This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2019.2898736, IEEE Transactions on Visualization and Computer Graphics

RingText: Dwell-free and hands-free Text Entry for Mobile Head-Mounted Displays using Head Motions

Wenge Xu, Hai-Ning Liang, Yuxuan Zhao, Tianyu Zhang, Difeng Yu, Diego Monteiro, and Yong Yue

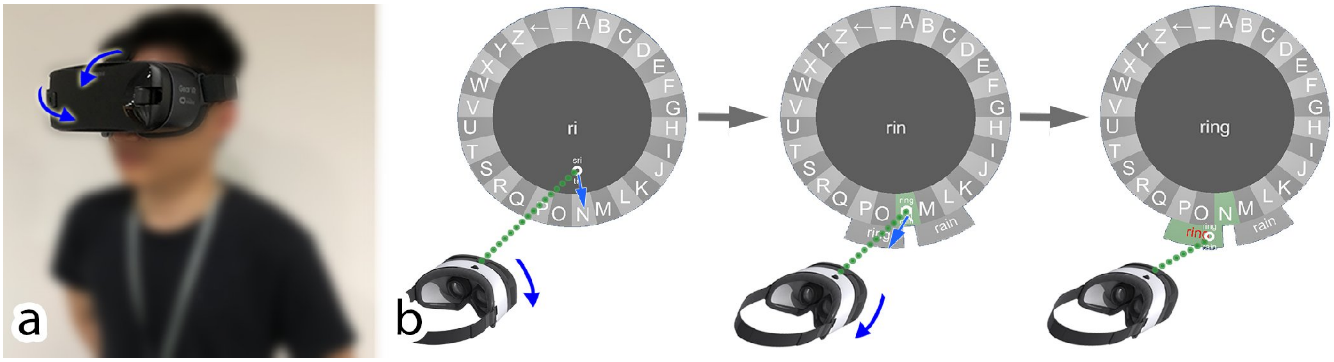

Fig. 1. (a) Text entry on a mobile head-mounted display through head motions; (b) To finish typing ‘ring’ after a user has already entered the letters ‘r’ and ‘i’, the user selects the letter ‘n’. The entered text is shown in the center of the screen; two candidate words are shown in the regions below and on each side of the last letter ‘N’. Then the user goes to select the recommended word ‘ring’ by moving the head down. The design rationale of the technique is to minimize eye and head movements (or distance traveled), but still maintain a reasonably low error rate, of users of mobile virtual reality head-mounted displays.

Abstract—In this paper, we present a case for text entry using a circular keyboard layout for mobile head-mounted displays (HMDs) that is dwell-free and does not require users to hold a dedicated input device for letter selection. To support the case, we have implemented RingText whose design is based on a circular layout with two concentric circles. The outer circle is subdivided into regions containing letters. Selection is made by using a virtual cursor controlled by the user’s head movements—entering a letter region triggers a selection and moving back into the inner circle resets the selection. The design of RingText follows an iterative process, where we initially conduct one first study to investigate the optimal number of letters per region, inner circle size, and alphabet starting location. We then optimize its design by selecting the most suitable features from the first study: one letter per region, narrowing the trigger area to lower error rates, and creating candidate regions that incorporate two suggested words to appear next to the current letter region (close to the cursor) using a dynamic (rather than fixed) approach. Our second study compares text entry performance of RingText with four other hands-free techniques and the results show that RingText outperforms them. Finally, we run a third study lasting four consecutive days with 10 participants (5 novice users and 5 expert users) doing two daily sessions and the results show that RingText is quite efficient and yields a low error rate. At the end of the eighth session, the novice users can achieve a text entry speed of 11.30 WPM after 60 minutes of training while the expert (more experienced) users can reach an average text entry speed of 13.24 WPM after 90 minutes of training.

Index Terms—Virtual Reality, text entry, circular keyboard layout, mobile head-worn/mounted displays, dwell-free input.

1 Introduction

Mobile virtual reality (VR) head-mounted displays (HMDs) allow users to perceive and interact with immersive virtual environments anytime and anywhere, through the use of smartphones, whose sensors can capture input commands from users [1]. This enables a new interaction scenario called Nomadic VR [2] where a user could operate a Mobile VR HMD in an “uninstrumented” environment and often in public areas (e.g. on the subway/bus, library, or coffee shop). Since most current-generation smartphones can be converted into VR headsets, a notable number of consumer versions of mobile VR HMDs (such as Samsung Gear VR and Google Cardboard) are now marketed to the masses.

Although mobile VR HMDs typically come with a controller device, there are cases where users cannot access the controller; for example, the controller is not around, or the users’ hands are occupied with other activities. Besides, hands-free input will be useful for users who cannot manipulate a controller at all or with the precision required for text entry. Users who do not possess sufficient hand motor control skills like elderly users or those who have a motor deficiency disease will benefit from a hands-free technique. In this sense, having a technique that does not require users’ hands to hold a device for input can come in handy in a variety of situations and for various types of users and AR/VR devices.

Development of efficient text entry methods for HMDs without any dedicated handheld device has remained unexplored. A recent paper [3] reports a head-based text entry technique with dwell time that allows users to achieve an average of 10.59 word-per-minute (WPM) after training for 50 minutes. One limitation observed from their data is that the slowest users cannot improve much, even after having training. Another limitation is the dwell technique itself; it is well-known that dwell-based techniques can limit typing speed

- *Wenge Xu, Hai-Ning Liang, Yuxuan Zhao, Tianyu Zhang, Difeng Yu, Diego Monteiro, and Yong Yue are with Xi’an Jiaotong-Liverpool University, Suzhou, Jiangsu, China.*
- *Corresponding author: Hai-Ning Liang; E-mail: HaiNing.Liang @xjtlu.edu.cn*

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org.
Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx/.

because of an imposed waiting time for each character selection. Text entry rates of dwell-based methods are typically between 5 to 10 WPM [4]. By eliminating dwell time and optimizing the layout for selecting not only the letters but also the recommended spelling correction words, it is possible to increase WPM.

In this paper, we explore the feasibility of applying a circular keyboard layout with two concentric areas for text entry that is both dwell-free and hands-free for mobile VR HMDs (see Fig. 1 for a picture of the technique and how it works). We have conducted three studies. The first study evaluates and compares how three possible factors (number of letters per region of the outer circle, size of the inner circle for resetting selection, and alphabet starting position) affect the efficiency of text entry, error rates, workload, and simulator sickness. Informed by both quantitative and subjective data, we then have improved and optimized the best layout (and features) from the first study further by narrowing the letter trigger area, adding a spelling correction feature, and incorporating dynamic, instead of fixed, candidate word regions for fast selection. Unlike other techniques that show the recommended candidate words in a fixed position [3], [5], our dynamic candidate regions are designed based on Fitts' law [6] to enable users to choose quickly the desired word suggested by a spelling correction algorithm. In a second study, we have compared the text entry performance of our technique, RingText, with four other possible techniques: dwell QWERTY, dwell circular, Swype circular, and Swype QWERTY—the results show that RingText outperforms them. Finally, we have conducted a 4-day study with two daily sessions and 10 participants to evaluate the learning effects of RingText on speed and error rates. Our last session results indicate that the five novice users can achieve an average of 11.30 WPM (s.e. = 0.80) with 3.29% (s.e. = 0.34%) of the total error rate, and that the five 'expert' users (those who had performed the best in the second study) can achieve an average of 13.24 WPM (s.e. = 0.80) with 2.90% (s.e. = 0.22%) of the total error rate. Our results also show that our technique leads to a high selection rate of the recommended words due to the use of dynamic recommended word regions.

The contributions of this work include: (1) the first example of a formal evaluation of the circular keyboard layout for text input in VR; (2) the first comparison of hands-free text entry mechanisms for both circular and QWERTY keyboard layouts in VR; (3) a case for the use of dynamic (rather than static) locations for recommended words—to our knowledge, this is a first case that shows the usefulness of using dynamic locations of these words; and (4) a demonstration of the effectiveness of RingText, a circular layout text entry technique that relies on head motions and uses dynamic locations for recommended words, through a 4-day user study.

2 Related Work

In this section, we provide the literature review with respect to text entry for mobile VR HMDs; dwell-free text entry techniques; circular layouts; and dynamic vs. fixed positioning and use patterns of candidate words.

2.1 Text Entry for Mobile VR HMDs

One of the biggest challenges for mobile VR HMD is to avoid the need of the peripheral devices generally used in stationary VR systems such as keyboards and mice [7] and game controllers [8]. This "accessory constraint" poses extra difficulties for text entry in immersive virtual environments (IVE) and limits the use of not only VR and also AR HMDs.

One possible solution is to use speech-based text entry techniques. Bowman et al. [9] made a comparison among a speech-based text entry, a pen and tablet keyboard metaphor, a one-hand chording keyboard, and pinch gloves, and found that the speech technique is the fastest medium for entering text in IVE at around 14 WPM. A recent speech-based multimodal text-entry system called SWIFTER [10] has claimed to reach an average input rate of 23.6 WPM. Despite their potential use in text entry, one major limitation of speech recognition techniques is that their performance suffers in noisy environments [7]. Furthermore, they can bring privacy problems when the user uses a speech text entry method to input a password or send messages to friends in a public environment, like a bus, coffee shop, or library. This represents a severe shortcoming for mobile VR HMDs which are often operated in an "uninstrumented" environment or public areas.

