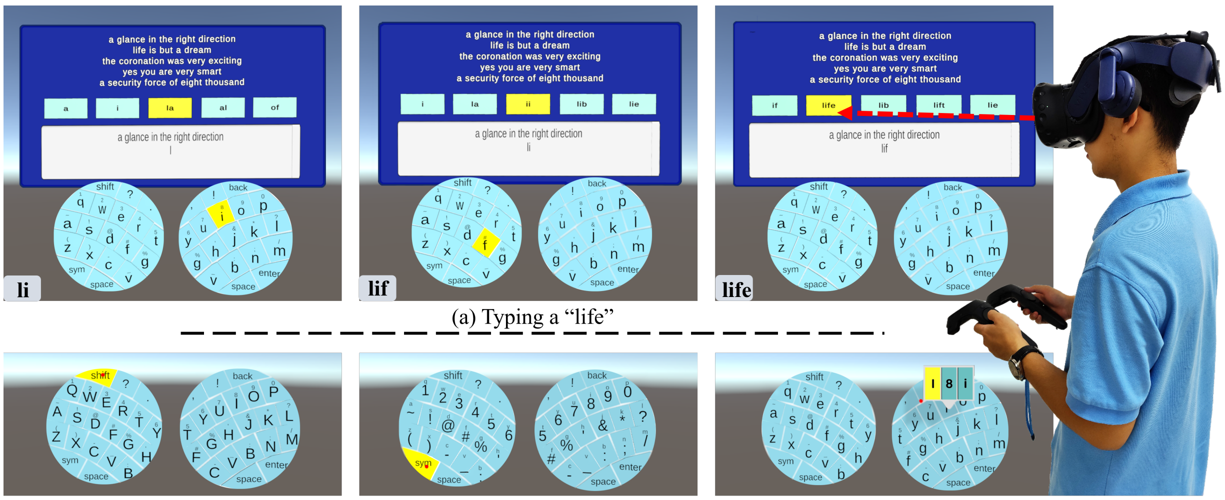
FanPad: A Fan Layout Touchpad Keyboard for Text Entry in VR

Jian Wu[1*] Ziteng Wang[1†] Lili Wang[1,2‡] Yuhan Duan[1§] Jiaheng Li[1¶]

[1] State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing, China
[2] Peng Cheng Laboratory, Shenzhen, Guangdong, China



(a) Typing a "life"

(b) Switching to capitals (c) Switching to symbols (d) Capital/symbol selection box

Figure 1: An overview of FanPad and the typing virtual environment mainly consists of the FanPad, an input box, and a display board above for showcasing the entered text. (a) shows the process of typing a "life" on the layout of FanPad: to type a 'life,' touch the touchpad and release it on the desired key position to enter characters. While doing so, one can simultaneously review the suggested word corrections by turning their head. Confirm the entry by pressing the trigger. (b)-(d) show the optional layout of FanPad, FanPad-Ov with more overlap area and other practical functions: (b) Switching to capitals. (c) Switching to symbols. (d) A multi-selection box to select the uppercase or the symbol corresponding to a key through a longer touch.

Abstract

Text entry poses a significant challenge in the realm of virtual reality (VR). This paper introduces FanPad, a novel solution designed to facilitate dual-hand text input within head-mounted displays (HMDs). FanPad accomplishes this by ingeniously mapping and curving the 26 typing keys (T26) QWERTY keyboard onto the touchpads of both controllers. The curved key layout of FanPad is derived from the natural movement of the thumb when interacting with the touchpad, resembling an arc with a thumb-length fixed radius.

To optimize the experience, we introduce a customization process for the FanPad curve to better cope with individual hand shapes and thumb movements. We also provide a version with more overlap area named FanPad-Ov for different users with different typing habits.

Our first user study examined the effects of curving and different overlap areas by comparing four potential layouts. The results clearly favor the FanPad and FanPad-Ov layout compared to the no-curving version, SKPad(-Ov). Subsequently, the second user study was conducted to assess long-term performance and improvement on customized FanPads. Notably, novices achieved a typing speed of 19.73 words per minute (WPM), demonstrating a remarkable increase of 58.47% after a 60-phrase training in six days. The highest typing speed reached an impressive 24.19 WPM.

Index Terms: Human-centered computing—Human computer interaction (HCI)—Interaction techniques—Text input; Human-centered computing—Human computer interaction (HCI)—HCI design and evaluation methods—User studies

1 Introduction

Text entry is used in various virtual reality (VR) scenarios, such as labeling virtual objects, communicating between multiple users, web browsing, and data annotation. Although many techniques, such as speech [1, 5, 23, 35, 38], sensor-based mid-air typing [13,19,36,45,46,48,50], and gaze-based typing [14,24,38] have been explored, controller-based typing techniques [2,4,11,20,25,41,52] remain the most commonly used method of text entry because of their convenience and do not require additional equipment. Many controller-based text entry methods use controllers to aim [20], knock [4], or shoot [41] to select a target character. The related movement of the arm in large spaces often causes fatigue, leading to a cumbersome experience and inefficient performance. To ease the use of the controller when deploying text entry, some researchers choose to use the controller's touchpad to type in VR [15, 17]. However, these methods did not fully utilize the touchpad area and consider the movement of the user's thumbs, leading to inappropriate

*e-mail: lanayawj@buaa.edu.cn
†Ziteng Wang contributed equally to this paper, e-mail: 21373237@buaa.edu.cn
‡Lili Wang is the corresponding author, e-mail:wanglily@buaa.edu.cn
§e-mail: 21373207@buaa.edu.cn
¶e-mail: 21373169@buaa.edu.cn

keyboard layout, poor typing experience, and adapting difficulty.

This paper introduces a unique multi-letter keyboard with a fan layout specifically designed for touchpad-based dual-hand text entry technique. Our approach involves two steps: mapping and curving. Firstly, we divide the traditional QWERTY keyboard in half and map it to the respective touchpads on two controllers as a keyboard layout named SKPad. Next, the mapping row is curved to align with the natural movements of the thumb, resulting in a distinctive fan-shaped keyboard layout named FanPad. Furthermore, we developed a customization process to tailor FanPad to individual hand sizes and poses, enhancing its overall usability and comfort. We also provide two different overlap areas where characters are mapped to both controllers, leading to four layouts: SKPad, SKPad-Ov, FanPad, and FanPad-Ov. Users could choose the specific layout on their own to gain the most comfortable experience.

