

Typing on Midair Virtual Keyboards: Exploring Visual Designs and Interaction Styles

Jiban Adhikary^[0000-0002-5471-5090] and Keith Vertanen^[0000-0002-7814-2450]

Michigan Technological University, Houghton, Michigan, USA
{jiban,vertanen}@mtu.edu

Abstract. We investigate typing on a QWERTY keyboard rendered in virtual reality. Our system tracks users' hands in the virtual environment via a Leap Motion mounted on the front of a head mounted display. This allows typing on an auto-correcting midair keyboard without the need for auxiliary input devices such as gloves or handheld controllers. It supports input via the index fingers of one or both hands. We compare two keyboard designs: a normal QWERTY layout and a split layout. We found users typed at around 16 words-per-minute using one or both index fingers on the normal layout, and about 15 words-per-minute using both index fingers on the split layout. Users had a corrected error rate below 2% in all cases. To explore midair typing with limited or no visual feedback, we had users type on an invisible keyboard. Users typed on this keyboard at 11 words-per-minute at an error rate of 3.3% despite the keyboard providing almost no visual feedback.

Keywords: Text Entry, Virtual Reality, Head Mounted Display (HMD)

1 Introduction

Virtual reality (VR) and augmented reality (AR) head-mounted displays (HMDs) offer the promise of rich, three-dimensional, interactive user experiences. As technology advances, HMDs will become more capable, portable, and fashionable. As this starts to occur, we believe users will shift to VR and AR devices to perform many of the activities currently supported by smartphones or other portable touchscreen devices. One of the core interaction primitives on touchscreen devices is text input. Similar to the situation with mobile devices, users will likely not want to carry an additional input device to support text input (e.g. a wireless keyboard). Further, using auxiliary devices can be troublesome in VR and AR scenarios where a user may be standing or moving around. Ideally, input would be possible relying solely on the HMD. While HMDs typically have a microphone, speech input can present social or privacy concerns. Moreover, correcting speech recognition errors can be time consuming, especially for text containing uncommon words [1]. HMDs increasingly feature a front-facing camera that can track the location and pose of a user's hands. Our work explores text input in VR using such hand tracking.

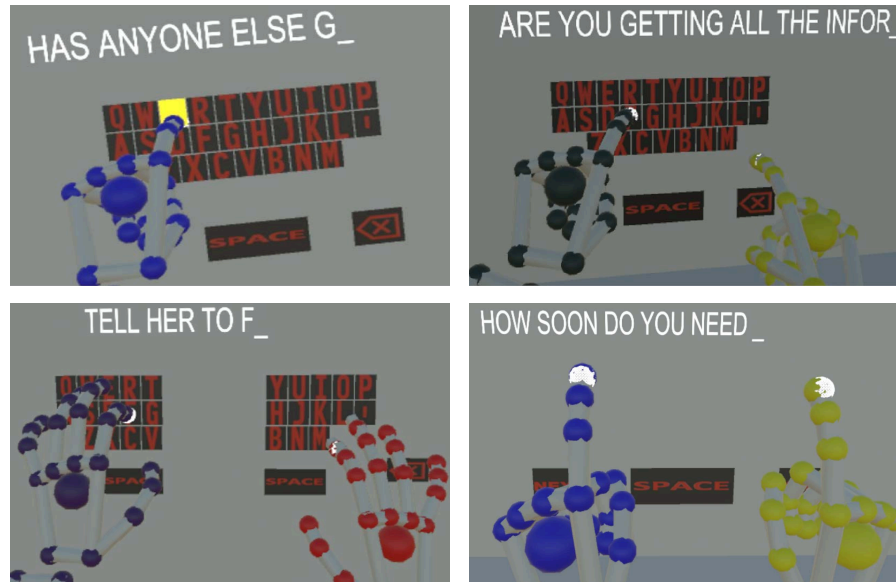


Fig. 1. Entering text using the normal keyboard with one hand (top left), the normal keyboard with two hands (top right), the split keyboard with two hands (bottom left), and the invisible keyboard (bottom right).

For new or infrequent users of text input in VR or AR, we think a familiar input method leveraging users' experience with auto-correcting touchscreen keyboards may be preferable. Thus we explore text entry by having users tap on a midair virtual keyboard. This keyboard is located directly in front of the user. This position allows an ergonomic front-facing head position, visual guidance of hands to the keys, and positions a user's hands in the tracker's field of view. Our work also provides a walk-up-and-use interface to enter text. While much of the previous work in midair text entry needs training of an hour or more [9,17,19,18,31,33], our system requires almost no training.

Our focus is on the visual design of the keyboard and the impact of typing with one or both hands. Our system tracks a user's hands via a Leap Motion depth camera mounted on a VR HMD. In the virtual environment, users see a rendering of their hands as well as a virtual keyboard (Figure 1). In a user study, we compare user performance and preference for typing with the index finger of one or both hands. We also compare typing with both hands on a normal layout versus a split layout. To our knowledge, we are the first to study bimanual typing on a midair auto-correcting virtual keyboard. Despite users seeing only a virtual version of their hands, and without tactile feedback or word predictions, users typed at 16 words-per-minute (wpm) at an error rate below 1% on the normal layout. We found the normal layout was superior to the split layout.

After our main study, participants completed an exploratory session in which they first defined a midair keyboard in a size and location of their own choosing.

Participants then typed sentences in midair with no visual keyboard reference. We found typing on an invisible midair keyboard can be surprisingly effective; users entered 71% of sentences on an invisible keyboard with zero errors. This suggests midair keyboard input may be possible even when visual feedback is limited, for example HMDs with a small display area for users with normal vision, or a normal display area for users who are low vision. In some cases, people may also need to input text when visual feedback is non-existent, for example audio-only AR or for users who are blind.

Our contributions in this work are as follows:

- (i) With little practice, novices typed on a midair keyboard with autocorrect and achieved acceptable walk-up performance; 16 words-per-minute at less than 1% character error rate. This is likely sufficient for applications requiring only modest amounts of text input. Our approach does not require user training or specialized input devices (as required by much existing work).
- (ii) We provide the first comparison of one- and two-handed midair keyboard performance. Unlike touchscreen keyboards, we did not find a performance advantage to typing with two fingers in a walk-up-and-use scenario.

2 Related Work

In this section, we review existing work related to text entry in virtual reality. For a detailed overview of the techniques by classification, strengths, limitations, and performance, see Dube and Arif [8].

AR keyboards. ARKB [16] exploited depth information obtained via a stereo camera attached to an HMD. A user's fingers were marked with colored markers. The stereo camera tracked the markers and detected collision with an augmented reality QWERTY keyboard. No user trial results were reported. PalmType [28] allowed text input for smartglasses using a QWERTY keyboard interface on a user's palm. In a user study, users wrote at 8 wpm on an optimized PalmType QWERTY keyboard with Vicon tracking system. Using a wrist-worn infrared sensor and a touchpad, users wrote at 5 wpm.