Other researchers have investigated touchscreen-based text entry techniques [11]–[13] and reported fairly good entry speeds (e.g. 17-23 WPM with a prediction algorithm [13]). However, because users are not able to precisely locate their hands before the first press in IVE [11], the typing process might require extra movements for selecting the target characters. Moreover, since a smartphone might already be used as a display for the mobile VR HMD, an extra touchpad is required for text input, and the use of hands is needed, something that is not possible in situations where users' hands are occupied.

Numerous mid-air typing techniques have been explored for virtual environments including wearable glove-based techniques [9] and motion tracking techniques [14]. Although such techniques enable mobile text entry and some of them allow a fast text entry speed (23 WPM for novice users as reported in [14]), these techniques might require expensive extra sensors or devices like cameras or sensor-equipped gloves. In addition, most of them require a substantial learning curve [7] and may confine users to a fixed location and position.

Current common mobile VR HMDs are designed to be operated using head rotation [1], [11] by which users can move the cursor placed in the middle of the view to select target objects. Yu et al. [3] proposed and explored three types of text entry techniques using head-based interaction: Tap, Dwell, and Gesture with text entry speeds of 10.59, 15.58, and 19.04 WPM respectively for novice users after 6 training sessions. Among them, only their Dwell technique requires no extra device. Further, the input speed of their Dwell technique is not that high even with a prediction and error-correction algorithm (10.59 WPM). For these reasons, one of our key motivations is to propose a more efficient head-based device-free technique for mobile VR HMDs. Our design will eliminate dwell time and avoid the need of using hands (or additional input devices). More importantly, we aim to reduce motion sickness of users by minimizing the need to make large head motions.

2.2 Dwell-free Text Entry Techniques

Instead of dwelling on the target for a predetermined duration to trigger a selection [15], dwell-free techniques allow users to enter text on-the-fly. Kristensson and Vertanen [16] investigated an eye gaze dwell-free text entry approach in a non-VR scenario and indicated that dwell-free eye typing could theoretically be significantly faster than existing techniques with a theoretical text entry speed of 46 WPM. Although this result is based on an error-free simulation, it suggests a possible research direction for dwell-free text entry techniques.

Dwell-free typing techniques can be divided into two major groups: gesture-based and selection-based. EyeWrite [17], the first gesture-based eye typing technique, has been shown to be significantly faster, easier to use, and prone to cause less ocular fatigue than the on-screen keyboard [18]. Eye-S [19] allows users to draw letters through sequential movements on nine hotspots and is claimed to reach 6.8 WPM for expert users. A later eye-typing technique, EyeSwipe [20], enables users to glance at the vicinity of the respective characters in the middle of the word but carefully selects the first and last characters of a word using the "reverse crossing" technique. It can reach 11.7 WPM on average for ten participants with 30 minutes of training. This technique is not fully dwell-free since it requires users to look at the hotspot for a pre-defined threshold time to confirm the sequence starting point. Gesture-based techniques are shown to suffer from low-performance issues [21].

Several selection-based dwell-free typing techniques have also been proposed. EyeK [22] allows users to select a character by moving the pointer inside-outside-inside the activation area. The authors have claimed it can achieve an average speed of 6.03 WPM. Filteryedping [23] can filter out unintentionally triggered letters from the sequence of letters swiped by the user and predicts the possible words. This technique is reported to achieve an average text entry speed of 14.75 WPM. One common drawback for most of these selection-based dwell-free techniques is that they might require extra movements to type the word (e.g. inside-outside-inside movements [22]). When used in HMDs this additional movement can increase motion sickness, which instead should be reduced.

There are some recent developments for VR HMD with eye tracking but the cost of such devices is much higher than the standard HMD. For instance, the price of a FOVE 0 is $599 USD which is 7 times higher than the Samsung Gear VR ($76) and also higher than other PC HMDs (i.e. Oculus CV1 – $399). Also, some research (e.g. [24]) suggests that head-based typing is as fast as gaze typing but can induce fewer errors. In line with this, we believe that dwell-free techniques have benefits for head-based text entry, including fast character selection, less error-prone than gaze typing, and high levels of acceptance by mobile VR HMD users.

2.3 Circular Layout

2.3.1 Circular Keyboard Layout

The circular keyboard is first designed to work with pen input for desktops and touchscreen phones (e.g. Cirrin [25]). Later circular keyboard styles are designed to work without the stylus. TUP [26] maps the letters at fixed positions around a circle. Users place their finger on the location of the letters for selection. With the aid of a prediction algorithm, novice users can achieve 6-7 WPM.

The circular layout has also been used in gaze typing. pEYEs [27] employed a hierarchical circular interface with gaze-based input and reported a speed of 7.85 WPM for novice users and 12.33 WPM maximum for an expert user. Topal et al. [28] developed SliceType by applying a language prediction model to merge keys of their inner-outer circle layout. Their method can achieve 3.45 WPM for gaze input with 1 second dwell time. Apart from these works, the circular layout is also used in huge wall displays [29], VR with Dual Thumbsticks controller [8], and smartwatch [5], [30], [31]. So far, the best result for novice users using circular layout is appeared in WrisText [41], participants were able to type as fast as 15.2 WPM at the end of the fifth session.

2.3.2 Hierarchical Marking Menu

A hierarchical marking menu uses a set of multi-level radial menus and "zig-zag" marks to make selections [32]. This design concept has been used in many areas, such as fractal menus for AR HMDs [33] and Swipeboard [34] for smartwatch text entry where users can reach 19.58 WPM after two hours training. However, these examples are not based on a circular layout. Our review shows that there does not seem to be any research that has explored a hierarchical marking menu design with alphabet letters and suggested words using a circular layout.

2.4 Placement of Candidate Words

Auto-complete, recommended words, and spelling corrections are commonly used in both research prototypes [3], [5], [23] and commercial products, like phones and tablets, to show possible words that users are trying to type. These suggested words are typically placed just above the T9 and QWERTY keyboard layouts.

Our review of the literature also shows that not much research has looked at the placement of suggested words for users to choose from. For QWERTY layouts, it is common to find word suggestions to be placed just above [3] or below [23] the virtual keyboard—the assumption seems to be that this placement will lead to fast and accurate selection. In addition, the placement is usually fixed in one region. While fixed placement either above or below the keyboard works for QWERTY layouts, this design may not be the most optimal for other keyboard layouts.

For a circular keyboard layout, placing the candidate words far away from the keyboard [5] makes it difficult for users to check the words and select them. The candidate regions and its selection used in the circular layout on smartwatches are efficient; the user can choose a candidate word by pinching the thumb and index fingers [41] or by pressing a side button [31]. However, these techniques applied in smartwatches are unlikely feasible for hands-free and controller-free HMD text entry scenarios.

Beyond smartwatches, our research points to a lack of research in the design and use of candidate word regions for circular keyboards. Their placement should be such that the user does not need to look back-and-forth between the keyboard and the suggested words, which are updated after each letter entry. Besides, if a cursor or a pointer is used for selection, its placement should aim to reduce the distance between the last selected key on the keyboard and the potential word that the user has in mind. In VR systems when using hands-free and controller-free circular text entry layout, dynamically positioning the suggested words could be a way to minimize the back-and-forth eye movement to check the words and can also reduce the distance (and hence the time) that is needed to make a fast selection. Our technique uses a dynamic location positioning for recommended words and, as described later, results from our experiment show that indeed dynamic placement brings advantages for text entry for circular layouts using head motions for selection.

3 RINGTEXT

3.1 Layout

To achieve dwell-free, our technique divides the boundary of the outer circle into equal size regions to hold the characters (see Fig. 2 below). The region can potentially hold one or more characters. The inner circle can be regarded as the rest/reset area; users can stay at the center, while their eyes are searching for the next letter. To minimize learning, we have organized the letters based on alphabetical order to leverage users' familiarity with this sequencing.

Keyboard size was determined in a pilot study with 8 participants. We rendered the virtual keyboard far away from the user (8 meters) to avoid the parallax effect [3] and tested the keyboard size with a radius of 5, 5.5 and 6 meters in this preliminary study. We employed the 5.5-meter keyboard in our subsequent studies because of these participants' preference.

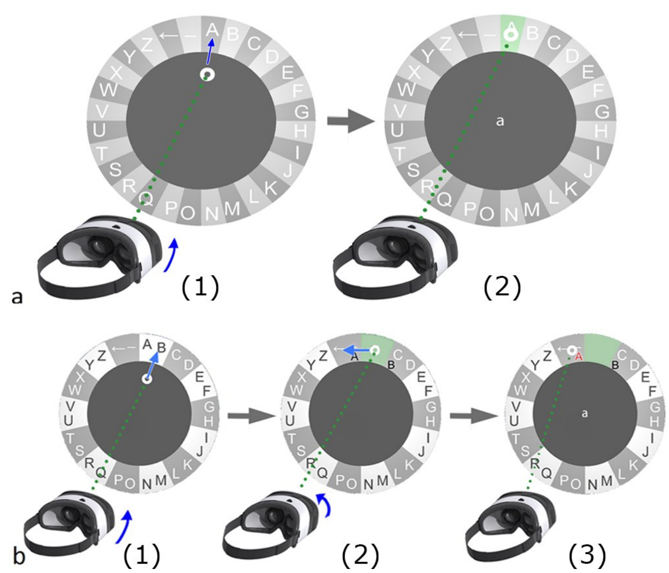

Fig. 2. Design of the layouts and selection mechanism. (a) The 1 letter per region selection mechanism; and (b) The 2 letters per region selection mechanism. In both cases, a user is selecting the letter 'A'.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2019.2898736, IEEE Transactions on Visualization and Computer Graphics

3.2 Selection Technique

In this section, we describe briefly how our selection mechanisms work. First of all, it is important to note that two keyboard layouts are used in our first experiment. The first layout has only one letter per region and the second has two.