Finally, we designed two user studies to evaluate the performance of our methods. The first user study examined the effects of curving and enlarging the overlap area by comparing individual layouts. Compared with SKPad(-Ov), the results clearly favor FanPad(-Ov) in terms of performance and show a significant difference in typing speed and preference between users in terms of the overlap area. Then, the second user study was conducted to assess the efficiency of customized FanPads and analyze users' learning curves. Notably, novices achieved a typing speed of 19.73 WPM, demonstrating a remarkable increase of 58.47% just after a 60-phrase training spanning six days. The highest typing speed during the six-day study reached 24.19 WPM.

The results clearly lean towards the FanPad and FanPad-Ov layouts, in contrast to the no-curving versions named SKPad(-Ov). Furthermore, users demonstrated different preferences between the two overlapping versions, rendering it a customization experience.

To summarize, our FanPad text entry method offers the following contributions:

- We have designed a multi-letter keyboard with a fan layout that aligns with the natural movement of users' thumbs, specifically tailored for touchpad-based text entry techniques.
- We have proposed a customization process to fine-tune the fan keyboard layout, making it adaptable to a wide range of hand poses and sizes.
- We have introduced an overlapping layout customization to adapt to different users with different typing habits.
- We have conducted two comprehensive user studies to assess and evaluate the performance of our FanPad method.

2 Prior Work

This section briefly reviews the recent prior text entry techniques in VR. For a more detailed existing methods examination, please refer to the survey work [8].

2.1 Non-controller-based Techniques

Using physical keyboards to type in VR is a familiar and effective technique [44]. However, it is a problem how to combine the physical keyboards and VR environments well. Many researchers have attempted to overcome the usability challenges in HMDs [32, 43]. Then, to further enhance immersive experiences, Lin and Knierim et al. tried visualizing users' hands and keyboards in the virtual environment (to enhance immersive experiences). Jiang et al. introduced force feedback, combined with the visualized hands and keyboard, to increase the entry efficiency [18]. Recently, Pham et al. introduced a new keyboard, worn on a hawker's tray in front of the user, to offer a flexible and efficient text entry solution [34].

With the development of smart touchscreens, text entry in VR via touch has become popular [12, 21, 22, 28, 31, 37]. Gugenheimer et al. leveraged the back of the HMDs as a touch-sensitive surface, and the user can select virtual content and type by clicking the corresponding location of the touch screen [12]. Lu et al. proposed a technique that allows users to type on a touchscreen with an imaginary QWERTY keyboard [28]. By mounting a mirror above the phone screen, Matullic et al. enabled creating a semi-transparent overlay of thumb to help the user aim for targets [31].

Mid-air techniques allow users to type in mid-air without any physical keyboard or touch devices and provide reasonable text entry speed and accuracy [1, 9, 13, 19, 36, 39, 42, 45, 46, 48–50]. Whitmire et al. present a reconfigurable glove-based input device that enables thumb-to-finger touch interaction by sensing continuous touch position and pressure [45]. Jiang et al. proposed a one-handed text entry technique based on touches between fingers [19]. Then, Xu et al. changed the input device into a miniature fingertip keyboard, and users can type with just his/her thumb-tip and index finger [48, 49]. Recently, Adhikary et al. investigated text entry in VR using hand tracking and speech. Users can speak a sentence and correct errors with the tracked hand on a mid-air keyboard [1].

While all three of these technologies provide reasonable text entry speed and accuracy, the addition of a physical keyboard and a touch screen tends to increase the complexity of the system, and additional motion tracking systems often need to be added in order to achieve consistency between virtual visual feedback and the tactile sensation of the real hand. These cumbersome devices limit the practicality.

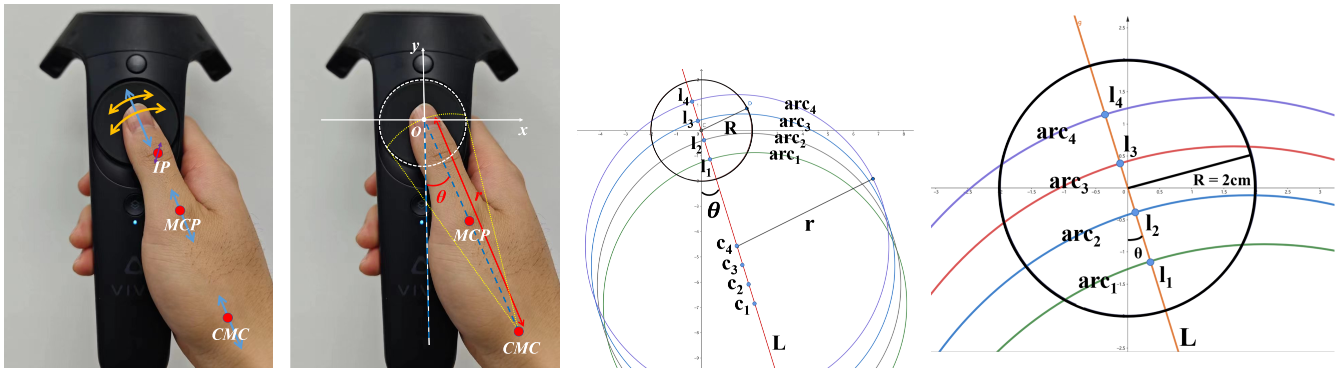
Without using controller and external devices, speech-based text entry techniques in VR have received much attention [1, 5, 23, 35, 38]. Bowman et al. compared the speech technique with the other three methods and found that the speech technique was the fastest [5]. Then, Pick et al. proposed SWIFTER, a speech-based multimodal text entry metaphor, which strives for simplicity and good performance. Recently, Kimura et al. introduced a silent speech text entry using electropalatography. Users can type by spelling words without voicing due to the tracked tongue movement [23]. While speech-based technologies perform well, they typically suffer from noise and privacy issues and error correction problems.

Head and gaze-based techniques have also been explored in VR. Head-based text entry methods were investigated and compared with other methods [41, 51]. RingText [47] was a dwell-free method. The user can use his/her head movement to control the virtual cursor for VR selection. Gaze in text entry is often used as a selection assistant [14, 24, 38]. Kumar et al. use gaze for assisting selection in touch-based text entry [24] and Sengupta et al. for assisting in speech-based text entry system [38]. He et al. present TapGazer, a text entry technique where users type by tapping their fingers in place and then selecting target words with gaze [14]. Recently, Cui et al. proposed GlanceWriter, which allows users to enter text by glancing over keys one by one [6]. Head or gaze-based techniques are likely to cause motion sickness [52] or eye strain.

