minimize the need for target-precise clicks, using only directional gestures. The idea comes from previous work reporting that, for touchscreen devices, blind users would prefer gestures than buttons [14]. It is expected that by identifying corners user will have sense of necessary paths to swipe. User swipes in one of six directions to activate or deactivate a dot:

(1) Bottom-right to top-left corner;
(2) Middle-right to middle-left corner;
(3) Top-right to bottom-left corner;
(4) Bottom-left to top-right corner;
(5) Middle-left to middle-right corner and
(6) Top-left to bottom-right corner.

Confirmation of Braille cell is a double tap anywhere on screen.

**Connect**: This method is similar to Touch, but allows connecting dots to compose the braille cell (similar to patterns for unlocking smartphones). This interaction design is inspired by IPPITSU [31] and applications like SwiftBraille [1], already used in smartphones. User swipes on screen, passing through the desired dots area sequentially. Confirmation on Braille cell is performed by a short timeout (1.2 seconds), after finger releases screen. This allows user to perform a click to activate dots leaved off path, as long as it happens before the timeout finishes. If such a click occurs, the timeout is reset, meaning that one can compose the cell using only taps instead of tracing the path, if performed on time. Double tap here is not necessary for confirmation, except for entering a white space.

**Serial**: Based on TypeInBraille [16], with some modifications as the cited work uses three fingers input at same time in some situations, which is not feasible on Google's WearOS and Apple's WatchOS (this gesture is system reserved). It uses line-by-line composition of the Braille cell, thus user must deal with two dots per interaction. A line can be filled in three different ways:

(1) Pressing one finger at the right or left portion of screen activates its respective dot;
(2) Pressing two fingers anywhere on screen activates two dots;
(3) Swiping up gesture skips current line, leaving dots off.

After any of these actions, a cursor moves to next line automatically, expecting the composition of the next two dots. Once the three lines are completed, character is immediately confirmed and the next letter composition begins. Figure 1.b shows this prototype.

**Perkins**: Last prototype works in a similar way to Serial, but with column-by-column insertion, accessing possible user Perkins machine abilities. User can fill a column in four different ways:

(1) Pressing one finger at the top, middle or bottom portion of screen activates its respective dot;
(2) Pressing two fingers at the top portion of screen activates two top-most dots. Two fingers at the bottom portion of screen activates two bottom-most dots. Two fingers separated, on top and bottom, activates top-most and bottom-most dot;
(3) Swiping down gesture activates all three dots;
(4) Swiping up gesture skips column, leaving dots deactivated.

After any of these actions, cursor moves to next column automatically, expecting the composition of next three dots. If both columns are completed, character is immediately confirmed and the next letter can begin. Figure 1.c shows this prototype.

## 3.2 Extra Interaction Features

To complement most of the proposed interactions, some features can be used, based either on previous work or empirical evaluation. Above follows this features that could be chosen via a settings page.

**Vibration patterns per dot**: when activating or deactivating a dot, system already uses a soft vibration on device to work as feedback of performed action. We here extend this giving semantics the vibration, making each dot activation to cause the smartwatch to vibrate in a pattern related to the number of the activated dot. Those are 50ms duration vibrations, with 30ms interval.

**Dot dual tone feedback**: each dot can emit an audio feedback when activated, to help user have a better notion of what is being composed before character conclusion. We assume that speaking out loud dot number would be time consuming and confusing when dealing with numeric text-entry, thus opted for using a tone generator to emit sound for 100ms. Dual-Tone Multi-Frequency was adopted as an industry standard, already present during number

typing in many keypads such as cellphones and door locking terminals. This can be easily implemented in many devices;

**Screen rotation**: In early stages of development it was noticed user could prefer to work with a rotated screen. This would be useful in contexts such as sitting in front a table, using it as a support. It was implemented as an anticlockwise 90° rotation.

## 3.3 Study Protocol

To understand how the proposed methods and features affects users experience and which of them should potentially guide our following studies, an interview protocol was elaborated.

Initially, basic demographic and experience related information is gathered. Follows exposition on project idea, then a minute to interact with the smartwatch using its screen reader. The user is then presented to the six dots on screen, an introductory screen made for speaking out loud each dot that is clicked, allowing participant to get familiar with screen shape and dots disposal.