The one letter layout uses a simple go-and-hit selection approach—i.e. as the cursor leaves the center entering the region of a letter, this letter is instantly selected (Fig. 2a). Since the second layout has two letters per region, the simple go-and-hit does not work. For this layout, we use the following approach: as the cursor leaves the center entering the two-letter region, these letters are split and parallelly placed opposite to each other just outside the current 2-letter region. The user then chooses the desired letter by moving the cursor towards the letter. As the cursor hits the area, the selection is made (Fig. 2b). The users must move the cursor back to the inner circle to restart the selection process—so to make the process consistent.

We also explored 3- and 4- letter-per-region keyboard designs, which have a selection mechanism similar to the 2 letters per region design. However, participants from our preliminary study believed those two designs to be too complicated to use; besides both led to a high error rate.

3.3 Visual and Sound Feedback

Our technique incorporates a sound effect to notify the user after a letter has been selected. To complement the sound, the colour of the region also changes when the cursor enters the region so that the user knows whether the cursor is in the correct region. Also, the colour of the letter also changes for 0.2 seconds to inform the user that the letter has been selected. The typed words are placed at the center so that user can easily see them.

Additional visual feedback is provided for the 2 letters per region layout. That means that once a region is selected the letters within it will move to their respective nearest neighbours. The new position of the letters serves as a visual guide for the user to know to which direction to rotate their head to make the selection (Fig. 2b (2)).

3.4 Advantages of RingText

Our technique leverages the advantages of small head motions such as low cost and higher accuracy when compared to eye gaze [24], [35]. Also, as the head moves, the eyes can move along, which might help users to perform faster the visual search of letters (and as described later, to find the recommended words). Further, we make use of head movements to eliminate the need for hand-held input devices; useful for a wide range of mobile scenarios when such devices are unavailable or inconvenient to use; it is actually preferred and suggested to use head pointing (or movement) when a hand-held controller is not available (see [36]). Finally, our layout allows us to reduce selection time through dwell-free selection—selection is made only with small head movements.

We next describe the three studies. The first study explores the factors that can influence typing speed and error rates so that they could be optimized in our technique. Study 2 then compares the tuned method with four other hands-free methods to evaluate their relative performance. Finally, Study 3 explores the performance of both novice and "expert" users over a longer training period.

4 Study One

The goal of this experiment was to evaluate the effect of (1) the number of letters per region on the outer circle, (2) the size of the inner circle for resetting the selection, and (3) the starting position of the letters on speed and error rate. We also evaluated workload and simulator sickness.

4.1 Participants and Apparatus

18 participants (13 males and 5 females) between the ages of 18 and 28 (M = 20.83, SD = 2.60) were recruited from a local university

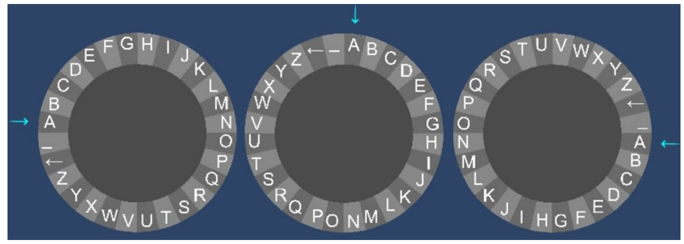

Fig. 3. Three alphabet starting positions. From left to right: alphabet starting on the left; alphabet starting on the top; and alphabet starting on the right.

campus. All participants were familiar with the alphabet because the language of instruction at the university is English but were not native users. Participants had normal or corrected-to-normal vision and reported an average of 4 for experience with the QWERTY keyboard on a scale from 1 ('No Skill') to 5 ('Expert'). 14 participants had previous experience with HMDs before the experiment—they had either seen and/or interacted with them.

The experiment was conducted on a 96-degree field of view Samsung Gear VR with an S6 Edge+ smartphone. Unity3D was used to develop and implement our proposed head-based text entry technique. Our application also logged the cursor movement data for further analysis (like the heat map of selection areas).

4.2 Design

The experiment used a 2 × 2 × 3 within-subjects design with three independent variables. The first was the number of letters per region (LPR) which had two levels: 1 LPR and 2 LPR. The second was the inner circle size (Center Size) which had two levels: Large (65% of the whole circular layout size—3.575-meters) and Small (55% of the whole circular layout size—3-meters). The last variable was the alphabet starting position: Left, Top, and Right (see Fig. 3).

LPR and Center Size were counterbalanced; the alphabet starting positions were randomly assigned but also balanced for each condition. All three alphabet starting positions were tested by each participant. Each layout was randomly tested by 6 participants.

Each participant transcribed 8 phrases for each layout combination. All phrases were randomly sampled from the MacKenzie's phrase set [37] with no repeated phrases within the session. Each phrase was displayed in the central area. The Gear VR touchpad was applied only for the user to switch to the next phrase. Text entry speed was measured in WPM, with a word defined as five consecutive letters, including spaces. The error rate was calculated based on the standard typing metrics [38], where the total error rate (TER) = not corrected error rate (NCER) + corrected error rate (CER).

4.3 Procedure

Before each session, all participants were briefed about the experiment details; then a 1-minute training was provided for the participants before each layout to allow them to familiarize with it. After each layout, the participants were asked to fill the NASA-TLX [39] and simulator sickness questionnaire (SSQ) [40]. Because our technique required frequent neck motions, we also added additional Neck Fatigue questions to SSQ. A 1-minute break was given if the participant felt tired. Before the experiment ended, all participants were asked to choose their preferred layout (LPR × Center Size) and alphabet starting position. This experiment took on average 45 minutes per participant. In total, we collected 18 participants × 2 Center Sizes × 2 LPR × 8 phrases = 576 phrases.

4.4 Results

We employed a mixed factorial ANOVA and Bonferroni corrections for pair-wise comparisons. We also used a Greenhouse-Geisser adjustment to correct for violations of the sphericity assumption. Effect sizes were reported whenever feasible (η_p^2).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2019.2898736, IEEE Transactions on Visualization and Computer Graphics

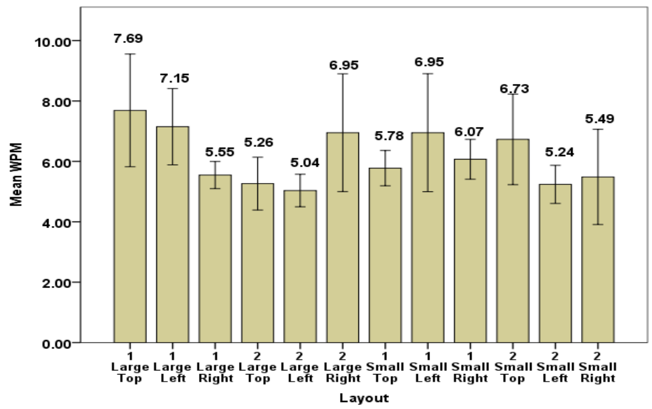

Fig. 4. Mean text-entry speed across 12 types of RingText layouts. Error bars indicate ± 2 standard errors.

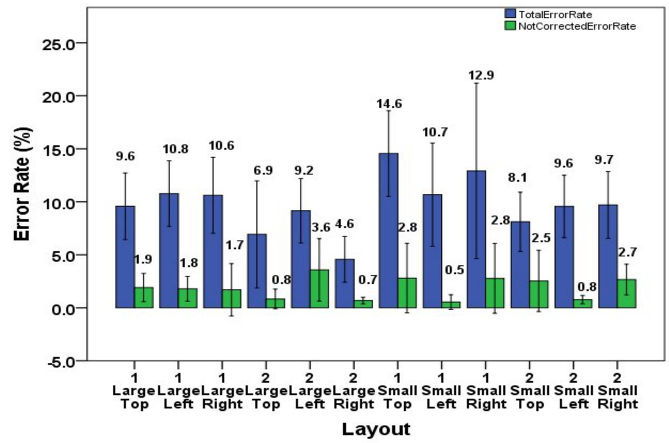

Fig. 5. Mean TER and NCER across 12 types of RingText layouts. Error bars indicate ± 2 standard errors.

4.4.1 Text Entry Speed

Figure 4 illustrates mean text entry speed for each layout. A 2 × 2 × 3 (LPR, Center size, alphabet starting position) ANOVA tests revealed a significant difference of LPR ($F_{1,60} = 4.042$, $p < .05$, $\eta_p^2 = .063$, observed power = .507), LPR × alphabet starting position ($F_{2,60} = 3.254$, $p < .05$, $\eta_p^2 = .098$, observed power = .598) and Center Size × LPR × alphabet starting position ($F_{2,60} = 4.364$, $p < .05$, $\eta_p^2 = .127$, observed power = .734) on WPM. No other factors were found to have a significant effect on WPM.