In summary, non-controller-based technologies either necessitate extra physical keyboards or touchscreen devices, reducing their practicality, or they must be paired with speech recognition or eye-tracking modules, leading to diminished input accuracy or potential dizziness.

2.2 Controller-based Techniques

Handheld controllers are widely used in current VR HMDs. Text entry methods based on the handheld controllers are simple and convenient [2, 4, 11, 20, 25, 41, 52]. Gu et al. use joysticks for text entry in VR [11]. Speicher et al. proposed to generate rays from the controller to point, select, and type words [41]. Boletsis et al. proposed a drum-like VR keyboard, and users can use rays emitted from controllers as the sticks and tap down on the keyboard for selection [4]. Recently, based on the drum-like keyboard technique, Bakar et al. optimized the rays as Crowbar-Limbs to reduce fatigue when typing in VR [2]. Instead of using rays, Yu et al. proposed



(a) (b) (c) (d)

Figure 2: The design of FanPad. (a) A rough representation of the intuitive and effortless motion of the thumb. (b) A mathematical description of the motion: the carpometacarpal (CMC) joint moves along an almost straight path with a vertical inclination angle of θ, and the thumb rotates around the joint at a radius of r. (c) As the CMC joint moves up and down along L, the touchpad is divided into five sections by four corresponding thumb trajectories arc_i. Each of these trajectories corresponds to a circle center denoted as c_i. (d) The arcs design of FanPad layout.

a circular-keyboard-based technique. Users can select characters on the keyboard using the thumbsticks of the game controller [52]. Recently, Leng et al. proposed a single-hand text entry technique by mapping the controller movement on a flower-shaped keyboard to select characters [25].

Since holding and moving the VR controller around the virtual space for text entry tend to cause fatigue, some researchers started to use the touchpad on the HTC Vive controller [15, 17, 26]. Jiang et al. proposed HiPad, which uses the touchpad on a single controller and selects letters on a partitioned circular keyboard [17]. Huang et al. proposed the climbing keyboard [15]. The technique splits the QWERTY keyboard into six parts (three rows and two columns) and uses the controller's tilting for row selection and touchpad for the column. Anh Nguyen et al. proposed a method [33] that divides the touchpad into several sectors and maps the sectors to different letter groups, and users can touch different sections to input. Bret Jackson et al. proposed a pen-based VR text input method [16] that allows users to input by manipulating the pen and does not require spatial positioning. Despite presenting a novel text input model, this study fell short in terms of portability and requires a room-like space. Z. Zhang et al. proposed a novel Bimanual text entry method [53] that designs corresponding sliding gestures for each letter according to its characteristics. Although it can achieve relatively high accuracy and speed, memorizing gestures is relatively burdensome for beginners.

For controller-based techniques, holding and moving the VR controller around the virtual space for text entry tends to cause fatigue, and the previous touchpad-based method does not consider the consistency between keyboard layout and thumb movement and tends to cause unfamiliar experiences and low efficiency. In this paper, we also use a touchpad for text entry and design a fan-shaped keyboard layout to match the movement trajectories of the thumb touching footprint, aiming for a higher text entry speed and lower error rate.

3 Method

We present FanPad, a text entry method for typing on the touchpad of VR controllers based on mapping the keys' positions on the QWERTY layout to the touchpads on both controllers and making adjustments for adapting to thumbs moving on the touchpads. A typical controller with a touchpad is the HTC VIVE controller, which is used as the input device in all of our experiments.

3.1 Keyboard Layout

The design of our FanPad is roughly divided into two stages: mapping and curving stages.

3.1.1 Mapping: SKPad

Inspired by the efficiency and familiarity of typing on a T26 QWERTY keyboard on a smartphone, we propose the *SKPad* layout. This layout optimizes the transition from mobile to touchpad typing, minimizing the learning curve and maintaining high input efficiency.

To achieve this, we divided the QWERTY keyboard on a mobile phone into two halves, mapping each to the respective touchpad on the left and right controllers as shown in Fig. 3. To address variations in users' typing habits, particularly for keys in the middle like 'g' and 'v,' we overlap them with the space key, ensuring dual allocation.

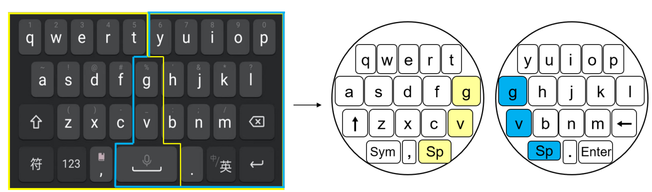


Figure 3: The split keyboard on a phone and the mapping onto touchpads. The rounded quadrilateral is for illustrative purposes, as the keys effectively cover the entire touchpad with even divisions in each row.

3.1.2 Curving: FanPad

After some beta tests on SKPad, we received some feedback of feeling unnatural and stiff when moving the fingers. Based on this feedback, we introduce FanPad to enhance user experience and input efficiency. FanPad takes inspiration from SKPad but introduces a unique adaptation by rotating and curving its keyboard rows to align with the natural movements of the thumb.

Unlike typing on a touchscreen phone where the thumbs rest at the same level as the keyboard, text entry on a controller's touchpad above necessitates a different approach, involving more inclined and extended thumb movements. In this context, having the thumb move horizontally and vertically in relation to the touchpad can be unnatural because these motions engage multiple joints. Besides, users may struggle to sense the horizontal and vertical directions within a VR environment intuitively.

Specifically, Joanna et al. proposed a model of the thumb moving area on a phone held by one hand [3]. Different from their holding gesture and model, our thumb's carpometacarpal (CMC) joint is above the controller and can move freely at any angle, while the CMC joint, the metacarpophalangeal (MCP) joint, the interphalangeal (IP) joint and the fingertip stay roughly on a straight line as the incline range of thumb is small (among 10° to 35°).

Based on these features above, we propose a more intuitive and effortless thumb motion model on the controller, which mainly involves two parts: 1. The fingertip generally traces an arc centered at the CMC joint, with the thumb's length from the CMC joint to fingertip r serving as the radius. 2. The CMC joint moves along a nearly straight line that is vertically inclined at an angle θ, which aligns with the bisector of the central angle corresponding to the arc and the natural inclination angle of the thumb. Additionally, the IP joint may exhibit minor movements perpendicular to the touchpad plane, causing the thumb to fold and changing the radius slightly. However, the influence is relatively small, and we ignore this to simplify the model. The models are illustrated in Fig. 2(a) and Fig. 2(b). and we design the FanPad layout to fit it.