Our keyboard is similar to VISAR [9], an AR midair auto-correcting keyboard. VISAR uses the same VelociTap [26] decoder for auto-correction as we use here. VelociTap takes a series of keyboard touch locations and outputs the most likely text. Using a Microsoft HoloLens HMD, VISAR tracked a user's hand location and employed a fixed spatial offset to approximate the location of a user's index finger. On a virtualized midair input surface users wrote at 6 wpm. With the help of word predictions and two hours of practice, the entry rate improved to 18 wpm. Compared to VISAR, our system tracks users' fingertips as opposed to users' hands. We add new knowledge regarding bimanual typing performance and compare a normal QWERTY layout versus a split layout. We also explore input in VR rather than AR.

HoldBoard [2] used a smartwatch to enter text on smartglasses. Users selected a character on the smartwatch's screen using a combination of thumb position and index finger tapping. The result was displayed on the smartglasses. Users

entered text at 10 wpm after eight sessions. Yu et al. [32] also explored touch-based smartglass text input. Using a Google Glass HMD, users wrote at 9 wpm using one-dimensional touch input coupled with auto-correction.

Finger tracking using Leap Motion. ATK [30] allowed two-handed touch typing on a midair keyboard using a Leap Motion sensor to track a user's fingers. It provided visual feedback on a desktop display. The sensor was stationary and placed horizontally on a table under a user's hands. ATK inferred text based on 3D fingertip kinematics. After practice, users entered text at 29 wpm. ATK's reliance on ten-finger typing may make it difficult for non-touch-typists. Also ATK used a stationary tracker which could be challenging in standing or mobile use scenarios. While ATK is potentially a fast midair input method that enables 10 finger input, it differs from our work in that: 1) ATK displayed the keyboard on a monitor, 2) the Leap Motion was stationary, and 3) users did not interact with virtual hands in an immersive virtual environment.

Sridhar et al. [23] and Feit et al. [10] used a Leap Motion to track users' fingers. In both works, users entered text in midair by learning specific multi-finger gestures. In contrast to their approaches, we allowed input using just the index fingers. Our approach is intuitive, does not require learning any gestures, and users can easily transfer their experience typing on auto-correcting touchscreen keyboards.

Adhikary and Vertanen [1] investigated text input in VR using speech and a midair keyboard. They also used a Leap Motion sensor to track a user's fingers. A user could enter text with or without speech. When using speech input, a midair keyboard provided a fallback mechanism for correcting speech recognition errors. The midair keyboard supported word prediction. In a study with 18 participants, users entered phrases where half of the phrases contained an uncommon word. Users wrote at 28 wpm with speech versus 11 wpm without speech.

Auxiliary input devices. Various work has investigated VR input using auxiliary devices. Yu et al. [31] investigated gesture typing using an HMD and a gamepad controller. Head rotation was used to control a pointer on a virtual keyboard. The fastest entry rate was achieved using a word-gesture keyboard [34] at 25 wpm after an hour of practice. PizzaText [33] presented a circular keyboard layout in VR. Using the dual thumbsticks of a hand-held controller, novices wrote at 9 wpm while experts wrote at 16 wpm after two hours of practice.

McGill et al. [19] showed that injecting real-word video in VR significantly improved typing on a physical keyboard in VR. Walker et al. [27] presented a system that assists HMD users in typing on a visually occluded physical keyboard. With the help of auto-correction and visual feedback after each key press, users typed at 40 wpm.

Grubert et al. [12] showed that desktop and touchscreen keyboards can be used as text entry devices in VR. By simply rendering a user's fingers in VR, they showed users retain 60% of their typing speed on a desktop keyboard with no significant learning effects. In another study, Grubert et al. [11] studied hand representations in VR while typing using a standard physical keyboard. Users

wrote at 34 wpm with no hand visualization, 36 wpm with fingertip visualization, and 39 wpm with video see-through of their hands.

Knierim et al. [14] also investigated hand visualization in VR while typing on a physical keyboard. They compared user performance on hand visualization, semi-transparent hands, and no hands. Results revealed that expert typists benefited from seeing their hands whereas novice users benefited from semi-transparent hands. To aid typing with a midair VR keyboard, Gupta et al. [13] investigated the utility of tactile feedback. They compared audio-visual only feedback with three different on-finger and on-wrist vibrotactile feedback. While the speed and accuracy across different feedback conditions were comparable, users preferred the tactile feedback conditions.

Markussen et al. [17] analyzed one-handed text input in midair using three selection-based techniques. The location of a user's index finger was sensed via a tracked glove. A large high-resolution displayed a keyboard and a dot representing a user's fingertip. Users typed the fastest at 13 wpm using a QWERTY keyboard after four hours of practice. Speicher et al. [22] investigated text entry techniques using an HMD, hand-held controllers, and visualization of hands sensed via a Leap Motion. Users wrote at 15 wpm with a low error rate of 1% by pointing using hand-held controllers. Using the hand visualization users wrote at 10 wpm with a much larger error rate of 7.6%. We used a similar hand visualization technique in our study. Vulture [18] allowed users to wear gloves and let users write on a word-gesture keyboard in midair. After five hours of practice, users wrote at 20 wpm.

Bimanual text entry. Bimanual text entry have been investigated for touchscreens [6,24,20,29] and game controllers [21]. Bi et al. [6] and Truong et al. [24] found bimanual gesture typing on touchscreens yielded better performance than unimanual input. Oulasvirta et al. [20] explored a bimanual split keyboard called KALQ. However, they did not compare unimanual and bimanual input. Using two hands, users typed at 37 wpm after one hour of practice. Sandnes et al. [21] showed bimanual game controller input on a QWERTY keyboard had an entry rate of 7 wpm.

Aschim et al. [4] studied one- and two-handed typing performance on a split touchscreen tablet keyboard. Similar to their finding, we found that a split keyboard does not provide better midair typing performance compared to a normal QWERTY layout. Alamdar et al. [3] proposed a new split keyboard layout for improving text entry rate by optimizing different split keyboard layouts and key dimensions. They reported between 7% to 18% improvement in the text entry rate over the other split keyboards. While the past work has investigated bimanual user interaction with touchscreen keyboards, in this paper we conduct the first study comparing unimanual and bimanual input on a midair virtual keyboard.

Key aspects of our work. Compared with past work, we focus on designing an input method that does not require user or system training, and does not require hand-held devices, gloves, or expensive tracking infrastructure. We also investigate reducing the visual occlusion of the keyboard by splitting the key-

board to allow better perception of the central visual area, and by eliminating almost all visual keyboard elements.