Post-hoc pairwise comparisons for LPR indicated that WPM for 1 LPR was significantly higher than 2 LPR ($p < .05$). Post-hoc pairwise comparisons for LPR × alphabet starting position indicated that the text entry rate in 1 LPR Left was significantly higher ($p < .01$) than 2 LPR Left. No other significant differences were found. To test for significant effects on Center Size × LPR × alphabet starting position, we made pairwise comparisons which revealed that participants were significantly ($p < .01$) faster when typing with 1 LPR Large Top than 2 LPR Large Top. Also, participants were significantly faster ($p < .05$) when typing with 1 LPR Large Left than 2 LPR Large Left. Additionally, 1 LPR Large Top led to significantly faster ($p < .05$) speed than 1 LPR Small Top. No other significant differences were found.

4.4.2 Error Rate

Figure 5 shows TER and UCER for each layout. ANOVA testes revealed a significant difference of LPR on TER ($F_{1,60} = 8.601$, $p < .01$, $\eta_p^2 = .125$, observed power = .823), while Center Size had a close to significant effect on TER ($F_{1,60} = 3.739$, $p = .058$, $\eta_p^2 = .059$, observed power = .477). No other significant differences were found on TER. No main effects were found to be significant on NCER. Center Size × alphabet starting position was the only interaction effect to be significant on NCER ($F_{2,60} = 3.683$, $p < .05$, $\eta_p^2 = .109$, observed power = .656). Post-hoc pairwise comparisons revealed the Large Left layouts (M = 2.69%, s.e. = 0.80%) had a close to significant ($p = .055$) more NCER than Small Left layouts (M = 0.66%, s.e. = 0.19%).

4.4.3 Subjective Feedback

NASA-TLX. ANOVA tests showed that there was no significant difference of Center Size ($F_{1,60} = 0.003$, $p = .910$, $\eta_p^2 = .000$, observed power = .051), LPR ($F_{1,60} = 2.021$, $p = .160$, $\eta_p^2 = .038$, observed power = .327) and alphabet starting position ($F_{2,60} = 0.048$, $p = .954$, $\eta_p^2 = .001$, observed power = .056) on the overall workload and its subscales (Mental, Physical, Temporal, Performance, Effort, Frustration). No interaction effects were found either.

Simulator Sickness. ANOVA tests yielded no significant difference of Center Size ($F_{1,60} = .265$, $p = .609$, $\eta_p^2 = .004$, observed power = .080), LPR ($F_{1,60} = .009$, $p = .923$, $\eta_p^2 = .000$, observed power = .051) and alphabet starting position ($F_{2,60} = .675$, $p = .513$, $\eta_p^2 = .022$, observed power = .158) on the overall simulator sickness scores and the subscales (Nausea, Oculomotion, Disorientation). No interaction effects were found on the overall simulator sickness scores and its subscales.

User Preference. 15 participants (out of 18) preferred the alphabet to start on the Top; 2 users on the Right; and 1 user on the Left. In terms of the layout, 7 participants selected 2 LPR with the small inner circle; 6 users chose 1 LPR with the large inner circle; 2 participants preferred 1 LPR with the small inner circle; 3 participants selected 2 LPR with the large inner circle.

4.5 Discussion

Because all layouts have similar simulator sickness and TLX workload, we discounted the results. We only considered the performance data, users' preference and comments to decide the final layout and select the features that would be optimized and tested in the next experiment.

Overall, 1 LPR was significantly faster than 2 LPR; TER could be potentially solved by a spelling correction algorithm—our results in the next experiments would support this. No significant difference was found between 1 LPR and 2 LPR on NCER. In addition, all participants commented that 1 LPR is much easier to understand and use than 2 LPR. Therefore, we decided to use 1 LPR layout.

Although Center Size only had a close to significant difference on TER, the results showed a reliable trend that a large center should result in lower TER. Thus, we decided to use the large inner circle to minimize the possibility of inducing errors.

Regarding the alphabet starting position, because it did not have any significant difference on WPM and error rates, we chose the alphabet starting at the top based on user preferences. Thus, the final layout we selected was the 1 LPR large center with the alphabet starting at the top.

During the data analysis, we also observed that selecting a letter that was next to the intended one was the main reason why error rates were high. For example, one of our participants wanted to delete an erroneously selected letter. He then moved to the delete letter region, but unintentionally entered the space region twice because the trigger area for the space region and the delete region were very close to each other. To overcome this problem, we decided to narrow the letter region trigger area for the 1 LPR layout; by doing this, we believe it could help reduce the TER and lead to a faster text entry speed than the Dwell Type approach.

Besides, our observations also suggested that if the technique could include a spelling correction method, it would minimize erroneous inputs, thereby reducing the time that participants would need to correct them. As such, it could potentially increase text entry speed.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2019.2898736, IEEE Transactions on Visualization and Computer Graphics

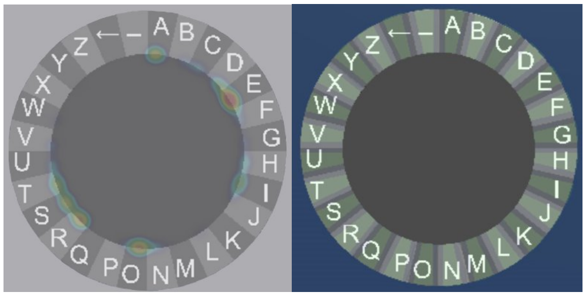

Fig. 6. (a; left) An example of a heatmap of triggered locations. (b; right) Smaller trigger area of the letter regions used in Study 2 and 3.

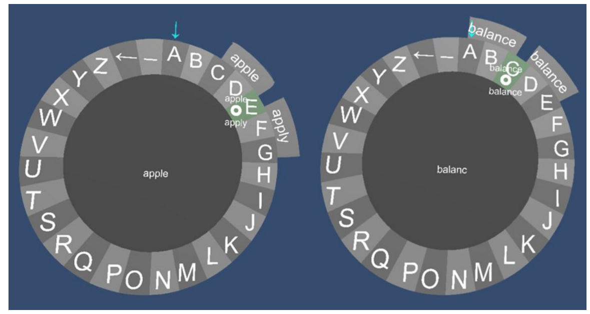

Fig. 7. Dynamic candidate word locations for the letters 'C' and 'E' regions. The two results of the spelling correction algorithm are displayed next to the current letter region and close to the cursor to minimize not only eye movement for checking the words but also head movement for rapid selection of the words.

5 Optimized Design

5.1 Narrower Trigger Area

Figure 6 (a; left) presents the heatmap of the letter triggered locations collected from one participant. It shows that triggered locations are not in the midsections of the border adjacent to the inner circle, but across the whole border areas. Since the trigger areas are very close to each other, users may not find it easy to hit the intended letter region when they are not familiar with the circular layout of RingText, thus leading to error rates that are inevitably high. As shown in Figure 6 (b; right), to lower error rates due to accidental erroneous selections, a narrower trigger area for each letter is used (20% smaller than the original size).

5.2 Spelling Correction

To further improve the performance of our text entry technique, SymSpell [41] was adopted with a dictionary of the ten thousand most frequently used English words [42]. To predict a word more precisely, we only allowed the algorithm to have its maximum search distance just two letters and return the top two spelling suggestions for the current typed letters. Figure 7 shows two examples of recommended words for two sets of letters.

5.3 Fixed vs Dynamic Candidate Word Locations

We also explored whether to use a fix location to show the spelling corrections or to have the locations change dynamically so that they would be shown based on the cursor's location. Suitable fixed locations could be the areas outside the circle, but this approach would force users to look back-and-forth frequently, and this was something we wanted to minimize to lessen simulator sickness. The central area could also be problematic because it might lead to erroneous selections because users would need to rotate their head to cross to other letter regions. Other possible solutions were to use dwell, or to use an additional input device; however, both approaches would go against our design criteria. Moreover, a fixed location within the center area would still require users to move their head or eyes every time they would enter a letter region and want to see whether the word(s) shown were the ones they would need.

Instead of placing the recommended words in a fixed position, a dynamic solution was chosen. Dynamic locations could be based on the current location of the cursor. However, this would also require dwelling or an additional input device for selection. In the end, we decided that the two recommend words could appear just outside of the current letter region and, by implication, next to the location of the cursor (see Fig. 7). This dynamic solution would minimize not only eye movements to check the words, but also head movements to select a word because of their proximity to the cursor and users' focal viewpoint. In one way, this represented an extension of our selection technique for letters but applied to select words without the need of dwelling time and an extra device.

Using this approach, the spelling correction would only work when the user entered a letter region. The words would disappear when the user went back into the center area. Similar to selecting a letter (by moving the cursor to the letter region) the user would move the cursor into the word region once. After each selection, the user must go back to the center area. The logic behind this was that after selecting a word, the user would need to go to another letter region.

To further encourage users to select recommended words and improve text entry speed, a space character was automatically added to the end of a word after its selection. This design rationale followed Fitts' law [6]. The completion time was analyzed based on Fitts' law and the formula proposed by Mackenzie [43]

$$MT = a + b\log_2\left(\frac{A}{W} + 1\right) \tag{1}$$

where MT was the average time to complete the movement; a and b were model parameters; A was the distance from movement origin to the target center; and W was the width of the target.