To evaluate the vibration patterns, dot speaking feedback is disabled and the six buttons are pressed by the researcher in a randomized order. User is then asked which number was pressed according to the vibration and if this feedback can be considered useful for awareness of the clicked button.

For evaluating dual tone audio feedback and screen rotation, user is asked to compose two braille code letters: *"s"* and *"z"*, with and without these extra features, one at a time. This letters where chosen due to its distribution of points, as presented on Figure 2. After each attempt, user answers, in a 5-level Likert scale from Completely Useless (1) to Very Useful (5) regarding the contribution of the feature for the Braille cell composition.

Follows the evaluation of the five prototypes. Due to its familiarity with the logic presented during the extra features evaluation, Touch prototype is always the first to be tested. The following prototypes order is randomized. For each of them, user is introduced to the interaction details and is free to compose up to three letters to validate if the explanation was clear. Then he or she is challenged to compose the following sentence: *"Olá, tudo bem?"*. The phrase Braille code, shown on Figure 2 is the Brazilian Portuguese equivalent to *"Hello, how are you?"*. It is used as a familiar expression that also requires capital letter, punctuation and spaces. As this study is not concerned with error rates or entry speed, we allow users to correct any mistakes made and even let them give up in case the process gets too frustrating. Our main concern is to expose different ways for composing the Braille cell. Completed this sentence, user must rate the method on a 5-level Likert score from (1) Too bad to (5) Very Good. He or she is then asked if it was clear which dot was being activated, if any difficulties where faced and if any positive highlights on the method could be presented.

To conclude study, after all prototypes are evaluated, volunteers are asked to rank them by preference, and then provide any suggestions regarding improvements on current methods. It is also questioned if it would be of their interest to acquire a smartwatch if it was possible to use similar writing technique.
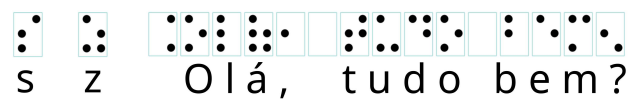


**Figure 2: Braille symbols for training letters and full sentence.**

## 3.4   Participants

Seven specialist participants volunteered for the experiments, four men and three woman, ranging in age from 38 to 61 years old (mean age 45). Four of the participants were born blind, learning Braille during their childhood. One participant lost sight during teenagers. The two remaining gradually lost sight during their twenties, and also learned Braille even though their use is less intensive than previously mentioned. Two are Braille teachers, four are public agents and one is a bank employee. Every participant uses smartphones during their daily routine with screenreaders. Two participants occasionally make use of onscreen Braille keyboard present on iPhone. Only one had access to smartwatches before, but without exploring all of its features, as it lacks of accessibility. Tests were applied individually and lasted around two hours each.

## 3.5   Materials

Our prototypes were tested on a ASUS ZenWatch2 005R. This smartwatch has a built-in speaker and is running WearOS 2.0. Overall watch face is 49.6 x 40.7mm, but the screen itself, or the clickable area is a square touchscreen asymmetrically positioned over the face, with 320x320 pixels ($\sim$ 42.5% screen-to-body ratio).

## 4   EVALUATION

Following is our objective results and information from interviews.

### 4.1   Prototypes evaluation

After interacting with each of the prototypes, participants gave a score, which is summarized on Figure 3(a), a chart for mean and standard deviation. To avoid any assumption about normality of data, a Kruskal-Wallis test (Chi-Squared) was conducted with 5% of significance and shows significant statistical difference among the five groups ($Chi-Sq = 15.78, p = 0.0033$). A multiple comparing test was performed with Tukey-Kramer test and its result is presented in Figure 3(b). We can observe that Perkins has no intersection with Connect and Swipe, but it has some intersection to methods Touch and Serial. The others methods has some intersection among them, however, it is very small between Swipe and Serial.

At the end of the experiment, prototypes are ranked, from most preferred to less approved. Figure 3(c) shows this preference order. Both analysis result on Connect method as the most approved.

Next we discuss in detail how the evaluation went with each prototype, considering the answers achieved from the questions.