3 Interface Design

Midair QWERTY keyboard layouts. Given the widespread familiarity with QWERTY desktop and touchscreen keyboards, this layout was a clear choice for providing walk-up-and-use functionality. Our system renders a QWERTY keyboard 40 cm in front of the user (Figure 1). Our system either displays a normal keyboard or a split keyboard. The normal keyboard is 20 cm \times 7.5 cm. Each half of the split keyboard is 10 cm \times 7.5 cm separated by 10 cm. We divided keys between the two halves based on Apple’s iPad split keyboard. We chose the distance between the split halves keeping two things in mind: 1) the halves are positioned such that they are comfortably within the reach of a user’s hands when extended, and 2) the user does not have to rotate their head when switching attention between the two halves. These design choices are further supported by Bachynskyi et al. [5] who suggested splitting the input space for the right hand and the left hand when making pointing gestures.

Vertical keyboard orientation. We placed the keyboard vertically in front of the user. This allowed users to see the keyboard and their input with minimal head movement. We opted not to use a horizontal keyboard orientation. As our depth sensor was mounted on the HMD, a horizontal keyboard would require a user to bend their neck which could be strenuous. Also in many use scenarios, e.g. messaging in a game, users may want to visually attend to things in front of them while also visually guiding their fingers over the keyboard.

Hand tracking and midair tapping. We rendered a user’s hands via the default visualization in the Leap Motion Orion Beta SDK v3.2. Possible alternatives to a Leap Motion sensor such as a Vicon tracking array may be more accurate, but are expensive, not very portable, and may require wearing special clothing or markers. We wanted to use a tracker that was inexpensive, walk-up-and-use, and portable. The Leap Motion meets all these requirements.

We opted to design our interface around the familiar interaction of tapping visual objects. This provides an easy-to-understand interaction primitive for users, namely making their (virtual) hands contact an object in three-dimensional space. Compared to an approach based on continuous gestures, we thought the tapping primitive would be more robust to tracker inaccuracies and also eliminate the need to train users on how to start or stop gestures.

We displayed the keyboard and rendered a user’s hands in the virtual environment. A key was registered and a click sound was played whenever a user’s index finger crossed the keyboard plane. Tapping a key caused the nearest key to light up for as long as the finger remained through the keyboard plane. The letter on the key nearest to a tap was added in the text area above the keyboard.

Input using index fingers. While we can detect any finger crossing the keyboard plane, we limited our interface to detecting just index fingers. On touchscreens, users usually type with index fingers or thumbs. We choose to use

index fingers as thumbs are less precise and unconventional based on prior midair interaction work. In piloting, we found the use of index fingers was indeed the most usable option. While users might switch fingers on a tablet, on a tablet users know when any finger accidentally contacts the touchscreen. In midair, there is no such tactile feedback. This could result in extra key presses due to the sympathetic motion of all of a hand's fingers. We found these extra key presses were surprising to users and difficult to avoid in the heat of text entry.

Backspace and space key. We expected users would occasionally trigger the wrong key due to inaccuracies in the hand tracking, the virtual hand visualization, or the virtual keyboard visualization. A backspace key in the lower right of the keyboard deleted previous taps from the pending taps. After typing all the letters of a word, users pressed a space key. The split keyboard had a space key on both sides allowing whatever hand was convenient to tap space. This is also consistent with Apple iPad's split keyboard design.

Pressing space sent the location of the pending taps for auto-correction (to be discussed shortly). The location of a tap was the two-dimensional coordinate on the keyboard plane where the tip of a user's index finger crossed the plane. After pressing space, the nearest key text for the pending taps was replaced with the best recognition result. Immediately after recognition, pressing the backspace key deleted the entire previous recognition result rather than backspacing individual characters. This allowed users to quickly delete recognition errors.

Auto-correction. Given the tracking and visualization challenges, as well as the lack of tactile feedback, the only hope of reasonably fast midair keyboard typing is to allow noisy user input but provide a strong auto-correction capability. We used the VelociTap decoder [26] for auto-correction. VelociTap takes the noisy tap locations as input and searches for the most likely word based on a probabilistic keyboard model, a character language model, and a word language model. It assumes tap locations follow a two-dimensional Gaussian distribution centered at each key. Each key is assumed to have the same distribution. The distribution's variance in the horizontal and vertical axes are independently controlled by two decoder parameters. For each tap, VelociTap calculates the likelihood of each key under the keyboard model. This likelihood is added to the probabilities from the decoder's character and word language models. The contribution of each language model is controlled by two additional parameters. The decoder also has two penalties allowing taps to be deleted, and characters to be inserted without a tap. We optimized the decoder's configurable parameters using data collected by five people who did not participate in our user study.

The decoder used a 12-gram character and a 4-gram word language model with a 100 K vocabulary. We trained the language models on billions of words of data from web forums, social media, and movie subtitles. The character and word language models had 25 M and 13 M n-grams respectively. Recognition used any previous text written for the current sentence as context for the language models. In our study, we opted not to provide other features such as word predictions. We wanted to focus on the performance of unimanual versus bimanual interaction, and on the two keyboard designs.

4 User Study

The goal of our main study was to explore unimanual and bimanual midair keyboard input using hand gestures sensed via a commodity sensor. We also wanted to see if we could reduce occlusion in the central visual area by splitting the keyboard. The split keyboard separates keys into two halves, potentially making hand-tracking or recognition by the decoder more accurate.

4.1 Participants

We recruited 24 participants via convenience sampling. None had uncorrected vision or motor impairments. Participants were aged 18–44 (mean 26.5, sd 6.8), 17 were male and 7 were female. 22 were right-handed and 2 were left-handed. All were familiar with QWERTY keyboards. 15 participants had used VR previously.

4.2 Experimental Design

We designed a within-subject experimental study with three counterbalanced conditions: UNIMANUAL, BIMANUAL, and SPLIT. In UNIMANUAL, we instructed participants to tap with the index finger of their dominant hand. In BIMANUAL and SPLIT, we told participants to tap with the index finger of both hands. SPLIT used the split QWERTY layout while UNIMANUAL and BIMANUAL used the normal layout.

4.3 Procedure

Participants first filled out a questionnaire asking demographic questions, and about their experience with text entry and VR. We seated participants at a desk and helped them adjust the HTC Vive HMD. The HMD had a Leap Motion controller mounted on the front. We gave participants a few minutes to become familiar with the virtual environment. During this familiarization period, we had participants move their head, lift both hands, and move their virtual hands.

We first explained to participants how the decoder’s auto-correction works. Participants then practiced in each condition. They practiced conditions in the same order they would experience them in the evaluation. In each condition, participants wrote four phrases during practice and 12 during evaluation. We used the `mem1–5` phrases from the Enron mobile dataset [25]. Participants never saw the same phrase twice. They had as long as they wanted to memorize phrases. Requiring that participants memorize phrases slightly increases entry rate at the expense of slightly increasing error rate [15]. To motivate participants and help them monitor their performance, we showed the entry and error rate after each phrase. We asked participants to enter phrases “quickly and accurately”.