In our case, the distance A from the current letter, to the word selection region would always be smaller than the distance to reach the "space bar". For W, we designed the candidate region to have a broader width than the "space bar" (Fig. 7), so the completion time to get a space between words from the candidate region, in our layout, would always be smaller than the time to get it from the "space bar" (except from "A" or "<-"). In this way, there was no need for users to hit the space letter region.

6 Study Two

The goal of Study Two was to compare five possible hands-free techniques, which were Dwell Circular (DC), Dwell-Free Circular (DFC), Swype Circular (SC), Dwell QWERTY (DQ) and Swype QWERTY (SQ). DFC was our technique that had been optimized based on features described earlier. Fig. 8 shows examples of using SQ, SC, and DFC to enter the words "hello world". The techniques are described briefly in the next section.

6.1 Design of the Testing Techniques

For each layout type, we kept the graphical aspects the same; the only difference was how letters could be selected. Between the circular and QWERTY layouts, we also kept all other parameters the same—e.g. the distance between the user and the keyboard. One difference between them was that the QWERTY layouts had 4 candidate words where circular layouts only had 2. The reason for QWERTY layouts to have 4 candidate words was because previous research using the QWERTY layout had used 4 words instead of 2.

For SQ, we adopted the method used in [20] for indicating the select action. An example of typing the word 'world' is shown in Fig. 8a. At the beginning, the user moves the cursor to the target, then a button representing an action appears above the target after a wait time of 400 ms (i.e. the start of a Swype path); after the button appears, the user moves the cursor to the button followed by moving the cursor back to the target to perform the selection. When the user

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2019.2898736, IEEE Transactions on Visualization and Computer Graphics

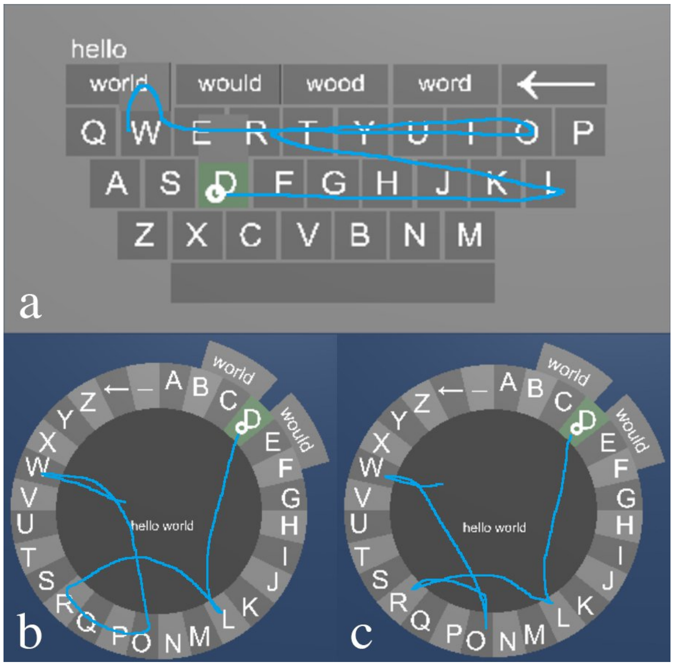

Fig. 8. (a; top) An example of typing the word 'world' in Swype QWERTY; the interface of the Dwell QWERTY was same but without popup buttons. (b; lower left) An example of typing a 'world' in Swype Circular. (c; lower right) An example of typing the 'world' in Dwell-Free Circular; the interface for Dwell Circular is the same except that users have to wait for 400 ms to select the letter from the letter regions.

finishes the Swype action, the system provides four recommended words in the candidate regions (See Fig. 8a, 'world' is the best-recommended word, 'word' is the fourth best-recommended word). The best match is automatically selected if the user starts Swyping the next vocable (e.g. 'world' in Fig. 8a), however, if the match is not the best the user must select it directly, following the same procedure as selecting a single letter. During the Swype action, only letters are active and selectable, other special characters (e.g. space/delete) are not.

As stated earlier, the design of the dwell-free technique (DFC) was based on the features derived from the first study and described in the previous section. An example of how to use DFC can be found in Fig. 8c. Of the three circular techniques, SC had a different selection feature; it allowed users to select the next letter (that was different from the last selected letter) without the need of returning to the inner circle—i.e. they could Swype to the next letter. An example of how to use SC is presented in Fig. 8b.

For two dwell techniques (DC and DQ), we set 400 ms for one letter input and dwell for another 400 ms to make the double input. We adopted 400 ms because any smaller dwell time would be error-prone and larger dwell time would cause a low text input rate. This was consistent with the implementations of dwell techniques in prior research (e.g. [27]).

Backspace deleted the last input, be that a complete word or a single letter. For all techniques, the system would append automatically a space if the word was selected from the candidate regions. Swype-based methods and the spelling correction used the Damerau–Levenshtein distance algorithm for word suggestions. The same dictionary [42] was used among all techniques. SC and SQ applied the Swype algorithm, other three techniques used the Symspell spell-correction algorithm as mentioned in the previous section where we set the algorithm with the max search distance of 2 to enhance the accuracy.

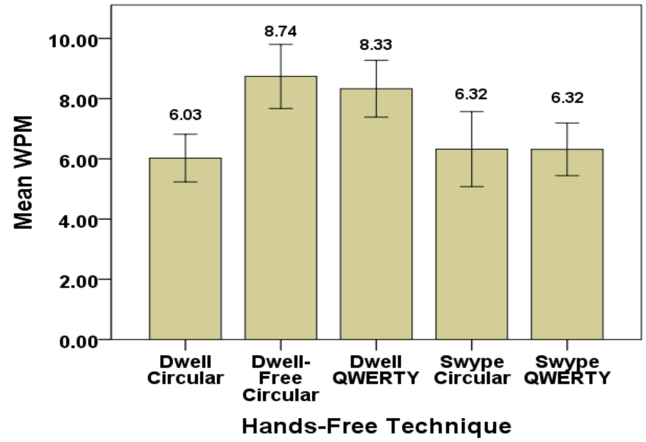

Fig. 9. Mean text entry speed across the 5 hands-free techniques. Error bars indicate ± 2 standard errors. The Dwell-Free Circular technique led to the fastest speed with 8.74 WPM on average.

6.2 Hypotheses

We had two hypotheses for this study. Our first hypothesis (*H1*) was that DFC should be the fastest technique. Our second hypothesis (*H2*) was that DFC should have the lowest error rate and the error rate should be significantly lower than other techniques.

6.3 Participants and Apparatus

15 participants (10 males and 5 females; aged between 18 to 26; M = 21.4, SD = 2.03) were recruited from the same university campus as in the Study One. None of the participants participated in Study One. Their alphabet familiarity was the same as in Study One since they were the same demographic. All participants had normal or corrected-to-normal vision and reported that they were familiar with the QWERTY keyboard (M = 4.1, from 1 – No Skill to 5 – Expert). Only one participant had no experience with HMD before. This experiment used the same apparatus as Study One.

6.4 Procedure and Design

The study followed a within-subjects design with one independent variable: Technique (DC, DFC, SC, DQ, and SQ). The order of the five hands-free techniques was counterbalanced. For each technique, participants needed to enter 8 phrases, which were randomly sampled from the MacKenzie's phrase set [37] with no repeated phrases within the same session. Each phrase was displayed at the center of the inner circle for the circular layouts and above the candidate regions for the QWERTY layouts—this was consistent with practices from previous studies. Participants were instructed to type as quickly and accurately as possible. Between sessions, they were encouraged to take breaks if they felt tired. The study lasted around fifty minutes. In total, we collected 15 participants × 5 hands-free techniques × 8 phrases = 600 phrases.

6.5 Results

We employed a one-way repeated measure ANOVA and Bonferroni corrections for pair-wise comparisons. We also used a Greenhouse-Geisser adjustment to correct for violations of the sphericity assumption. We indicate effect sizes whenever feasible (η_p^2).

6.5.1 Text Entry Speed

WPM ranged between 6.03 (s.e. = 0.40) for DC and 8.74 (s.e. = 0.53) for DFC (Fig. 9). ANOVA yielded a significant effect of Technique ($F_{1.507,21.091} = 12.746$, $p < .001$, $\eta_p^2 = .477$, observed power = .975). The pairwise comparisons showed significant differences between DC and DFC, DC and DQ, DFC and DQ, DFC and SC, DFC and SQ, DQ and SQ (all $p < .05$).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2019.2898736, IEEE Transactions on Visualization and Computer Graphics

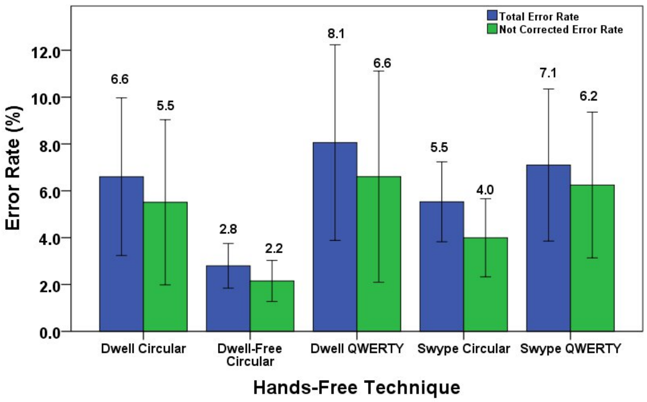

Fig. 10. Mean TER and NCER across 5 hands-free techniques. Error bars indicate ± 2 standard errors. The Dwell-Free Circular technique led to the lowest TER (2.8%) and NCER (2.2%).