**Touch**: Touch proved to be simple and basic to understand, being reported as easy to learn. Its success depends mostly of user habituating with dots disposition on screen. Confirmation of composed Braille cell using double tap action, although, did challenged some users, it should be performed on the middle area, outside of any button and with a short timeout between taps.

**Swipe**: The success of this prototype had strong dependence on the participant ability to perform the swipe gestures. Some users tend to begin their gestures outside of the clickable area of screen, which would lead to detection errors. This resulted on very mixed feelings on method (Figure 3(c)), being approved by a group, and reproved particularly by one participant who gave up its experiment with it. Those who approved found it interesting, but admit a need for practice to remember the directions. The interest on the prototype seems to overcome it's slow speed, causing some users to even prefer it over Touch.

**Connect**: Had the approval of the majority of participants, even though they performed in different ways. Some composed the braille cell tracing from one dot to the other. Other participants preferred to compose cell with the possibility of typing in a similar fashion of Touch method, but relying on the timeout for confirmation. A low error occurrence improved satisfaction with the prototype. It was noticed, though, that once activated one dot, it could not be deactivated by tracing back to its region, a feature that was requested upfront. Most participants affirm that with more time for practice, they would master this method, and type quickly.

**Serial**: Serial was complex to explain for most users. We point out two reasons. Firstly, as noticed by a couple of participants, there is no clear feedback to indicate which of the three lines is being composed, even though there is a sound indicating the line change. The cursor moves automatically to next line, but accidental click or swipe could have happened. Secondly, the idea of composing line by line may not be really intuitive to every blind user, as it goes against the numerical order of the cell, which works column by column. One user did enjoyed this method, being able to quickly use it, but we argue that she uses Braille less than the others.

**Perkins**: Already addressing one of Serial problems, Perkins strategy of inserting a cell column by column was well received by participants. However, more diverse interaction strategies are needed, such as two fingers click combinations, swipe up and down, which brings more complexity. It also presented issues, as the lack of clear feedback for which column is being composed and bad detection of two-finger clicks. Users need to have a good notion of the clickable space on screen to confidently press both fingers in acceptable positions. This lead to the worst performance of all methods, resulting in frustrations and one withdrawal.

### 4.2   Extra features evaluation

**Vibration patterns per dot** were considered of few utility for all participants. Most of them was not able to identify dot's number according only to the vibration, specially for values greater than 3. It was consensus that vibrations had time gaps too short to allow clear distinction and could be annoying. One participant, though, highlights that it could allow minimum feedback in some contexts where audio usage is not possible. Due to its disapproval, this feature was left turned off during the prototypes evaluation.

**Multi frequence dual tone feedback** was scored with mean 3.86 (SD 1.34). Overall participants argue that it's usability for distinguishing the state of a button (deactivating dots does not emit any sound, only the standard haptic feedback). It does provide a familiar experience to old keypads. Notwithstanding, none of the participants declares it would be enough to identify which of the dots was activated according only to the sound. This is why one of the participants gave up this method a very low score, even though he approved its use. After experiencing this feature, all participants opted for keeping it activated during the evaluation.

**Screen rotation** scored a mean of 4.29 (SD 0.76). It was agreed among participants as an optional feature, as it only changes the position in which watch has to be hold, but its evaluation raised a much more important discussion regarding braille dot layout, which will be explained during next session. After experiencing it, two of the participants opted for keeping screen rotated.

## 5   DISCUSSION

We discuss some highlighted analysis during experiments.