Thumb moving arc. Based on this model, we present a mathematical representation of the thumb's moving arc to optimize FanPad's design.

We initially establish a coordinate system with the touchpad's center as the origin, defining the horizontal and vertical directions as the x and y axes, respectively. Then, the boundary equation for the touchpad is expressed as $x^2+y^2=R^2$, with $R=2cm$ representing the touchpad's radius. The CMC joint, MCP joint, and IP joint lie approximately along a line L (depicted as the red straight line in Fig. 2(c)), which is inclined along the negative y-axis at an angle θ.

To ensure uniform distribution of rows on the circular touchpad, we divide the segment of line L within the touchpad into N equal segments, each with a length of $\frac{R}{N}$. Simultaneously, we identify $N-1$ equal points along this line. The joint point c_i is located at a distance of r from the i^{th} equidistant point l_i on the line L, and it defines the arc arc_i with a radius of r. The coordinates of c_i can be calculated as $\left(\sin\theta\left(\frac{(N-2i)R}{N}+r\right), -\cos\theta\left(\frac{(N-2i)R}{N}+r\right)\right)$. Finally, we formulate the mathematical expressions for these arcs as follows:

$$\begin{aligned} arc_i : &\left(x-\sin\theta\left(\frac{(N-2i)R}{N}+r\right)\right)^2+ \\ &\left(y+\cos\theta\left(\frac{(N-2i)R}{N}+r\right)\right)^2=r^2 \end{aligned} \tag{1}$$

Where N denotes the number of rows on the keyboard, which is 5 in our experiments, the diagrams of the arcs are shown in Fig. 2(c) and Fig. 2(d).

FanPad. The arcs mentioned above are used to form the rows in FanPad. Each row in FanPad is a thumb trace. We divide each row into multiple regions and allocate each region a key according to QWERTY's order, the same as SKPad. The remaining regions are allocated with common punctuation and function keys. The layout of FanPad is shown in Figure 4(c). The dividing lines between rows are the arcs described above.

3.1.3 Customized Design

As described in Sect. 3.1.2, the parameters θ and r determine the arcs used by FanPad. People may have different sizes of hand and holding gestures, as well as their suitable θ and r. We propose a customized mechanism to better deal with individual differences and find better parameters to approach each user's most natural thumb-moving arc.

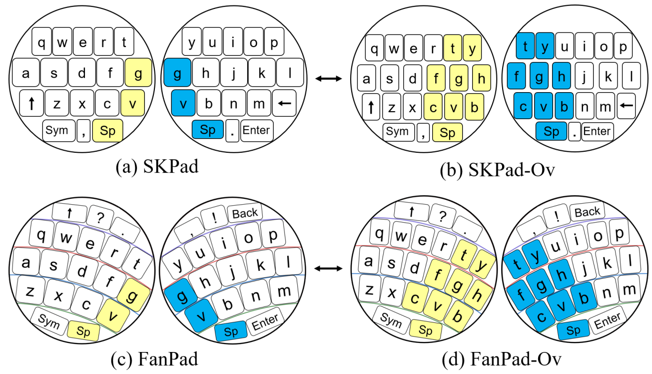

(a) SKPad (b) SKPad-Ov (c) FanPad (d) FanPad-Ov

Figure 4: Layout of SKPads and FanPads. (a) and (c) describe the standard layout, while (b) and (d) describe the layouts with customized aggressive overlaps.

The angle θ and length r can be customized anytime by a fitting process by clicking the grip button on the controller. In the process, users only need to slide their thumbs in a curve back and forth for 2 seconds naturally, without moving the other part of the hand. At the same time, 200 coordinates data on the curve is collected and used to calculate the arc's center coordinate (x,y) and radius (the thumb length r) approximately through the least squares method. So, θ can be calculated as:

$$\theta = \arctan(|y/x|) \tag{2}$$

Then, the customized θ and r will be applied in the FanPad curve.

Once the customization is complete, the users receive a notification to try out the customized key arrangements and mapping on the displayed keyboard and then exit the process by clicking the grip button again. If users are not satisfied or familiar enough, they can repeat the customizing process again.

Apart from that, users can also adjust it manually. It's worth mentioning that with customization, SKPad can be regarded as a special form of FanPad. When the user slides out a straight line during the customization process, FanPad will degenerate into the rough layout of SKPad.

3.1.4 Aggressive Overlap

In the layouts above, we divide the QWERTY keyboard into two parts from the middle and only overlap the g,v, and space keys. Based on experiment data and user surveys, we found that some users who had the typing habit, like typing "y" with their left hand, may attempt to reach the keys across the split line on the other controller pad, which caused a significant decline in efficiency and a negative experience. Building upon our prior exploration of the overlapping key design, which facilitated the engagement of both users' hands in triggering key pairs and thereby enhancing typing efficiency, we have integrated a similar approach to [7] into our keyboard design. Therefore, we modify the keyboard layout by enlarging the overlap area, which covers more letters, including t, y, f, c, h, and b, to maintain the input stability on these keys. Finally, we get four layouts: SKPad and FanPad, and their more overlapped versions, SKPad-Ov and FanPad-Ov, as shown in Fig. 4(b) and Fig. 4(d).

3.2 Typing Interaction

We use the HTC VIVE controller as the input device. The functional buttons on the controller and how we use them are shown in Fig. 5

3.2.1 Typing on Touchpad

Due to the perception loss of the thumbs' specific position in the VR environment, it's difficult to accurately locate the keys as in other scenarios such as smartphones.

As such, our approach enables users to modify the click position by dragging the thumb on the touchpad to the right point if the initial position is incorrect.

Our methods take the time when the thumb detaches from the touchpad as input timing, and the detachment position is truly used to calculate which character should be input. Once the user touches the touchpad, the customized contact location and the key calculated to be touched are highlighted to provide visual feedback on the selected character.