6.5.2 Error Rate

Figure 10 shows the TER and NCER for the five hands-free techniques. Although the difference between each technique seemed large, from the ANOVA test we only found a trend toward a significant effect of the techniques on TER ($F_{2.313,32.376} = 2.652$, $p = .079$, $\eta_p^2 = .159$, observed power = .525). In addition, there was no significant effect of Technique on NCER ($F_{2.282,31.952} = 2.315$, $p = .109$, $\eta_p^2 = .142$, observed power = .464).

6.6 Discussion

Our results support *H1* (DFC has outperformed all the other techniques in text entry rate). On the one hand, *H2* is not supported where the difference in TER and NCER between DFC and other techniques is not significant (although the trend seems to be towards significance for TER). On the other hand, DFC has led to the lowest TER and NCER.

Considering that all features, except for the selection mechanism, have been kept the same in the three circular layouts, our findings suggest that the go-and-hit selection seems to be a better approach for a circular layout and that can work well with head-based motions. Surprisingly, the performance of SC is much lower than DFC, even though it can make selections which do not require users to move the cursor back to the inner circle. The reason may be because in DFC users only need to consider whether the candidate regions have the target word and, if they do not, they can directly go back to the inner circle to do the reset and move to the next letter. In SC, on the other hand, users not only need to consider the candidate regions, but they also need to consider whether they should go back to the inner circle or go through the outer circle to select the next letter—this cognitive process would have added extra burden and time for users to make the decision. A closer analysis of the typing process shows cases that users accidentally have typed some letters unrelated to the target word; this might have been caused by the wrong selection during the Swype action as users accidentally move back to the inner circle to select the wrong letter when they had decided to go through the outer circle.

The text entry rate of DQ is in line with the DQ technique tested in [3]. For DQ, some users have commented that 0.4s is very (almost too) short and has made them frustrated and uncomfortable—they have felt that something is pushing them to move to the next letter very quickly in order to avoid unintentional selections—i.e. they have found it not very usable. In contrast, in a non-dwell technique like our DFC RingText, users have felt relaxed, and this might have been the reason that users have been able to achieve a significantly higher text entry rate and close significantly lower TER (but at the same time still feeling comfortable).

7 Study Three

Given that our dwell-free technique outperformed other 4 baseline techniques, we wanted to explore its performance if users could receive some more training for two groups, novices and experts. For the potential expert group, we ordered the participants from Study Two based on their average text entry speed, and invited those participants who achieved a relatively high text entry performance to continue for a 4-day study. For the novice group, we recruited participants who were not involved in either study 1 or 2. The design of the third study followed a similar approach reported in previous works [8], [30].

This third study was to last for four days with two daily sessions for each participant. The goal was to measure how well novice and expert users could improve their text entry speed and standard typing metrics [38] through practice over time.

7.1 Participants and Apparatus

10 participants (9 males and 1 female; aged from 19 to 28, M = 21.6, SD = 3.17) were recruited from the same university campus as the previous two experiments; 5 of them who achieved a relatively high text entry speed in Study 2 agreed to join this 4-day study. They formed the potential 'expert' group. The 5 participants who were not involved in experiments 1 and 2 formed the 'novice' group. These participants had similar visual acuity and alphabetical knowledge as the ones from the previous studies since they represented the same demographic. They reported an average 4 for experience with the QWERTY keyboard on a scale from 1 ('No Skill') to 5 ('Expert'). All participants had some previous experience with HMD before. This experiment used the same apparatus as the previous studies.

7.2 Procedure and Design

The study consisted of a series of sessions over 4 consecutive days, with two sessions per day. In each session, participants needed to complete 8 phrases, which were randomly sampled from the MacKenzie's phrase set [37] with no repeated phrases within the same session. Each phrase was displayed at the center of the inner circle. All eight sessions lasted approximately an hour. In total, we collected 640 phrases (10 participants × 8 sessions × 8 phrases).

7.3 Results

We employed a mix-design ANOVA with Sessions (from one to eight) as the within-subject variable and Group (novice and potential expert) as the between-subjects variable. Bonferroni correction was used for pair-wise comparisons and Greenhouse-Geisser adjustment was used for degrees of freedom if there were violations to sphericity in the data. We indicate effect sizes whenever feasible (η_p^2).

7.3.1 Text Entry Speed

ANOVA tests yielded a significant effect of Session ($F_{2.592,20.733} = 31.344$, $p < .001$, $\eta_p^2 = .797$, observed power = 1.000) and a close to significant effect of Session × Group ($F_{2.592,20.733} = 31.344$, $p = .058$, $\eta_p^2 = .276$, observed power = .591) on text entry speed. There was a significant effect of Group ($F_{1,8} = 8.127$, $p < .05$, $\eta_p^2 = .504$, observed power = .705) on text entry speed. This suggests that although participants in the two groups had a significant difference in text entry speed, their learning over time was somewhat similar.

Post-hoc pair-wise comparisons revealed significant differences between session 1-4, 1-5, 1-6, 1-7, 1-8, 2-4, 2-5, 2-6, 2-8, 3-8, 4-8, 5-8, 6-8 and 7-8 (all $p < .05$).

Overall, the average speed across all sessions was 10.45 WPM; s.e. = 0.28. In particular, the novice group achieved 8.9 WPM (s.e. = 0.30), while the potential expert group achieved 11.99 WPM (s.e. = 0.34). Figure 11 shows the mean WPM by sessions for each participant and the two groups. The average speed for the first session was 8.50 WPM (s.e. = 0.76); it bumped up to 12.27 WPM (s.e. = 0.62) in the last session, with an increase of 44.4%.

In the last session, the potential expert group improved their performance to 13.24 WPM (s.e. = 0.80) from the first session of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2019.2898736, IEEE Transactions on Visualization and Computer Graphics

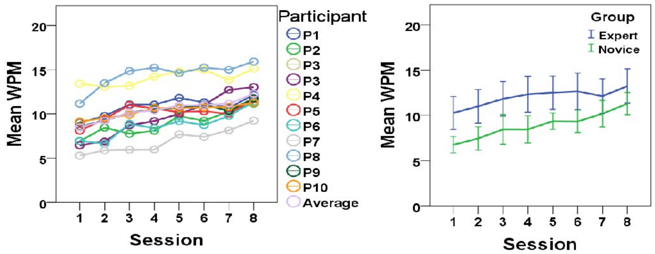

Fig. 11. Mean WPM using RingText over 8 sessions for each participant (left) and the mean WPM for each group (right). Error bars indicate ± 2 standard errors. The graphs show an upward trend for all participants. They also show that participants have not yet reached the peak.

10.26 WPM (s.e. = 0.72); the novice group improved to 11.30 WPM (s.e. = 0.80) from the first session of 6.75 WPM (s.e. = 0.72).

7.3.2 Error Rate

For TER, ANOVA tests yielded no significant effect of session ($F_{7,56}$ = 1.462, p = .200, η_p^2 = .154, observed power = .563), Group ($F_{1,8}$ = .109, p = .749, η_p^2 = .013, observed power = .060), or Session × Group ($F_{7,56}$ = .452, p = .864, η_p^2 = .054, observed power = .182). For NCER, ANOVA tests also yielded no significant effect of session ($F_{7,56}$ = .574, p = .774, η_p^2 = .067, observed power = .226), Group ($F_{1,8}$ = .157, p = .702, η_p^2 = .019, observed power = .064), or Session × Group ($F_{7,56}$ = .913, p = .504, η_p^2 = .102, observed power = .356).

Figure 12 shows the mean TER and NCER over eight sessions. Overall, the average TER and NCER across all sessions were 3.10% (s.e. = 0.25%) and 2.25% (s.e. = 0.14%) respectively. In particular, the average TER and NCER for the potential expert group were 2.90% (s.e. = 0.22%) and 2.44% (s.e. = 0.25%), whereas for the novice group they were 3.29% (s.e. = 0.34%) and 2.05% (s.e. = 0.22%).

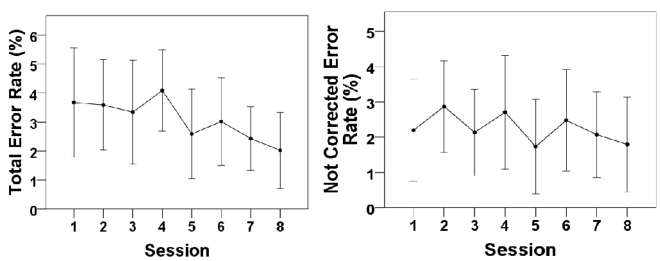

Fig. 12. Mean TER (left) Mean NCER (right) over 8 sessions. All Error bars indicate ± 2 standard errors.

7.3.3 Spelling Correction Statistics

The total words participants were supposed to type in the experiment were 3261 (excluding the words with length fewer than two letters). Of these words, 2822 (again excluding the words with length fewer than two letters) were selected from candidate regions, including 986 words predicted in advanced and 1836 words corrected in the last letter. For those 2822 words suggested by the spelling correction algorithm, there were 2341 correct selections and 185 wrong selections.

7.4 Is RingText applicable to AR/MR?

We conducted a small, follow-up experiment at the end of the eighth session to test whether RingText would be applicable to AR/MR devices and could lead to similar performance to the VR version. We asked participants to try our technique on Meta 2 AR goggles. Five participants agreed to do the experiment. Thus, we collected 5 participants × 8 phrases = 40 phrases.