After each condition, participants completed a questionnaire and rated their exertion using the Borg CR10 scale [7]. The study including an exploratory session (to be discussed in Section 5) took approximately an hour.

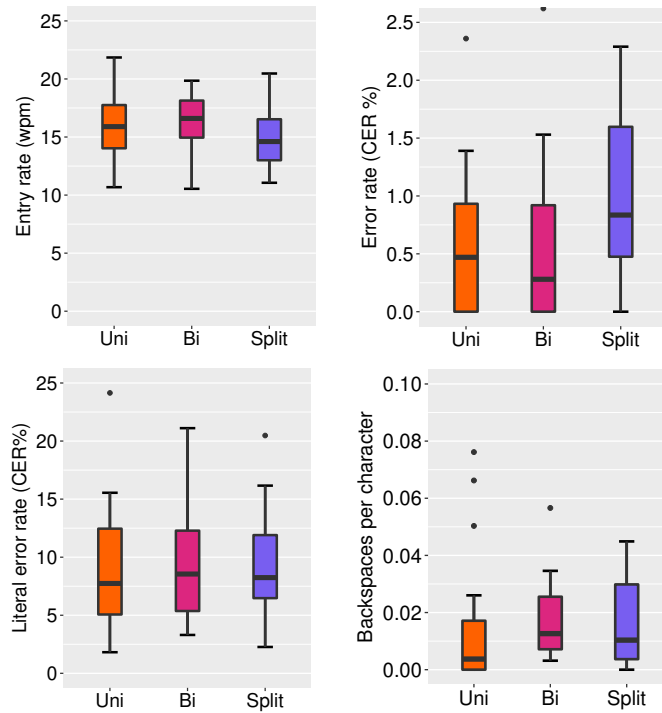


Fig. 2. Entry rate, character error rate (after recognition), literal error rate (before recognition), and backspaces per character in the study.

4.4 Results

Figure 2 shows our main results. Table 1 gives numeric results and statistical tests. In 10 phrases out of 864, participants left off two or more words at the end of a phrase. Likely this was because they forgot the phrase. We removed these instances from our analysis. This affected at most two phrases from any particular participant in any condition. Unless otherwise stated, we tested for significance using repeated measure analysis of variance (ANOVA). For pairwise comparison, we used paired t-tests and adjusted p-values using Bonferroni correction to guard against overtesting.

Entry rate. We calculated *entry rate* in words-per-minute (wpm). We considered a word to be five characters including space. We measured the duration of entering a phrase as the time between user tapping the first key of the phrase and tapping a done button. The done button was located below and to the right of the keyboard. Participants' mean entry rate was 16.1 wpm (sd 2.9) in UNIMANUAL, 16.4 wpm (sd 2.3) in BIMANUAL, and 14.7 wpm (sd 2.4) in SPLIT. An ANOVA test was significant (Table 1). Post-hoc tests showed SPLIT was slower than BIMANUAL. Other pairwise comparisons were not significant.

Condition	Entry rate (wpm)	Error rate (CER %)	Literal error rate (CER %)	Backspaces per character
UNIMANUAL	16.1 ± 2.9 [10.7, 21.9]	0.7 ± 0.9 [0, 3.0]	8.8 ± 5.4 [1.8, 24.1]	.014 ± .021 [.00, .08]
BIMANUAL	16.4 ± 2.3 [10.5, 19.9]	0.8 ± 1.2 [0, 4.4]	9.8 ± 5.6 [3.3, 21.1]	.017 ± .013 [.00, .05]
SPLIT	14.7 ± 2.4 [11.1, 20.5]	1.4 ± 1.5 [0, 5.9]	9.0 ± 4.4 [2.3, 20.5]	.017 ± .016 [.00, .05]
ANOVA	$F_{2,46} = 5.52, p < 0.01$	$F_{2,46} = 2.31, p = 0.11$	$F_{2,46} = 0.45, p = 0.64$	$F_{2,46} = 0.36, p = 0.7$
Effect size	$\eta_p^2 = 0.19$	$\eta_p^2 = 0.09$	$\eta_p^2 = 0.02$	$\eta_p^2 = 0.015$
Post-hoc	UNI \approx BI, $p = 1.0$	Not applicable	Not applicable	Not applicable
	UNI \approx SPLIT, $p = 0.079$			
	SPLIT $<$ BI, $p < 0.05$			

Table 1. Results are formatted as: mean ± SD [min, max]. The bottom section of the table shows the repeated measures ANOVA statistical test for each dependent variable. For significant main effects, we show pairwise post-hoc tests (Bonferroni corrected).

Error rate. We measured *error rate* using Character Error Rate (CER). CER is the number of character insertions, deletions, and substitutions needed to change the participant’s final text into the reference text divided by the total characters in the reference. Error rate was similar and low across all conditions: UNIMANUAL 0.74% (sd 0.9%), BIMANUAL 0.79% (sd 1.2%), and SPLIT 1.41% (sd 1.5%). An ANOVA test was not significant (Table 1). All participants had a CER of 3% or less with many achieving near perfect accuracy (Figure 3). Error rate was more variable in SPLIT. We conjecture this may be due to the sensor or the user being less accurate away from the keyboard center. We will investigate this further shortly.

We also measured the *literal CER* by comparing the text before auto-correction with the reference. Literal CER was much higher and similar across all conditions: UNIMANUAL 8.8% (sd 5.4%), BIMANUAL 9.8% (sd 5.6%), and SPLIT 9.0% (sd 4.4%). An ANOVA test was not significant (Table 1). The high literal CERs shows the importance of auto-correction for enabling accurate midair typing.

Interkey time. Interkey time was calculated as the time difference between two consecutive taps of letter keys in all the entered words. The interkey time in BIMANUAL was 0.62 s (sd 0.10), in UNIMANUAL was 0.65 s (sd 0.13), and in SPLIT was 0.71 s (sd 0.13). An ANOVA test was significant. In post-hoc tests, we found similar to entry rate, only SPLIT was significantly slower than BIMANUAL (SPLIT $<$ BIMANUAL, $p < 0.05$; UNIMANUAL \approx BIMANUAL, $p = 0.15$; UNIMANUAL \approx SPLIT, $p = 0.10$). Thus, locating and tapping keys in SPLIT did seem to contribute to that condition’s slower entry rate.