Buttons	Functions
Touchpad	Enter characters in normal mode, move caret in caret moving mode
Menu button	Backspace
Trigger	Click to confirm input, long-press to switch to caret moving mode
Grip button	Start parameters customization process

Figure 5: The functional buttons on HTC Vive controller and their usage

3.2.2 Word Correction and Completion

We use a word correction and completion mechanism implemented with SymSpell [10] to improve input efficiency and reduce the error rate. When a word is being typed, it records the typed part of the word and calculates words that are top 5 likely to be input to display in a candidate bar. We design the candidate bar just below the input box while the keyboard stays in a fixed place in the user's vision area, and the user can adjust its height manually in the scene to ensure all parts stay inside peripheral vision [27]. Moreover, we use the horizontal component of the forward direction of the HMD to determine the head direction to select the word in the candidate bar. The selected one is highlighted, and click the trigger to confirm entry. The letter sequence that has been input will be replaced, and a space will be automatically added.

3.2.3 Other Functions

To improve the utility of our methods, we have added other functions beyond inputting letters as follows.

Numbers and Symbols. Our keyboards contain numbers 0∼9 and most symbols in a standard QWERTY keyboard. There are two ways to input them. One is to click the *Sym* key on the keyboard, and the character on each key will switch to the corresponding number or symbol. Then click *Sym* again to switch back. The other is when a key is held down past a custom long-press duration (we set 1s in the context), a selection box with three new keys will pop up above it, including the uppercase letter, the corresponding number or symbol, and the lowercase letter from left to right. Then, slide left/right/directly release the thumb to input the uppercase/lowercase letter/corresponding number or symbol.

For the keys on the edge of the touchpad, we make adjustments to avoid situations where there is no place to move left or right. For example, a key on the left edge will decide what to input by how much the user slides right.

The adjustments will be conveyed to the user by highlighting keys in the selection box. Apart from the above two ways, some commonly used punctuation marks like commas and periods are assigned with independent keys.

Enter, Shift, Backspace. These special functions also have independent keys on our keyboard. The shift is used for switching the uppercase mode. Backspace can also be typed through the menu button on the controller. A push on the menu button types a backspace.

Caret Moving. We provide a caret position controlling mechanism in our methods. Holding down the trigger enters the caret moving mode. Typing is stopped in this mode, and touching the left/right part of the touchpad moves the caret left/right. Releasing the trigger quits the caret moving mode.

3.3 Generalizability

Generally speaking, our method is designed for controllers equipped with circular touchpads, allowing for direct compatibility with devices featuring such touchpads. However, many controllers incorporate touchpad designs that deviate from circular shapes (such as the Meta Quest Touch Pro). Our method cannot be directly applied in these cases as these touchpads might possess different usage characteristics. For such input devices, some additional adaptation is required to ensure the proper functionality of our method.

Fundamentally, our FanPad layout is designed around the arc-shaped motion of the thumb around the CMC joint during touch input. Consequently, for other touch input devices where finger touch movements follow an arc rotation, the FanPad can be readapted.

Moreover, the size of the touchpad is a crucial factor influencing the performance of our FanPad method. Within this study, we recommend touchpad radii to fall within the range of 1.5cm to 3.5cm for direct application of our method. Touchpads that are too small might result in significant input ambiguity, necessitating additional methods to resolve ambiguities. On the other hand, oversized touchpads may cause discomfort for users when frequently reaching the device's far edges, requiring prior calibration to confine input within the recommended size range.

4 User Study 1: Layout Comparison

We conducted the first user study to evaluate whether FanPad brings a better experience than SKPad. Besides, the performance and differences between the two overlapping areas are also explored. The evaluation metrics include text entry speed, error rate, and workload.

4.1 Hypotheses

We formulate three hypotheses for the experiment:

H.1. FanPad may have a higher text entry speed than SKPad since the customized curves allow users' fingerprints to move in a more natural and efficient manner.

H.2. Keyboards with more overlap areas may exhibit a higher error rate due to their smaller touch area for each key, which increases the likelihood of typing incorrect letters.

H.3. Keyboards with more overlap area may improve text entry speed since the probability of the target keys being exclusively located on another touchpad could be reduced.

4.2 Participants and Apparatus

We recruited 16 participants (eight males and eight females) between the ages of 20 to 30 from our university with engineering backgrounds. The average of their ages is 22.4375, and the standard variance of their ages is 1.8711. No one is a native English speaker or uses English as a working language, but all have English learning experience. Eleven participants had some experience with VR, and none had ever participated in any prior VR-related text entry experiments. All are familiar with the T26 software keyboard on smartphones.

The experiment was conducted on a computer with the Intel Core 11th i7 processor and the NVIDIA GTX 3080 graphics card, and the software was implemented with C# in Unity 2021.3.8.f1C1. Users typed with HTC VIVE controllers in the virtual environments displayed in the HTC VIVE.

4.3 Study Design and Procedure

The experiment used a within-subjects design with two independent variables: two different layout structures (SKPads and FanPads) and two different overlap areas (more overlap or not). All participants tested all four layouts in the order of Latin square design to reduce the impact of the order and proficiency. The tests require participants to type phrases displayed above the input field. All phrases are randomly extracted from the MacKenzie Phrase Set. [30].

Before conducting the study, we conducted a short training on basic VR skills for participants without VR experience. All participants were briefed on the functions of all buttons, input operations, and features associated with each layout. Before starting the formal tests, each participant practiced five phrases on each layout. Then, during the formal test, participants tested five phrases on each layout. We encouraged participants to use the customization function to obtain the fittest parameters for FanPad and FanPad-Ov layouts.

During the tests, all the operations performed by the participants were collected for further analysis, including the input sequence, the time stamps, and the customized parameters θ and r. The data of 16 (participants) × 4 (sessions) × 5 (phrases) = 320 phrases are collected.

After testing each layout, each participant was asked to fill out the NASA-TLX form on a scale of 0-7, and when finishing all layout tests, his/her preference for each layout was also collected.

According to Mackenzie [29], we calculated the text entry speed for WPM as follows:

$$WPM = \frac{|T|-1}{S} \times 60 \times \frac{1}{5} \tag{3}$$

where T is the target string, and S is the elapsed time in seconds from the first to the last pressed in the phrase. We also calculated the total error rate (TER) [40], including the not corrected error rate (NCER) and corrected error rate (CER).

4.4 Result

We analyzed the data using the Repeatable Two Factor Analysis ANOVA in this study. A significance threshold of 0.05 was used for P values to control the family-wise error rate.