The results from these five participants pointed to a positive experience. They were able to achieve an average text entry speed of 12.06 WPM with a low level of TER and NCER (1.82% and 1.44% respectively) on the Meta 2 HMD. This performance was very similar to the results in the last session using the Gear VR device (12.24 WPM, 1.42% TER, and 1.13% NCER).

Based on these results, we can infer that our technique has the potential to lead to comparable results not only in AR but also likely in MR HMDs as well; thus, it is very likely that RingText can be easily adapted to AR/MR devices.

7.5 DISCUSSION, LIMITATIONS, AND FUTURE WORK

Text Entry Speed and Error Rate. The average speed of RingText across sessions for novice and expert users are 8.9 WPM and 11.99 WPM. Novice users can type 11.30 WPM after 1 hour of practice where expert users can reach 13.24 WPM after 1.5 hours of training (including the time they spent in Study 2 with circular layouts). This result indicates that RingText outperforms some other dwell-free techniques such as EyeK [22], Eye-S [19], and EyeWrite [17] with 6.03 WPM, 6.87 WPM, 7.99 WPM, respectively. The text entry rate after training is comparable to the speech input (6-13 WPM) [44], [45], and leads to better performance than the head-based dwell method in [3] (10.59 WPM). In terms of word-level TER and NCER, RingText achieved a 3.10% and 2.25% across sessions, which are comparable with the head-based dwell techniques for HMDs reported in [3] (3.79% and 2.46%).

As mentioned before, all our participants are not native alphabet users. It can be argued that given their familiarity with the alphabet, native users could lead to higher text entry speeds than non-native users, similar to the result reported in [8]—this latter group are almost identical to our participants (they are university students within the same age range and whose language of instruction is English but are not native alphabet users). However, future work is needed to confirm whether native users could achieve a significantly better result than non-native alphabet users with RingText.

Overall, significant learning effects were observed in text entry speed, indicating the possibility of even higher text entry speeds with further practice—as Figure 11 shows an increasing trend for text entry speeds even in the final session and participants' performance has not peaked yet.

Design of dynamic, non-fixed candidate regions. This work makes the first attempt to combine the circular layout with dynamic candidate regions that are placed just next to the region of the last selected letter. The percentage of the candidate word selections shows that our candidate regions are used very frequently (86.5% of the words have been chosen from the candidate regions). There are three main reasons that explain why our design has led to such high frequent use.

(1) *Minimal checking time*. The time for users to check whether a candidate region had the correct suggested word is reduced as these regions are close to the current letter region which would likely be where the users would be paying to attention to at the moment.
(2) *Reduced travel distance*. Unlike the design in [5], users only need to travel a short distance to hit the region to select a word because the cursor is just next to the candidate regions.
(3) *Space automatically appended*. Users have commented that they have automatically thought of the candidate regions as an easy way to get the space character. Our observations show that even though in cases when all letters of a word are already entered corrected, participants would move the cursor to the candidate region to select because its distance is often shorter the distance to the letter region of the space character.

An additional option for the hands-free and controller-free scenario. Considering the design guidelines in [36], we recommend the RingText as an additional option for hands-free and controller-free scenarios, since the text entry rate is significantly better than the head pointing dwell techniques and comparable to the speech input [44], [45] but with no significant drawbacks in recognition problems and no privacy problems for users when typing in public places. There are several scenarios that people can use RingText; for example, when users receive a message while watching a movie in

VR or when they want to send a quick chat text in a VR multiplayer game, they can simply popup RingText and quickly type the message.

Limitations and Future Work. The present research has several limitations, which can also serve a possible direction for future work.

RingText is based on head-pointing so that it might be inappropriate for people who cannot rotate their head—e.g. users with a neck injury. Moreover, we have evaluated in a lab which shows that users have no issues using it in a non-public environment. We have not looked at issues of social acceptability when users want to use it in public places.

It would have been good to use a standardized interface usability survey (like the System Usability Scale) in our first two studies so that we can compare across techniques. This is something that could be done in future studies dealing with new keyboard designs.

RingText shared one limitation with other keyboard design where the default keyboard letters are in lowercase where uppercase letters, symbols, and emoji are required. Future research could explore how RingText would scale up to support uppercase characters and symbols. One possible solution is to use the forward head movement to switch between sub-layouts with different types of characters and symbols. We have tried measuring forward and backward head movements, and current mobile devices can detect these types of motions. It is possible to set a forward acceleration threshold which can be used as an indicator for when users want to switch layouts. Future research is needed to determine how this approach will work.

We have not investigated the optimal size of the trigger area for RingText. Smaller trigger areas of the letter regions can lead to a lower error rate, but it might also result in a lower text entry rate since users may miss the trigger area of the intended letter and must re-enter it to make the selection. Future work is needed to investigate the optimal size(s) of the trigger area to let users select letters quickly without incurring many mistakes. Additionally, we can apply a static decoding method [46] to handle the noise of the input further. This is similar to a method to mitigate the "fat finger" problem in smartphones [47] where users with large fingers may mistakenly select unintended buttons. In our case, it may be possible to use this model to help us understand which letters the user is aiming to type.

As stated earlier, participants in Study Three did not reach peak performance after 8 sessions. In similar experiments reported in [3], [8], [30], their participants had 5-6 sessions and could not reach it either. We designed the experiment with 8 sessions assuming that 2-3 extra sessions would have allowed participants to reach a stable text entry rate. It may be of interest to explore if there is a common minimum period of training time that participants need to reach maximum performance with RingText and similar techniques.

Finally, the dwell-time for Dwell technique and the algorithm for Swype technique tested are based on their common implementation. In the future, it may be useful to compare RingText with other variations of these techniques that use some optimized features.

Despite these limitations, our results show the potential use of circular layouts in head-based dwell- and hands-free text entry in mobile VR HMDs.

8 Conclusion

We have provided the first example of a formal evaluation of ring-based text input for mobile virtual reality (VR) head-mounted displays (HMDs) that is both dwell-free and hands-free. Our example technique, RingText, allows users to enter text by making small motions with their head and select letters from a circular keyboard layout with two concentric circles: the outer circle contains letters housed in distinct regions, while the inner circle serves to reset selection and allows users to search for the next letter.

In our first study, we determine the suitable size of the inner circle, the number of letters per region (LPR) in the areas of the outer circle, and alphabet starting position. The results show that 1 LPR leads to a significantly better performance in entry text speed; a larger center area can potentially decrease error rates, and users preferred the alphabet to start from the top. Based on the results, an optimized layout that shows two recommended words placed dynamically next to the cursor is adopted to develop RingText. Then, a first comparative study of hands-free text entry techniques in VR has been conducted by comparing the RingText with four other text entry mechanisms. Results show that RingText is the most efficient technique; it has led users to achieve a significantly higher text entry rate and close to a significantly lower total error rate. To further explore its performance, a third study is undertaken with 10 participants doing two daily sessions for 4 consecutive days. The results of this last study show that after 8 practice sessions even novice users can achieve an average text entry speed of 11.30 WPM while expert users can achieve 13.24 WPM in the last session. Because performance over these sessions shows an increasing trend, we believe that there is some place for improvement in their text entry speed with further practice sessions.

All in all, RingText is an efficient technique for text entry in mobile VR/VR head-mounted displays that do not require users to hold any additional input devices. We hope this work can inform future work on dwell-free and hands-free text entry techniques based on a circular layout for VR/AR/MR HMDs.

Acknowledgments

The authors would like to thank the anonymous reviewers for their valuable comments and helpful suggestions. The work is supported in part by Xi'an Jiaotong-Liverpool University (XJTLU) Key Special Fund (#KSF-A-03) and Research Development Fund.

References

[1] J. Y. Oh, J. Lee, J. H. Lee, and J. H. Park, "AnywhereTouch: Finger Tracking Method on Arbitrary Surface Using Nailed-Mounted IMU for Mobile HMD," in *International Conference on Human-Computer Interaction*, 2017, pp. 185–191.

[2] J. Gugenheimer, "Nomadic Virtual Reality: Exploring New Interaction Concepts for Mobile Virtual Reality Head-Mounted Displays," in *Proceedings of the 29th Annual Symposium on User Interface Software and Technology - UIST '16 Adjunct*, Tokyo, Japan, 2016, pp. 9–12.

[3] C. Yu, Y. Gu, Z. Yang, X. Yi, H. Luo, and Y. Shi, "Tap, Dwell or Gesture?: Exploring Head-Based Text Entry Techniques for HMDs," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems - CHI '17*, Denver, Colorado, USA, 2017, pp. 4479–4488.

[4] P. Majaranta, U.-K. Ahola, and O. Špakov, "Fast gaze typing with an adjustable dwell time," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2009, pp. 357–360.

[5] K. Katsuragawa, J. R. Wallace, and E. Lank, "Gestural Text Input Using a Smartwatch," in *Proceedings of the International Working Conference on Advanced Visual Interfaces - AVI '16*, Bari, Italy, 2016, pp. 220–223.

[6] P. M. Fitts, "The information capacity of the human motor system in controlling the amplitude of movement.," *J. Exp. Psychol. Gen.*, vol. 121, no. 3, p. 262, 1992.