One of our main concerns was if users would be able to familiarize themselves with the proposed layout, in a way that they could
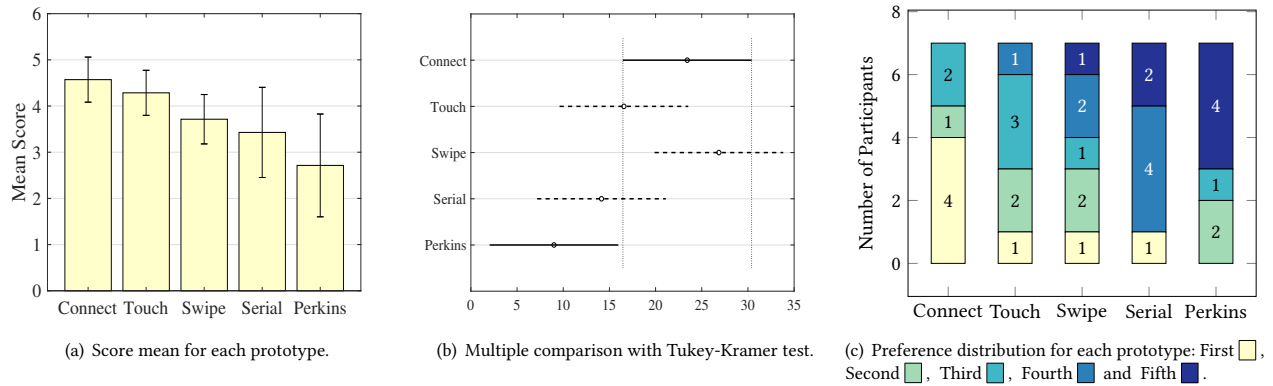
(a) Score mean for each prototype.

(b) Multiple comparison with Tukey-Kramer test.

(c) Preference distribution for each prototype: First ☐, Second ☐, Third ☐, Fourth ☐ and Fifth ☐.

**Figure 3: Results of Prototype Evaluation.**

feel secure to perform clicks even without literal dot number pronunciation. This was proven feasible, as in both Touch and Connect methods, a few minutes of practice lead participants to a comfortable experience. There are, however, challenges, such as the fact that the face of the device used has a considerable area which is non-clickable. This is unperceptive for blind users, which lead to some confusion on whether the bottom or top part of the screen were actually being pressed. Some participants suggested the use of markers on this areas to overcome this issue, which could be done with a 3D printed surface, as in [27]. It is, still a hardware-specific issue, as other watches have thinner or zero border.

However, having a spatial awareness of where each dot is on screen may not be enough to provide a good typing experience. As it was reported during the screen rotation evaluation, dots layout is a key feature for users. We forced participants to work with only one disposition: first column 1-2-3 and second column 4-5-6, from top to bottom. Three participants, although, argued that they would prefer to have 4-5-6 on the first column and 1-2-3 on the second, as this is the order used with slate and stylus, the common method for writing Braille on paper. This affects those literate on Braille from childhood. Two participants asked for a different layout: 3-2-1 on first column, 6-5-4 on the second, due to their use of the Perkins machine and Braille keyboards on smartphones that had this layout (the work on [28] seems to inspire these methods). For such insights, we believe user should be able to decide which layout makes more sense to its Braille composition.

Swipe gestures were proved to work well with a group of participants, but not so well with others. We analyze that the way some participants swiped was slower and applied more pressure, possibly a consequence of their use of hands for overall tactile feedback in daily routine. This, added to the fact that many times the gesture begun on the non-clickable area of the watch, could be the cause of the lack of detection. Due to that, we now believe that swipe gestures on watch face should assume more secondary actions.

Our evaluation achieved its objective. Many of the proposed methods seemed interesting from upfront, but once confronted with real users, a better idea of what is or not useful could be obtained. For such task as text entry, complex methods as Serial and Perkins did not work well. As result, Connect and Touch received more positive feedback, being simple to understand and providing fast input once user is secure with layout.

While interacting with prototypes, participants were extremely sincere on exposing their frustrations and excitements. One participant in particular affirms that he feel secure to type with Connect even faster than how he currently types with QWERTY on smartphones. Every participant responded that they would use one of this methods if they had a smartwatch, mostly complementary to voice messages. The necessity of a smartwatch was contested by two participants, which could not see yet sufficient reason to buy one. It was unanimous the support for developing a Braille alternative for text entry on considering both daily use and education.

## 6 FUTURE WORK

Future work plans on addressing current issues and attend to suggestions, such as the aforementioned custom dot layout disposition.

For the best ranked prototypes, we are working on expand their functionality including character erasing, word and character navigation, shortcuts and orthographic corrections. A natural path to take is to provide suggestions to users based on a word model. We aim to conduct a more extensive user study, with more participants and other known input methods to compare efficiency and efficacy.