4.4.1 Typing Performance

Entry speed. Fig. 6 compares the result of the text entry speed of novices between the four layouts in terms of WPM. The mean WPM are 9.76, 10.48, 11.37, and 11.40 for SKPad, SKPad-Ov, FanPad, and FanPad-Ov, respectively. The ANOVA result shows there is a significant difference between the two layout structures, SKPads and FanPads, with $F_{1,30} = 33.331, p < 0.001, \eta_p^2 = 0.526$, but no significant difference between the two overlap areas ($F_{1,30} = 0.427, p = 0.519, \eta_p^2 = 0.014$) in entry speed and in the cross-factor results

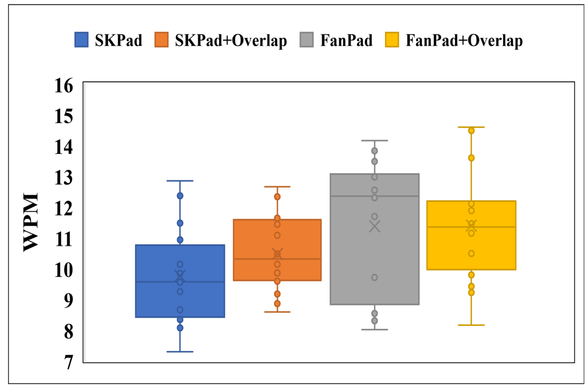


Figure 6: Mean typing speed of the four layouts. The three horizontal lines for each box indicate the maximum, median, and minimum (from top to bottom). The top and bottom edges of each box are the third and first quartiles, and the fork marker shows the mean.

($F_{1,30} = 2.499, p = 0.124, \eta_p^2 = 0.077$), i.e. the layout structures and the overlap areas.

Error rate. Fig. 7 illustrates the error rate results for each layout. The average TER are as follows:10.17% for SKPad, 8.20% for SKPad-Ov, 7.16% for FanPad, and 7.56% for FanPad-Ov. The average NCER is 0.40% for SKPad, 1.26% for SKPad-Ov, 0.91% for FanPad, and 0.80% for FanPad-Ov. It's worth noting that the different layout structures do not lead to a significant variance in TER ($F_{1,30} = 1.909, p = 0.177, \eta_p^2 = 0.06$) or NCER ($F_{1,30} = 0.005, p = 0.945, \eta_p^2 < 0.001$), nor in overlap areas ($F_{1,30} = 0.245, p = 0.624, \eta_p^2 = 0.008$), ($F_{1,30} = 2.012, p = 0.166, \eta_p^2 = 0.063$), and interaction term ($F = 0.831, p = 0.369, \eta_p^2 = 0.027$), ($F_{1,30} = 2.139, p = 0.154, \eta_p^2 = 0.067$) concerning TER and NCER.

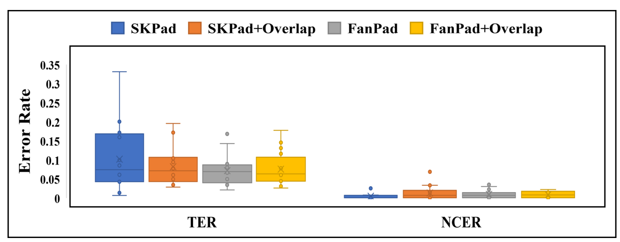
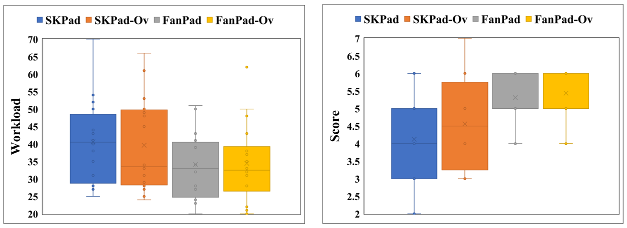


Figure 7: Error rate of the four layouts. Refer to Fig. 6 for the meanings of the horizontal lines, box edges, and fork markers.

4.4.2 Workload and Preference

We calculated the weighted average score for each NASA-TLX questionnaire, and the results are shown in Fig. 8(a). The average NASA-TLX scores for the four layouts are 40.6(SKPad), 39.6(SKPad-Ov), 34.1(FanPad), and 34.4(FanPad-Ov). Post-hoc pairwise comparisons using the Wilcoxon signed-rank test show no significant differences between any layout pairs.

Fig. 8(b) shows participants' subjective preference scores for each layout. The average scores are 4.1, 4.3, 5.3, and 5.4 for SKPad, SKPad-Ov, FanPad, and FanPad-Ov, respectively. Post-hoc pairwise comparisons using the Wilcoxon signed-rank test show significant differences between FanPad and SKPad ($Z = -2.47, p = 0.014$), FanPad-Ov and SKPad ($Z = -2.745, p = 0.006$), and FanPad-Ov and SKPad-Ov ($Z = -2.204, p = 0.027$). For other layout pairs, the Wilcoxon signed-rank test shows no significant differences.

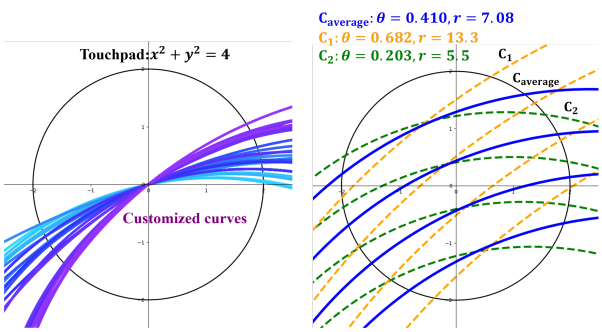


(a) NASA-TLX result (b) Preference result

Figure 8: NASA-TLX result and Participants' preference scores for the four layouts. Refer to Fig. 6 for the meanings of the horizontal lines, box edges, and fork markers.

4.4.3 Customized Curve Parameters

We collected the customized parameters θ, r for each test (both FanPad and FanPad-Ov) and visualized them as curves they derive. Fig. 9(a) shows all the curves derived from the collected parameters, while C_1, C_2, and C_3 in Fig. 9(b) shows the maximum, minimum, and average θ and r accordingly. The average length of thumbs is 7.08cm, and the inclination angle is 0.410rad, aligning well with our curving model in Sect. 3.1.2.



(a) (b)

Figure 9: (a) Visualization of all customized curves for FanPad and FanPad-Ov. (b) Boundary conditions and average curves with individual parameters.