Correction and tap behavior. Participants rarely used backspace. The backspaces per final output character were: UNIMANUAL 0.014, BIMANUAL 0.0169, and SPLIT 0.0166. An ANOVA test was not significant (Table 1). Recall right after recognition, tapping backspace deleted the recognized word. Word deletions per output word was low in all conditions: UNIMANUAL 0.024, BIMANUAL 0.032, and SPLIT 0.040. Taken together, it seems participants trusted auto-correction and, as evidenced by the low final CER, it delivered acceptable accuracy.

We were interested how often participants used their right index finger to tap a key on the left side of the keyboard and vice-versa. Figure 4 shows all

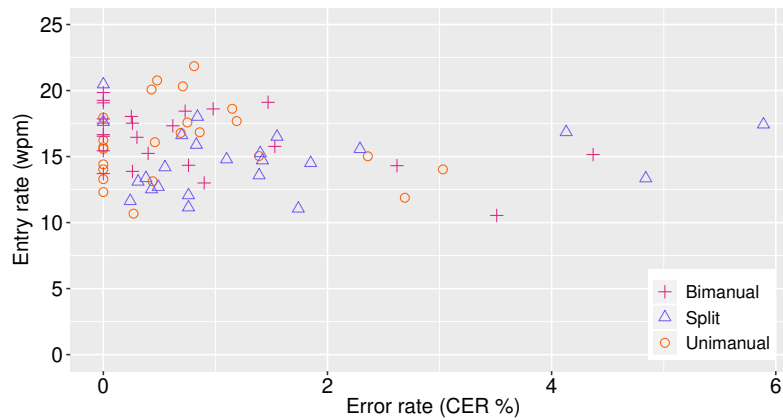


Fig. 3. Error rate and entry rate of all participants in each of the three conditions.

taps in the two bimanual conditions. We can see in BIMANUAL, participants frequently typed letters on the left side of the keyboard with their right hand. This even happened in SPLIT, albeit to a lesser extent. In BIMANUAL, despite 53.9% of the reference phrase letters being on the left side of the keyboard, only 50.2% of participants' taps were with their left hand. This shows a tendency for participants (who were almost all right-handed) to favor their dominant hand.

4.5 Subjective Feedback

After each condition, participants rated statements on a 5-point Likert scale (1=strongly disagree and 5=strongly agree). The mean rating for "I entered text quickly" was: UNIMANUAL 4.17, BIMANUAL 4.08, and SPLIT 3.75. A Friedman's test was not significant ($\chi^2(2) = 3.96$, $p = 0.14$).

The mean rating for "I entered text accurately" was: UNIMANUAL 3.88, BIMANUAL 3.54, and SPLIT 3.17. A Friedman's test was significant ($\chi^2(2) = 10.27$, $p < 0.01$). SPLIT was significantly lower than UNIMANUAL (difference = 18.5, critical = 16.6). Other pairwise differences were not significant (BI-SPLIT 11.5, BI-UNI 7.0). This shows that participants noticed the lower accuracy of SPLIT.

After each condition, we asked participants for a positive and negative aspect of that condition. Table 2 shows a list of such comments. At the end of our study, we asked participants to rank conditions in terms of quickness, accuracy, effort, and overall. The most preferred conditions were as follows:

- Quickness — BIMANUAL 10, UNIMANUAL 8, SPLIT 6
- Accuracy — BIMANUAL 9, UNIMANUAL 9, SPLIT 6
- Effort — UNIMANUAL 11, BIMANUAL 8, SPLIT 5
- Overall — BIMANUAL 10, UNIMANUAL 7, SPLIT 7

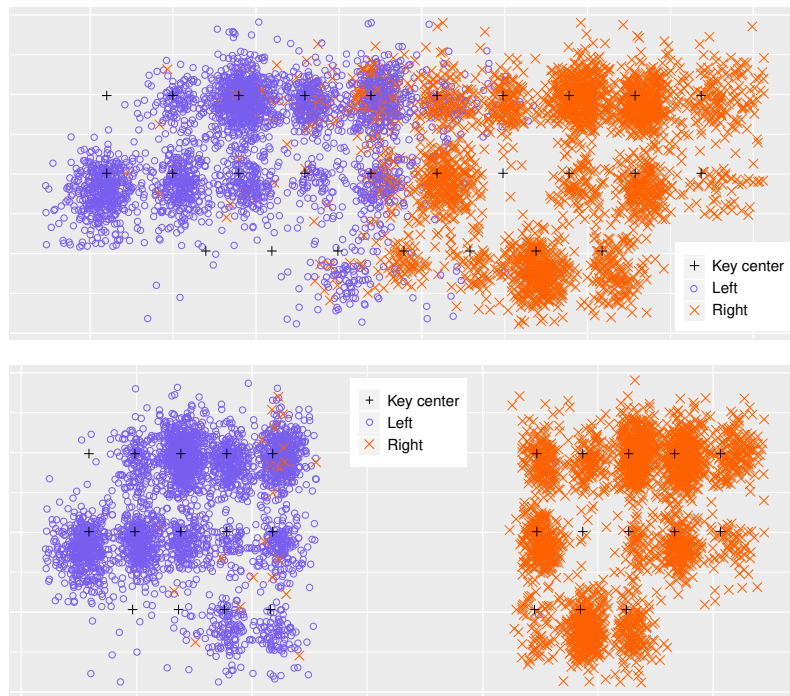


Fig. 4. Taps with the left and the right index finger in BIMANUAL (top) and SPLIT (bottom). Center of the keys are shown for better visualization.

Participants rated their exertion on the Borg CR10 scale [7] where 0=no exertion and 10=extremely strenuous. Exertion in all conditions corresponded to “moderate exercise”: UNIMANUAL 3.38, BIMANUAL 3.08, and SPLIT 3.04. A Friedman’s test was not significant ($\chi^2(2) = 3.27$, $p = 0.20$). While we had anticipated bimmanual input would be more exerting, participants rated it similarly.

Taken in aggregate, the subjective feedback shows participants were varied in their perceptions and preferences about the different visual keyboard layouts (single versus split) and interaction styles (unimanual versus bimmanual). This suggests midair virtual keyboards may want to support several layouts and support both one- and two-handed typing.

4.6 Further analysis

Quantitative results (e.g. Figure 3) show that six participants had an error rate of more than 3.0% in various conditions. This error rate and open comments from participants suggested that some participants experienced occasional hand tracking issues, especially in the SPLIT and BIMANUAL conditions. In our pilot testing with five people prior to the study, we did not encounter tracking issues with the keyboard layouts or interaction styles. In the study, however,

UNIMANUAL	
+	“This felt faster and easier”
+	“I became more comfortable using this system”
-	“It’s tiring while using only one index finger”
-	“At time the virtual fingers would move on their own”
BIMANUAL	
+	“This felt more natural”
+	“I definitely felt more confident”
-	“Seemed to flash when both hands overlapped”
-	“Hands got in each other’s way”
SPLIT	
+	“It became easier to play with the keyboard”
+	“It was a learning experience”
-	“Had trouble typing accurate with my left hand”
-	“I felt my dominant hand was more accurate”

Table 2. Selected positive and negative comments from the study.

four participants in particular seemed to have issues. It is possible something about their particular hand motion or relative distance from the Leap Motion controller made their hands particularly difficult to track.