## 7 CONCLUSION

In this study we explore different interaction methods for blind people to write Braille in a smartwatch. Five different prototypes were evaluated and three extra features were implemented for complementing the interactions. Seven specialists participated on the experiment, providing qualitative feedback on the proposed methods. Three of the prototypes were considered acceptable for future evaluation: Connect, Touch and Swipe. Among the extra features, dual tone generator and screen rotation were elected useful.

We believe the discussion here made can be important not only to the extension of our project but also for other researches that aim for alternatives on text entry in smartwatches for blind people. Hopefully we can find solutions that truly attend to these users, and provide them power to make use of new technology.

## ACKNOWLEDGMENTS

# REFERENCES

[1] Mohammad M. Albanna. 2017. SwiftBraille - Connect The Dots. Retrieved December 24, 2018 from http://en.swiftbraille.com/

[2] Maraim Alnfiai and Srinivas Sampalli. 2016. An Evaluation of SingleTapBraille Keyboard: A Text Entry Method That Utilizes Braille Patterns on Touchscreen Devices. In *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '16)*. ACM, New York, NY, USA, 161–169. https://doi.org/10.1145/2982142.2982161

[3] Shiri Azenkot, Jacob O. Wobbrock, Sanjana Prasain, and Richard E. Ladner. 2012. Input Finger Detection for Nonvisual Touch Screen Text Entry in Perkinput. In *Proceedings of Graphics Interface 2012 (GI '12)*. Canadian Information Processing Society, Toronto, Ont., Canada, Canada, 121–129. http://dl.acm.org/citation.cfm?id=2305276.2305297

[4] Jae-Min Cha, Eunjung Choi, and Jihyoun Lim. 2015. Virtual Sliding QWERTY: A new text entry method for smartwatches using Tap-N-Drag. *Applied Ergonomics* 51 (Nov. 2015), 263–272. https://doi.org/10.1016/j.apergo.2015.05.008

[5] Xiang 'Anthony' Chen, Tovi Grossman, and George Fitzmaurice. 2014. Swipeboard: a text entry technique for ultra-small interfaces that supports novice to expert transitions. ACM Press, 615–620. https://doi.org/10.1145/2642918.2647354

[6] Daniel Dalton. 2016. SoftBrailleKeyboard. Retrieved December 24, 2018 from https://docs.google.com/document/d/1OX6yaFjkK61VfROdU9uQNxjEZ_QNWxiMnUY_1ECnIyRQ/pub

[7] Aritra Dhar, Aditya Nittala, and Kuldeep Yadav. 2016. TactBack: VibroTactile Braille Output Using Smartphone and Smartwatch for Visually Impaired. In *Proceedings of the 13th Web for All Conference (W4A '16)*. ACM, New York, NY, USA, 13:1–13:2. https://doi.org/10.1145/2899475.2899514 00000.

[8] Brian Frey, Kate Rosier, Caleb Southern, and Mario Romero. 2012. From Texting App to Braille Literacy. In *CHI '12 Extended Abstracts on Human Factors in Computing Systems (CHI EA '12)*. ACM, New York, NY, USA, 2495–2500. https://doi.org/10.1145/2212776.2223825 00000.

[9] Mitchell Gordon, Tom Ouyang, and Shumin Zhai. 2016. WatchWriter: Tap and Gesture Typing on a Smartwatch Miniature Keyboard with Statistical Decoding. ACM Press, 3817–3821. https://doi.org/10.1145/2858036.2858242

[10] Aakar Gupta and Ravin Balakrishnan. 2016. DualKey: Miniature Screen Text Entry via Finger Identification. ACM Press, 59–70. https://doi.org/10.1145/2858036.2858052

[11] Jonggi Hong, Seongkook Heo, Poika Isokoski, and Geehyuk Lee. 2015. SplitBoard: A Simple Split Soft Keyboard for Wristwatch-sized Touch Screens. ACM Press, 1233–1236. https://doi.org/10.1145/2702123.2702273

[12] Min-Chieh Hsiu, Da-Yuan Huang, Chi An Chen, Yu-Chih Lin, Yi-ping Hung, De-Nian Yang, and Mike Chen. 2016. ForceBoard: using force as input technique on size-limited soft keyboard. ACM Press, 599–604. https://doi.org/10.1145/2957265.2961827

[13] Dot Inc. 2018. Dot Smartwatch. Retrieved December 24, 2018 from https://dotincorp.com/

[14] Shaun K. Kane, Jacob O. Wobbrock, and Richard E. Ladner. 2011. Usable Gestures for Blind People: Understanding Preference and Performance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, New York, NY, USA, 413–422. https://doi.org/10.1145/1978942.1979001 00206.