[7] J. Grubert, L. Witzani, E. Ofek, M. Pahud, M. Kranz, and P. O. Kristensson, "Text Entry in Immersive Head-Mounted Display-Based Virtual Reality Using Standard Keyboards," in *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, 2018, pp. 159–166.

[8] D. Yu, K. Fan, H. Zhang, D. Monteiro, W. Xu, and H. Liang, "PizzaText: Text Entry for Virtual Reality Systems Using Dual Thumbsticks," *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 11, pp. 2927–2935, Nov. 2018.

[9] D. A. Bowman, C. J. Rhoton, and M. S. Pinho, "Text input techniques for immersive virtual environments: An empirical comparison," in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 2002, vol. 46, pp. 2154–2158.

[10] S. Pick, A. S. Puika, and T. W. Kuhlen, "SWIFTER: Design and evaluation of a speech-based text input metaphor for immersive virtual environments," in *3D User Interfaces (3DUI), 2016 IEEE Symposium on*, 2016, pp. 109–112.

[11] J. Gugenheimer, D. Dobbelstein, C. Winkler, G. Haas, and E. Rukzio, "FaceTouch: Enabling Touch Interaction in Display Fixed UIs for Mobile Virtual Reality," in *Proceedings of the 29th Annual Symposium on User Interface Software and Technology - UIST '16*, Tokyo, Japan, 2016, pp. 49–60.

[12] Y. R. Kim and G. J. Kim, "HoVR-Type: Smartphone as a typing interface in VR using hovering," in *Consumer Electronics (ICCE), 2017 IEEE International Conference on*, 2017, pp. 200–203.

[13] Y. Lu, C. Yu, X. Yi, Y. Shi, and S. Zhao, "BlindType: Eyes-Free Text Entry on Handheld Touchpad by Leveraging Thumb's Muscle Memory," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 1, no. 2, pp. 1–24, Jun. 2017.

[14] X. Yi, C. Yu, M. Zhang, S. Gao, K. Sun, and Y. Shi, "ATK: Enabling Ten-Finger Freehand Typing in Air Based on 3D Hand Tracking Data," in *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology - UIST '15*, Daegu, Kyungpook, Republic of Korea, 2015, pp. 539–548.

[15] P. Majaranta and K.-J. Räihä, "Twenty years of eye typing: systems and design issues," in *Proceedings of the 2002 symposium on Eye tracking research & applications*, 2002, pp. 15–22.

[16] P. O. Kristensson and K. Vertanen, "The potential of dwell-free eye-typing for fast assistive gaze communication," in *Proceedings of the symposium on eye tracking research and applications*, 2012, pp. 241–244.

[17] J. O. Wobbrock, J. Rubinstein, M. Sawyer, and A. T. Duchowski, "Not Typing but Writing: Eye-based Text Entry Using Letter-like Gestures," in *Proceedings of the Conference on Communications by Gaze Interaction (COGAIN)*, Leicester, 2007, pp. 61–64.

[18] J. O. Wobbrock, J. Rubinstein, M. W. Sawyer, and A. T. Duchowski, "Longitudinal evaluation of discrete consecutive gaze gestures for text entry," in *Proceedings of the 2008 symposium on Eye tracking research & applications - ETRA '08*, Savannah, Georgia, 2008, p. 11.

[19] M. Porta and M. Turina, "Eye-S: a full-screen input modality for pure eye-based communication," in *Proceedings of the 2008 symposium on Eye tracking research & applications*, 2008, pp. 27–34.

[20] A. Kurauchi, W. Feng, A. Joshi, C. Morimoto, and M. Betke, "EyeSwipe: Dwell-free Text Entry Using Gaze Paths," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16*, Santa Clara, California, USA, 2016, pp. 1952–1956.

[21] M. Porta, "A study on text entry methods based on eye gestures," *J. Assist. Technol.*, vol. 9, no. 1, pp. 48–67, Mar. 2015.

[22] S. Sarcar, P. Panwar, and T. Chakraborty, "EyeK: an efficient dwell-free eye gaze-based text entry system," in *Proceedings of the 11th Asia Pacific Conference on Computer Human Interaction - APCHI '13*, Bangalore, India, 2013, pp. 215–220.

[23] D. Pedrosa, M. D. G. Pimentel, A. Wright, and K. N. Truong, "Filteryedping: Design Challenges and User Performance of Dwell-Free Eye Typing," *ACM Trans. Access. Comput.*, vol. 6, no. 1, pp. 1–37, Mar. 2015.

[24] J. P. Hansen, K. Tørning, A. S. Johansen, K. Itoh, and H. Aoki, "Gaze typing compared with input by head and hand," in *Proceedings of the 2004 symposium on Eye tracking research & applications*, 2004, pp. 131–138.

[25] 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*, 1998, pp. 213–214.

[26] M. Proschowsky, N. Schultz, and N. E. Jacobsen, "An intuitive text input method for touch wheels," in *Proceedings of the SIGCHI conference on Human Factors in computing systems*, 2006, pp. 467–470.

[27] A. Huckauf and M. H. Urbina, "Gazing with pEYEs: towards a universal input for various applications," in *Proceedings of the 2008 symposium on Eye tracking research & applications - ETRA '08*, Savannah, Georgia, 2008, p. 51.

[28] C. Topal, B. Benligiray, and C. Akinlar, "SliceType: Fast Gaze Typing with a Merging Keyboard," *ArXiv Prepr. ArXiv170602499*, 2017.

[29] G. Shoemaker, L. Findlater, J. Q. Dawson, and K. S. Booth, "Mid-air text input techniques for very large wall displays," in *Proceedings of Graphics interface 2009*, 2009, pp. 231–238.

[30] J. Gong *et al.*, "WrisText: One-handed Text Entry on Smartwatch using Wrist Gestures," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*, Montreal QC, Canada, 2018, pp. 1–14.

[31] X. Yi, C. Yu, W. Xu, X. Bi, and Y. Shi, "COMPASS: Rotational Keyboard on Non-Touch Smartwatches," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems - CHI '17*, Denver, Colorado, USA, 2017, pp. 705–715.

[32] G. Kurtenbach and W. Buxton, "The limits of expert performance using hierarchic marking menus," in *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '93*, Amsterdam, The Netherlands, 1993, pp. 482–487.

[33] M. Kytö, B. Ens, T. Piumsomboon, G. A. Lee, and M. Billinghurst, "Pinpointing: Precise Head- and Eye-Based Target Selection for Augmented Reality," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*, Montreal QC, Canada, 2018, pp. 1–14.

[34] X. "Anthony" Chen, T. Grossman, and G. Fitzmaurice, "Swipeboard: a text entry technique for ultra-small interfaces that supports novice to expert transitions," in *Proceedings of the 27th annual ACM symposium on User interface software and technology - UIST '14*, Honolulu, Hawaii, USA, 2014, pp. 615–620.

[35] W. Feng, M. Chen, and M. Betke, "Target reverse crossing: a selection method for camera-based mouse-replacement systems," in *Proceedings of the 7th International Conference on PErvasive Technologies Related to Assistive Environments*, 2014, p. 39.

[36] M. Speicher, A. M. Feit, P. Ziegler, and A. Krüger, "Selection-based Text Entry in Virtual Reality," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*, Montreal QC, Canada, 2018, pp. 1–13.

[37] 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*, 2003, pp. 754–755.

[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*, 2003, pp. 113–120.

[39] S. G. Hart, "NASA Task load Index (TLX). Volume 1.0; Paper and pencil package," 1986.

[40] R. S. Kennedy, N. E. Lane, K. S. Berbaum, and M. G. Lilienthal, "Simulator Sickness Questionnaire: An Enhanced Method for Quantifying Simulator Sickness," *Int. J. Aviat. Psychol.*, vol. 3, no. 3, pp. 203–220, Jul. 1993.

[41] W. Garbe, *SymSpell: 1 million times faster through Symmetric Delete spelling correction algorithm*. 2018.

[42] "the most frequent 10 000 words of English and Example Sentences - The list of most commonly used words." [Online]. Available: http://www.use-in-a-sentence.com/english-words/10000-words/the-most-frequent-10000-words-of-english.html. [Accessed: 01-Apr-2018].

[43] I. S. MacKenzie, "Fitts' law as a research and design tool in human-computer interaction," *Hum.-Comput. Interact.*, vol. 7, no. 1, pp. 91–139, 1992.

[44] G. González, J. P. Molina, A. S. García, D. Martínez, and P. González, "Evaluation of Text Input Techniques in Immersive Virtual Environments," in *New Trends on Human-Computer Interaction*, J. A. Macías, A. Granollers Saltiveri, and P. M. Latorre, Eds. London: Springer London, 2009, pp. 1–10.

[45] L. Hoste, B. Dumas, and B. Signer, "SpeeG: a multimodal speech- and gesture-based text input solution," in *Proceedings of the International Working Conference on Advanced Visual Interfaces - AVI '12*, Capri Island, Italy, 2012, p. 156.

[46] J. Goodman, G. Venolia, K. Steury, and C. Parker, "Language modeling for soft keyboards," in *Proceedings of the 7th international conference on Intelligent user interfaces*, 2002, pp. 194–195.

[47] D. Vogel and P. Baudisch, "Shift: a technique for operating pen-based interfaces using touch," in *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '07*, San Jose, California, USA, 2007, p. 657.