To investigate this further, we reviewed screen recordings of all the participants’ sessions. We flagged 12 of the 854 phrases written in all conditions as having hand tracking issues. Among the flagged phrases, eight were in SPLIT, three were in BIMANUAL, and one was in UNIMANUAL. We removed the flagged phrases from the data and recomputed the entry and error rate in each condition. Removing these phrases and recomputing the entry rate and error rate yielded similar results. Entry rates after filtering were: 16.4 wpm UNIMANUAL, 16.6 wpm BIMANUAL, and 14.7 wpm SPLIT. Character error rates were: 0.7% UNIMANUAL, 0.7% BIMANUAL, and 1.2% SPLIT.

5 Design Exploration: Invisible Keyboard

After completing the main study, participants took part in a final design exploration. The focus of this part was to investigate input of small text passages where a visual keyboard may not be possible or desirable. For example, a user may be in an instrumented environment (e.g. car) and need to lookup a contact. To achieve this, the user could trace a rectangle in midair and then type the contact’s name in that spatial area. Or in a VR game, a player may want to type a quick message to another player. The player could keep their eyes on their environment while specifying a keyboard off to the side, typing their message using their peripheral vision or motor memory.

In addition to the above rationale, this exploration had two additional objectives. First, given the freedom to define a keyboard in the virtual environment, we were interested what keyboard size participants would choose. Second, after

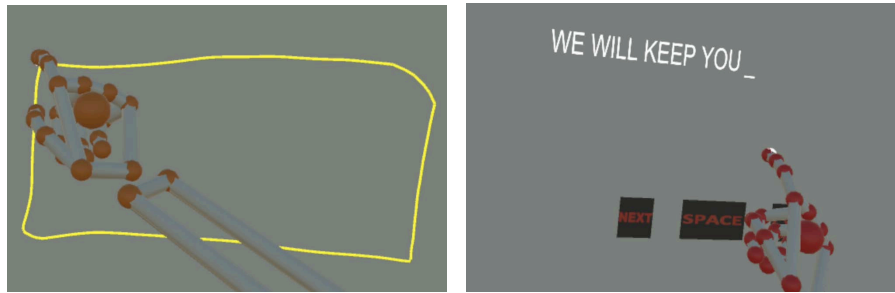


Fig. 5. User defining the keyboard geometry (left), and entering text on an invisible keyboard with almost no visual feedback (right).

the participants had been exposed to unimanual and bimanual midair input in the previous experiment, we wanted to see which method they would choose if they were allowed to use either interaction style.

5.1 Procedure

Users first defined the keyboard's size and location by tracing a rectangle with their index finger. A line was displayed as the rectangle was drawn (Figure 5 left). Users made a thumbs up gesture once they completed their rectangle. The system then drew a keyboard within the rectangle. Users could accept the keyboard geometry or define it again. During input using the invisible keyboard, the only visual feedback was the space key, the backspace key, a key to advance to the next sentence, and the current text (Figure 5 right).

In this exploratory session, there was no practice period and participants typed 12 phrases. We dropped four participants due to technical issues. We found in 14 phrases out of 240, participants forgot part or all of the target phrase. We removed these phrases from our analysis.

5.2 Results

Entry rate and error rate. On average, participants wrote at 10.6 wpm (sd 3.0, min 5.5, max 17.4) with an error rate of 3.3% (sd 3.0%, min 0.5%, max 13.1%). As might be expected, input was slower compared to the 15–16 wpm seen in the main study with a visible keyboard. Participants were able to achieve a completely correct input for 71% of their entries.

Backspaces-per-character. Compared to our main study, we observed a substantial increase in backspacing. The backspace to output character ratio was 0.05 (sd 0.04, min 0.0, max 0.13). The deleted word to final word ratio was 0.15 (sd 0.09, min 0.0, max 0.33).

One versus two hands. We allowed participants to type with one or both hands. We observed 15 out of 20 participants used both hands while the remainder used one hand almost exclusively. For the given reference phrases, 52.1%



Fig. 6. The keyboard rectangles defined by participants in the invisible keyboard. The numeric values are the keyboard areas in square centimeters. Rectangles are arranged in a row-order matrix by ascending area.

letters were on the left side of the keyboard and 47.9% letters were on the right side. Participants tapped 36% of keys using their left hand versus 64% using their right hand. Thus it seems that with the invisible keyboard, participants tended to use their dominant hand even more than in the BIMANUAL condition of the main study.

Keyboard geometry. Participants defined their keyboard geometry on average 2.0 times. 11 participants defined the keyboard in a single attempt. Participants defined keyboards of various sizes (Figure 6). The mean area of the defined rectangles was 495 cm^2 (sd 254, min 155, max 1103). The area of the QWERTY normal keyboard rectangle used in the main study was 150 cm^2 . Thus all participants defined a larger keyboard than the one in the main study. This suggests keyboard developers consider a bigger default geometry or allow users to define their own geometry.

Subjective feedback. At the end of this session, participants rated statements on a 5-point Likert scale (1=strongly disagree and 5=strongly agree). Participants rated the statement “I entered text quickly” at 3.00, and “I entered text accurately” at 2.55. Participants rated the statement “I successfully obtained my desired keyboard size” at 4.20. The mean rating for the statement “I found it easy to enter text without any visual feedback” was 3.05. The mean rating for the statement “I was able to easily understand when and what key I typed” was 3.25. Thus most participants were satisfied with their ability to draw the keyboard as they wanted. However, the ratings on the other statements suggest the invisible keyboard was perceived as slow and not that accurate. Open comments were in general positive (Table 3). Most of the participants remarked that the invisible keyboard was easier than they thought it would be.

INVISIBLE
+ “Hitting a close enough key was good enough”
+ “The adjustable size of the keyboard is good”
+ “It got easier with practice - when I understood my keyboard”
+ “It was fun. Easier than I thought it would be”
- “The keyboard was too small to use for me”
- “Sometimes the fingers were folded”

Table 3. Selected positive and negative comments from exploratory session.