4.5 Discussion

Basic conclusion. First of all, the results support H.1 but disagree with H.2 and H.3. FanPads significantly outperform SKPads in entry speed, while more or less overlap area does not make a difference. In addition, no significant difference is found between the four layouts in terms of error rate. However, keyboards with different overlap areas may not significantly differ in average text entry speed but could vary between users: 3 of the 16 participants were 10% faster with FanPad than with FanPad-Ov, and 5 were the opposite. The increased overlapped version is specifically designed to cater to users who have the habit of typing across the middle line. For the target users, although it may result in more time spent correcting errors, the optimization of typing keys in the overlap area may outweigh the reduction in text entry speed. However, for other users, it could have a detrimental impact. It also corresponds to the results of workload and preference. Thus, the more overlapped version should be regarded as a customized option for users to choose according to their typing habits, performance, and preferences.

Secondly, arcs generated from customized parameters can represent the typical natural thumb sliding trajectory for the majority of users, aligning with the general grip posture of the controller as shown in Fig. 2(a). However, it's possible to encounter extreme parameters due to unique controller grip postures, particularly among VR novices using the controller for the first time. For example, an extreme parameter $r = 13.3$ (the orange curves in Fig. 9(b)), occurs in a novice. However, that participant still reaches 10.49 WPM, meaning that the customized FanPads have good adaptability to various grip postures beyond our preset.

Workload and preference. The subjective questionnaire evaluations regarding workloads suggest no significant disparities among various keyboard layouts. This outcome likely stems from the shared operational similarities between FanPad and SKPad. Both utilize touchpad-based methodologies that impose minimal workload, primarily necessitating thumb movement on the touchpad. This streamlined process consolidates all actions into a single step, eliminating the need for multiple selections or sub-operations.

Regarding user preference, the evident favoritism towards FanPad validates the efficacy of our design. The integration of customized curves elevates the user experience, rendering thumb movements more natural and facilitating precise positioning and transitions to the correct location. This enhancement significantly bolsters overall usability, benefiting a wider range of users.

Limitation. Currently, we offer two different versions of our method, FanPad, and FanPad-Ov, catering to users who employ cross-midline typing habits by increasing the overlapping area. However, providing only two options with different overlapping keys lacks flexibility, as diverse typing habits among users might necessitate varying overlapping key configurations. Results from User Study 1 indicate that the current overlapping key settings are effective for only a subset of users in terms of enhancing text input efficiency. The current design of user studies isn't sufficient to draw more generalized conclusions; further exploration allowing users to customize specific overlapping area characters is needed. This exploration would accommodate diverse typing habits and investigate the impact of overlapping key positions on typing efficiency. Presently, the user studies have not addressed such customization, marking one of the limitations of this research.

4.6 Comparison with state-of-the-art methods

Furthermore, we compare the typing performance of FanPads with the state-of-the-art (SOTA) methods, HiPad [17], Climbing Keyboard [15] (abbreviated as Climbing hereinafter) and Flower [25] in Table 1, and further analyze the workload and personal preference. The experimental data for the other three methods compared were sourced directly from the original data presented for those three methods in the paper.

Before comparing, it should be clarified that FanPad and SKPad use a simple word completion method called Symspell. Among the methods we compared, Climbing stated in the paper that it used word selection with the same dissertation engine in both methods. Meanwhile, Flower also utilized techniques for word completion and correction. HiPad is similar to nine-key typing and is based on prediction methods. All the methods compared utilize word correction and prediction functionalities.

Entry speed. The participants who had never used this kind of entry method before reached 11.37 WPM in FanPad and 11.40 WPM in FanPad-Ov on average after only ten phrases of short training(including five phrases for training and five phrases for testing), and the fastest reached 14.15 WPM in FanPad and 14.59 WPM in FanPad-Ov. HiPad reaches an average of 9.14 WPM for novices after ten phrases of training. That data of Climbing is 11.21 WPM. Flower reaches an average of 8.96 WPM for novices on the first day of use after 12-14 phrases of training. Note that HiPad is based on the touchpad while Flower is based on the controller's movement, and both are one-handed methods while Climbing is based on touchpad but both-handed. FanPads outperform HiPad and Flower and is roughly equal to Climbing.

Error rate. The average of TER is 7.16% for FanPad and 7.56% for FanPad-Ov. The average NCER is 0.91% for FanPad and 0.80% for FanPad-Ov, while the average TER for novices of HiPad, Flower, and Climbing are 4.42%, 2.9%, and 14.61% and the NCER are 0.10%, 0.25%, and 3.1% respectively. FanPads have a higher error rate than HiPad and Flower, which is partly because the space for each key in FanPads is smaller than in HiPad and Flower. FanPad has 19 keys in a touchpad, FanPad-Ov has 22, HiPad has 7, and Flower has a much bigger space that requires the hand to move a lot. Though FanPad's error rates are higher, modifying an error is simpler and faster, which ensures efficiency. Overlap doesn't significantly affect the error rate, though more overlap means denser keys in a touchpad. It indicates that the current density of keys is still in a reasonable range.

Table 1: Novice performance comparison with Flower, HiPad, and Climbing keyboard

Technique	VR experience	WPM	TER	NCER
FanPad	11/16	11.37	7.16%	0.91%
FanPad-Ov	11/16	11.40	7.56%	0.80%
Flower	10/10	8.96	4.42%	0.10%
HiPad	8/10	9.14	2.9%	0.25%
Climbing	-/10	11.21	14.61%	3.1%

5 User Study 2: Performance and improvement evaluation in long time use

After finishing the first study, we conclude that the typing performances on FanPads outperform SKPads, and users have different preferences towards different overlap areas, making it a customization option. Thus, we had 8 participants in the previous to conduct an additional six-day experiment on FanPads with their customized overlap area. The goal of the experiment is to evaluate the learning cost of FanPads. We were also interested in the improvement curve and the best performance after a period of practice.

5.1 Participants and Apparatus

The average age of the remaining 8 participants is 22.625, and the standard variance is 1.7344. Among them, four people have VR experience. Their English proficiency and familiarity with the 26 keyboards have been reported in User Study 1.

5.2 Study Design and Procedure

Firstly, each participant chooses one of the two overlap areas according to their previous performance and preference. Among 8 participants, 3 preferred FanPad, while others preferred FanPad-Ov. Then, they began the six-day experiment consisting of five phrases of practice and five phrases of test every day. The data of 6 (days) × 8 (participants) × 5 (phrases) = 240 phrases are collected. Particularly, we set a three-day gap between the two user studies to mitigate the potential training effect of user study 1.