[15] Mateus M. Luna, Fabrizzio A. A. M. N. Soares, Hugo A. D. Nascimento, Joyce Siqueira, Eeduardo F. Souza, Thamer H. Nascimento, and Ronarldo M. Costa. 2018. Text Entry on Smartwatches: A Systematic Review of Literature. In *2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, Vol. 02. 272–277. https://doi.org/10.1109/COMPSAC.2018.10242

[16] Sergio Mascetti, Cristian Bernareggi, and Matteo Belotti. 2012. TypeInBraille: Quick Eyes-Free Typing on Smartphones. In *Computers Helping People with Special Needs*, Klaus Miesenberger, Arthur Karshmer, Petr Penaz, and Wolfgang Zagler (Eds.). Lecture Notes in Computer Science, Vol. 7383. Springer Berlin Heidelberg, 615–622. https://doi.org/10.1007/978-3-642-31534-3_90

[17] Elke Mattheiss, Georg Regal, Johann Schrammel, Markus Garschall, and Manfred Tscheligi. 2014. Dots and Letters: Accessible Braille-Based Text Input for Visually Impaired People on Mobile Touchscreen Devices. In *Computers Helping People with Special Needs*, Klaus Miesenberger, Deborah Fels, Dominique Archambault, Petr Penaz, and Wolfgang Zagler (Eds.). Lecture Notes in Computer Science, Vol. 8547. Springer International Publishing, 650–657. https://doi.org/10.1007/978-3-319-08596-8_100

[18] Aske Mottelson, Christoffer Larsen, Mikkel Lyderik, Paul Strohmeier, and Jarrod Knibbe. 2016. Invisiboard: maximizing display and input space with a full screen text entry method for smartwatches. ACM Press, 53–59. https://doi.org/10.1145/2935334.2935360

[19] Thamer Hobylon Nascimento, Fabrizzio Alphonsus A. M. N. Soares, Pourang Polad Irani, Leandro Luiz Galdino de Oliveira, and Anderson da Silva Soares. 2017. Method for Text Entry in Smartwatches Using Continuous Gesture Recognition. IEEE, 549–554. https://doi.org/10.1109/COMPSAC.2017.168

[20] João Oliveira, Tiago Guerreiro, Hugo Nicolau, Joaquim Jorge, and Daniel Gonçalves. 2011. Blind People and Mobile Touch-based Text-entry: Acknowledging the Need for Different Flavors. In *The Proceedings of the 13th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '11)*. ACM, New York, NY, USA, 179–186. https://doi.org/10.1145/2049536.2049569

[21] João Oliveira, Tiago Guerreiro, Hugo Nicolau, Joaquim Jorge, and Daniel Gonçalves. 2011. BrailleType: Unleashing Braille over Touch Screen Mobile

[22] Phones. In *Proceedings of the 13th IFIP TC 13 International Conference on Human-computer Interaction - Volume Part I (INTERACT'11)*. Springer-Verlag, Berlin, Heidelberg, 100–107. http://dl.acm.org/citation.cfm?id=2042053.2042066

[22] Stephen Oney, Chris Harrison, Amy Ogan, and Jason Wiese. 2013. ZoomBoard: a diminutive qwerty soft keyboard using iterative zooming for ultra-small devices. ACM Press, 2799. https://doi.org/10.1145/2470654.2481387

[23] Stefania Pizza, Barry Brown, Donald McMillan, and Airi Lampinen. 2016. Smartwatch in Vivo. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 5456–5469. https://doi.org/10.1145/2858036.2858522 00000.

[24] Franck Poirier and Mohammed Belatar. 2016. UniWatch: A Soft Keyboard for Text Entry on SmartWatches Using 3 Keys. In *Human-Computer Interaction. Interaction Platforms and Techniques*, Masaaki Kurosu (Ed.). Vol. 9732. Springer International Publishing, Cham, 341–349. http://link.springer.com/10.1007/978-3-319-39516-6_33 DOI: 10.1007/978-3-319-39516-6_33.