6 Discussion

We explored how to enable efficient text entry in virtual environments without the use of auxiliary input devices. Our system relied on midair hand gestures. Further, we aimed to design a system that could be used with little or no training. In our study, we focused on how users interact with a familiar virtual keyboard interface. We anticipated bimanual typing would be faster, but our results showed similar entry rates for unimanual and bimanual typing. We think there are a number of possible explanations for this:

1. We conducted a single one-hour session. It is possible users may take more time to become accurate tapping midair targets in VR, especially with their non-dominant hand.
2. Our midair entry rates were also relatively slow at 16 wpm. This speed is consistent with the 18 wpm entry rate reported on an AR midair keyboard [9]. These slow speeds may be because users are struggling to precisely target keys or trigger midair taps. If users are focused on visually guiding their finger or on successfully triggering a tap, they may not be able to effectively plan for their subsequent tap. At least for touchscreen bimanual typing, it has been observed that users employ strategies such as pre-positioning over the next letter [20]. It could be that midair bimanual tapping is too cognitively taxing to allow effective use of such strategies.
3. It could also be that users had to visually guide their finger to each target. Since their hands were in midair, they could not anchor and use motor memory like they can do on small form factor devices like a phone. It would be interesting to investigate typing performance with a higher accuracy hand tracker and with tapping on a rigid surface overlaid with a virtual keyboard.

We found typing on a midair keyboard requires a good auto-correction algorithm. Participants’ error rate was around 9% before auto-correction but only around 1% afterwards. The infrequent use of backspaces indicates that participants were largely relying on the system to automatically correct their input.

In our study, users had to tap all of a word’s characters. Users tapped the spacebar to send the noisy input sequence to the auto-correct algorithm. Sometimes, the recognized word was different from a user’s intended word. This occasionally led to time consuming correction episodes. This issue could be mitigated

by adding predictions slots above the keyboard. These slots would allow users to understand what they will get if they tap the spacebar. These slots could also provide alternative options such as word predictions based on the current prefix of a word. If the system needs to support the input of uncommon words such as proper names, a slot providing the literal keys typed may be helpful.

Our study also tested a split keyboard layout. In principle, the split layout forces hand separation which should make tracking easier as it avoids one hand occluding the other. However, we found the error rate before and after correction was similar to the normal layout. The split layout was also slightly slower at 15 wpm compared to the normal layout. We suspect this was because either users had trouble locating which side of the keyboard a letter was on, or it forced more use of a user's non-dominant hand. It may require a longitudinal study to understand if a split layout is a useful design.

Despite the lack of performance advantages to bimanual typing, participant opinions were mixed with some preferring bimanual and some preferring unimanual. We also found participants rated physical exertion similar for bimanual and unimanual typing. Thus it may be worth supporting both input styles on midair keyboards.

We found our invisible keyboard was quite successful in allowing users to enter text without any visual feedback of the keys. Moreover, we found text can be inferred from an invisible keyboard with variable sizes. Whether typing performance depends on a smaller versus larger keyboard size needs further investigation, but our findings suggest that in VR use scenarios where users want to visually attend to other visual content, a user-defined invisible keyboard may be a viable approach.

We think our keyboard designs will be useful in scenarios where users want to send a short message, e.g. while playing a game. It could also be useful in virtual chat rooms or when searching for something in a VR application. We did not look at entering and editing large text passages. We think our current design is best suited for small amounts of text. Supporting efficient entry and editing of large amounts of would require design of features allowing the user to select and change regions of text, and allowing navigation through a passage that might not fit in the HMD's field of view.

One shortcoming of current AR and VR systems is the lack of haptic feedback during interaction with virtual objects. While in our design, we provided audio and visual feedback to signal a keyboard tap, providing additional haptic feedback may be beneficial. This could be done, for example, by aligning the keyboard with a physical surface, or by vibrating a wearable device such as a smartwatch.

7 Conclusion

We investigated text entry in virtual environments on a midair virtual keyboard. We compared two keyboard designs: a normal QWERTY layout and a split layout. We investigated the speed, accuracy, ergonomics, and user satisfaction of

both these designs. We also compared unimanual versus bimanual interaction on these two different keyboard layouts. We found novice users' performance was similar at around 15–16 words-per-minute with a low error rate of less than 2% for the different visual designs and interaction styles.

Novice users were able to easily learn to use the system and achieved accurate text input despite inaccuracies introduced by the hand tracking, keyboard and hand visualization, and the lack of haptic keyboard feedback. Users were mixed in their preference on typing with one or two index fingers. We found participants reported similar exertion for one- and two-handed interaction.

Finally, we explored a design with only minimal visual feedback. Despite the lack of any visible keyboard or key outlines, users were able to type at 11 words-per-minute at a low 3% error rate. We hope our findings will inform and advance the design of improved text entry methods for use in virtual environments.

Acknowledgments

This material is based upon work supported by the NSF under Grant No. IIS-1750193.

References

1. Adhikary, J., Vertanen, K.: Text Entry in Virtual Environments using Speech and a Midair Keyboard. *IEEE Transactions on Visualization and Computer Graphics* **27**(5), 2648–2658 (2021). <https://doi.org/10.1109/TVCG.2021.3067776>
2. Ahn, S., Heo, S., Lee, G.: Typing on a Smartwatch for Smart Glasses. In: *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces*. pp. 201–209. ISS '17, ACM, New York, NY, USA (2017), <https://doi.acm.org/10.1145/3132272.3134136>
3. Alamdar Yazdi, M.A., Negahban, A., Cavuoto, L., Megahed, F.M.: Optimization of Split Keyboard Design for Touchscreen Devices. *International Journal of Human-Computer Interaction* **35**(6), 468–477 (2019)
4. Aschim, T.B., Gjerstad, J.L., Lien, L.V., Tahsin, R., Sandnes, F.E.: Are Split Tablet Keyboards Better? A Study of Soft Keyboard Layout and Hand Posture. In: *Human-Computer Interaction – INTERACT 2019*. pp. 647–655. Springer International Publishing, Cham (2019)
5. Bachynskyi, M., Palmas, G., Oulasvirta, A., Weinkauff, T.: Informing the Design of Novel Input Methods with Muscle Coactivation Clustering. *ACM Transactions on Computer-Human Interaction* **21**(6) (Jan 2015), <https://doi.org/10.1145/2687921>
6. Bi, X., Chelba, C., Ouyang, T., Partridge, K., Zhai, S.: Bimanual Gesture Keyboard. In: *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*. pp. 137–146. ACM (2012)
7. Borg, G.: Borg's Perceived Exertion and Pain Scales. *Human Kinetics* (07 1998)
8. Dube, T.J., Arif, A.S.: Text Entry in Virtual Reality: A Comprehensive Review of the Literature. In: *Human-Computer Interaction. Recognition and Interaction Technologies. Lecture Notes in Computer Science*, vol. 11567, pp. 419–437. Springer International Publishing, Cham (2019), https://doi.org/10.1007/978-3-030-22643-5_33

9. Dudley, J.J., Vertanen, K., Kristensson, P.O.: Fast and Precise Touch-Based Text Entry for Head-Mounted Augmented Reality with Variable Occlusion. *ACM Transactions on Computer-Human Interaction (TOCHI)* **25**(6) (12 2018). <https://doi.org/10.1145/3232163>
10. Feit, A.M., Sridhar, S., Theobalt, C., Oulasvirta, A.: Investigating Multi-Finger Gestures for Mid-Air Text Entry. In: *ACM womENCourage* (2015)
11. Grubert, J., Witzani, L., Ofek, E., Pahud, M., Kranz, M., Kristensson, P.O.: Effects of Hand Representations for Typing in Virtual Reality. In: *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. pp. 151–158 (2018)
12. Grubert, J., Witzani, L., Ofek, E., Pahud, M., Kranz, M., Kristensson, P.O.: 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)*. pp. 159–166 (2018)
13. Gupta, A., Samad, M., Kin, K., Kristensson, P.O., Benko, H.: Investigating Remote Tactile Feedback for Mid-Air Text-Entry in Virtual Reality. In: *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. pp. 350–360. IEEE (2020)
14. Knierim, P., Schwind, V., Feit, A.M., Nieuwenhuizen, F., Henze, N.: Physical Keyboards in Virtual Reality: Analysis of Typing Performance and Effects of Avatar Hands. In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. pp. 1–9 (2018)
15. Kristensson, P.O., Vertanen, K.: Performance Comparisons of Phrase Sets and Presentation Styles for Text Entry Evaluations. In: *Proceedings of the 2012 ACM International Conference on Intelligent User Interfaces*. pp. 29–32. IUI '12, ACM, New York, NY, USA (2012), <https://doi.acm.org/10.1145/2166966.2166972>
16. Lee, M., Woo, W.: ARKB: 3D Vision-based Augmented Reality Keyboard. In: *Proceedings of the 13th International Conference on Artificial Reality and Telexistence. ICAT'03* (2003)
17. Markussen, A., Jakobsen, M.R., Hornbæk, K.: Selection-based Mid-air Text Entry on Large Displays. In: *IFIP Conference on Human-Computer Interaction*. pp. 401–418. Springer (2013)
18. Markussen, A., Jakobsen, M.R., Hornbæk, K.: Vulture: A Mid-Air Word-gesture Keyboard. In: *Proceedings of the 32nd Annual ACM Conference on Human factors in Computing Systems*. pp. 1073–1082. ACM (2014)
19. McGill, M., Boland, D., Murray-Smith, R., Brewster, S.: A Dose of Reality: Overcoming Usability Challenges in VR Head-Mounted Displays. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. pp. 2143–2152. CHI '15, ACM, New York, NY, USA (2015), <https://doi.acm.org/10.1145/2702123.2702382>
20. Oulasvirta, A., Reichel, A., Li, W., Zhang, Y., Bachynskyi, M., Vertanen, K., Kristensson, P.O.: Improving Two-Thumb Text Entry on Touchscreen Devices. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. pp. 2765–2774. ACM (2013)
21. Sandnes, F.E., Aubert, A.: Bimanual Text Entry Using Game Controllers: Relying on Users' Spatial Familiarity with QWERTY. *Interacting with Computers* **19**(2), 140–150 (2006)
22. Speicher, M., Feit, A.M., Ziegler, P., Krüger, A.: Selection-based Text Entry in Virtual Reality. In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. pp. 1–13 (2018)

23. Sridhar, S., Feit, A.M., Theobalt, C., Oulasvirta, A.: Investigating the Dexterity of Multi-Finger Input for Mid-Air Text Entry. In: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems. pp. 3643–3652. ACM (2015)
24. Truong, K., Hirano, S., Hayes, G.R., Moffatt, K.: 2-Thumbs Gesture: The Design and Evaluation of a Non-Sequential Bi-manual Gesture Based Text Input for Touch Tablets. Tech. rep., Knowledge Media Design Institute (KMDI), University of Toronto, Canada (2013)
25. Vertanen, K., Kristensson, P.O.: A Versatile Dataset for Text Entry Evaluations Based on Genuine Mobile Emails. In: Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices & Services. pp. 295–298. MobileHCI '11, ACM, New York, NY, USA (2011). <https://doi.org/10.1145/2037373.2037418>
26. Vertanen, K., Memmi, H., Emge, J., Reyas, S., Kristensson, P.O.: VelociTap: Investigating Fast Mobile Text Entry Using Sentence-Based Decoding of Touchscreen Keyboard Input. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 659–668. CHI '15, ACM, New York, NY, USA (2015), <https://doi.acm.org/10.1145/2702123.2702135>
27. Walker, J., Li, B., Vertanen, K., Kuhl, S.: Efficient Typing on a Visually Ocluded Physical Keyboard. In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems. pp. 5457–5461. ACM (2017)
28. Wang, C.Y., Chu, W.C., Chiu, P.T., Hsiu, M.C., Chiang, Y.H., Chen, M.Y.: Palm-Type: Using Palms As Keyboards for Smart Glasses. In: Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services. pp. 153–160. MobileHCI '15, ACM, New York, NY, USA (2015), <https://doi.acm.org/10.1145/2785830.2785886>
29. Ye, L., Sandnes, F.E., MacKenzie, I.S.: QB-Gest: QWERTY Bimanual Gestural Input for Eyes-Free Smartphone Text Input. In: International Conference on Human-Computer Interaction. pp. 223–242. Springer (2020)
30. Yi, X., Yu, C., Zhang, M., Gao, S., Sun, K., Shi, Y.: 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 and Technology. pp. 539–548. UIST '15, ACM, New York, NY, USA (2015), <https://doi.acm.org/10.1145/2807442.2807504>
31. Yu, C., Gu, Y., Yang, Z., Yi, X., Luo, H., Shi, Y.: Tap, Dwell or Gesture?: Exploring Head-Based Text Entry Techniques for HMDs. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 4479–4488. CHI '17, ACM, New York, NY, USA (2017), <http://doi.acm.org/10.1145/3025453.3025964>
32. Yu, C., Sun, K., Zhong, M., Li, X., Zhao, P., Shi, Y.: One-Dimensional Handwriting: Inputting Letters and Words on Smart Glasses. In: Proceedings of the SIGCHI Conference on Human factors in Computing Systems. pp. 71–82. CHI'16, ACM (2016)
33. Yu, D., Fan, K., Zhang, H., Monteiro, D., Xu, W., Liang, H.: PizzaText: Text Entry for Virtual Reality Systems Using Dual Thumbsticks. *IEEE Transactions on Visualization and Computer Graphics* **24**(11), 2927–2935 (2018)
34. Zhai, S., Kristensson, P.O.: The Word-Gesture Keyboard: Reimagining Keyboard Interaction. *Communications of the ACM* **55**(9), 91–101 (Sep 2012). <https://doi.org/10.1145/2330667.2330689>