[25] Tomoki Shibata, Daniel Afergan, Danielle Kong, Beste F. Yuksel, I. Scott MacKenzie, and Robert J.K. Jacob. 2016. DriftBoard: A Panning-Based Text Entry Technique for Ultra-Small Touchscreens. ACM Press, 575–582. https://doi.org/10.1145/2984511.2984591

[26] Joyce Siqueira, Fabrizzio Alphonsus Alves de Melo Nunes Soares, Deller James Ferreira, Cleyton Rafael Gomes Silva, Luciana de Oliveira Berretta, Cristiane Bastos Rocha Ferreira, Igor Moreira Felix, Anderson da Silva Soares, Ronaldo Martins da Costa, and Mateus Machado Luna. 2016. Braille Text Entry on Smartphones: A Systematic Review of the Literature. In *2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC)*, Vol. 2. 521–526. https://doi.org/10.1109/COMPSAC.2016.74

[27] Joyce Siqueira, Fabrizzio Alphonsus Alves de Melo Nunes Soares, Cleyton Rafael Gomes Silva, Luciana de Oliveira Berretta, Cristiane Bastos Rocha Ferreira, Igor Moreira Felix, and Mateus Machado Luna. 2016. BrailleÉcran: A Braille Approach to Text Entry on Smartphones. In *2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC)*, Vol. 2. 608–609. https://doi.org/10.1109/COMPSAC.2016.5

[28] Caleb Southern, James Clawson, Brian Frey, Gregory Abowd, and Mario Romero. 2012. An Evaluation of BrailleTouch: Mobile Touchscreen Text Entry for the Visually Impaired. In *Proceedings of the 14th International Conference on Human-computer Interaction with Mobile Devices and Services (MobileHCI '12)*. ACM, New York, NY, USA, 317–326. https://doi.org/10.1145/2371574.2371623

[29] J. Timmermann, W. Heuten, and S. Boll. 2015. Input methods for the Borg-RPE-scale on smartwatches. In *2015 9th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth)*. 80–83. https://doi.org/10.4108/icst.pervasivehealth.2015.259220

[30] Colton J. Turner, Barbara S. Chaparro, and Jibo He. 2016. Text Input on a Smartwatch QWERTY Keyboard: Tap vs. Trace. *International Journal of Human–Computer Interaction* (Aug. 2016), 1–8. https://doi.org/10.1080/10447318.2016.1223265

[31] K. Ushida, Y. Sekine, and S. Hasegawa. 2014. IPPITSU: A one-stroke text entry method for touch panels using Braille system. In *2014 IEEE 3rd Global Conference on Consumer Electronics (GCCE)*. 374–375. https://doi.org/10.1109/GCCE.2014.7031257

[32] Keith Vertanen, Haythem Memmi, Justin Emge, Shyam Reyal, and Per Ola Kristensson. 2015. VelociTap: Investigating Fast Mobile Text Entry using Sentence-Based Decoding of Touchscreen Keyboard Input. ACM Press, 659–668. https://doi.org/10.1145/2702123.2702135

[33] Aku Visuri, Zhanna Sarsenbayeva, Niels van Berkel, Jorge Goncalves, Reza Rawassizadeh, Vassilis Kostakos, and Denzil Ferreira. 2017. Quantifying Sources and Types of Smartwatch Usage Sessions. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 3569–3581. https://doi.org/10.1145/3025453.3025817 00000.

[34] Xin Yi, Chun Yu, Weijie Xu, Xiaojun Bi, and Yuanchun Shi. 2017. COMPASS: Rotational Keyboard on Non-Touch Smartwatches. ACM Press, 705–715. https://doi.org/10.1145/3025453.3025454

[35] B. Šepić, A. Ghanem, and S. Vogel. 2015. BrailleEasy: One-handed Braille Keyboard for Smartphones. *Studies in Health Technology and Informatics* 217 (2015), 1030–1035. https://doi.org/10.3233/978-1-61499-566-1-1030 00000.