5.3 Result

5.3.1 Text Entry Speed

The average entry speed is 12.45, 14.05, 15.30, 16.76, 18.30, and 19.73 WPM in 6 days, with an increase of 58.47%. Fig. 10 shows each participant's entry speed and increment in each day. The highest typing speed among these data reached an impressive 24.19 WPM.

5.3.2 Error Rate

For TER, the average data are 7.32%, 8.15%, 8.79%, 7.97%, 7.26%, and 5.56% during six days.

For NCER, the average data are 0.43%, 1.78%, 0.42%, 1.40%, 0.76%, and 0.77% during six days.

5.4 Discussion

We also compare the typing performance and improvement of FanPads with the SOTA methods Flower [25] for the same six-day experiment. The results are summarized in Table 2.

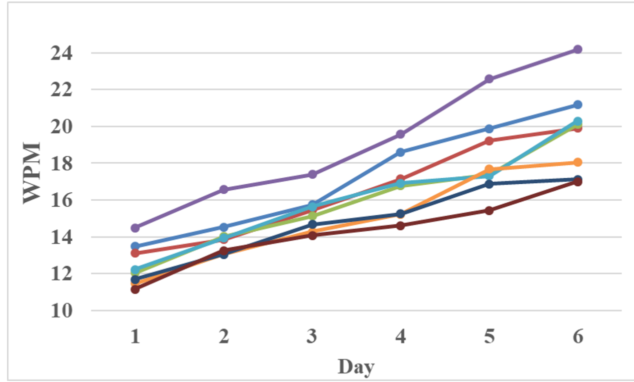


Figure 10: WPM of FanPads across six days from every participant

Entry speed. After training for 60 phrases in customized FanPads, the average typing speed rises 58.47% from 12.45 WPM to 19.73 WPM, while the novice of Flower reaches 17.65 WPM on average, and the rising rate is 96.99% after six training days with 12-14 phrases daily. Moreover, the average typing speed on FanPads outperforms the 5 of 10 experts with more training on Flower. The novice of HiPad reaches 13.57 WPM on average, and after 60 phrases of training, topped out at 18.72 WPM. Climbing reaches an average of 16.48 WPM after 50 phrases of training. In all, the speed FanPads reaches is outstanding among HiPad, Flower, and Climbing.

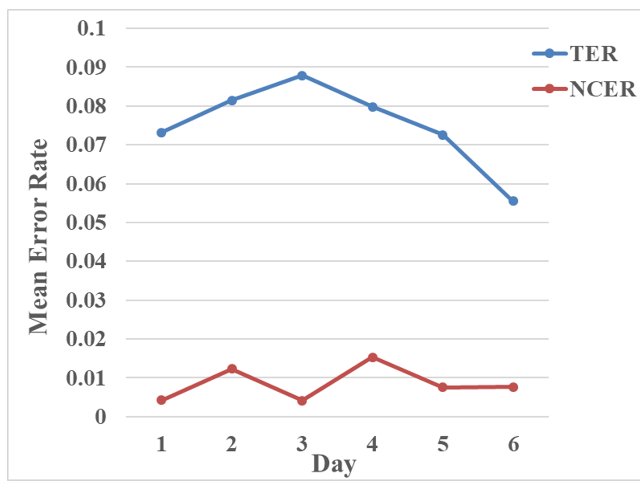


Figure 11: Average TER and NCER of FanPads across six days

However, the increase rate is relatively lower than Flower. It is probably because the fixed time of making sure that the thumb is at the right point becomes the bottleneck. Users sometimes need to pause briefly to check whether the right key is clicked at the first touch.

Error rate. In terms of error rate, no significant trend is observed over the six days because after having a certain level of proficiency, the main factor affecting the error rate becomes personality, strategy, and spelling level. During the six-day test, the average TER and NCER are 7.51% and 0.86% for FanPads, compared with 2.50% and 0.09% for Flower. FanPads have a higher error rate than Flower, the same as user study 1. Further optimizing the word correction mechanism may reduce the error rate.

Table 2: Comparison with Flower, HiPad, and Climbing keyboard on the average WPM on the last day of the user study. For Flower and FanPad, the WPM increment between the first day and the last day, and the average TER and NCER during all six days

Technique	WPM	WPM increment	TER	NCER
FanPad	19.73	58.47%	7.51%	0.86%
Flower	17.65	96.99%	2.50%	0.09%
HiPad	13.57	-	-	-
Climbing	16.48	-	-	-

6 Conclusion, Limitation, and Future Work

We introduce the FanPad, a touchpad-based text entry technique with a fan layout in HMDs. It's designed by splitting and mapping the T26 keyboard onto a curved structure on touchpads corresponding to the natural movement of the thumb when interacting with the touchpad. In addition, we provide customized curves and overlap areas to make the FanPads better fit different hand sizes and poses. To validate and assess the performance of the FanPad, we designed and executed two comprehensive user studies. In the first user study, the results demonstrated a significant increase in efficiency compared to the conventional no-curving keyboard layout (SKPad). After a six-day training and testing (user study 2), the FanPad technique reaches 19.73 WPM on average, outperforming the SOTA methods, HiPad (13.57 WPM), Climbing (16.48 WPM), and Flower (17.65 WPM). Notably, the highest typing speed reaches an impressive 24.19 WPM.

Our FanPad has demonstrated exceptional performance in comparison to other touchpad-based text entry methods. Nevertheless, it is important to acknowledge its inherent limitations. Firstly, the densely arranged keys on the touchpads may have an impact on input accuracy, potentially leading to an increased error rate. Secondly, it's worth noting that FanPads currently require the presence of a physical touchpad on the controller to capture the thumb-touching position effectively. As a result, it is not compatible with controllers featuring joysticks for now.

In the future, we plan to enhance this method through several key strategies. Firstly, we intend to create a new keyboard layout with a reduced number of keys to mitigate typing errors. One potential approach involves clustering, which groups adjacent keys together for improved accuracy. Besides, improving the correction algorithm or adding a dynamic border algorithm for touching around the keys' junctions may also reduce typing errors. Secondly, we aim to transition from a physical touchpad to a virtual one by utilizing the visual function on the HMDs to monitor the thumb's position and angle to map the input key for typing on the controller without the touchpad, or simply wave the controller to simulate the thumb-touching movement on a virtual touchpad. These expansions will make the method compatible with a wide range of input devices.

Additionally, we conducted comparisons between our method and other state-of-the-art methods, employing their original performance data. Given that their efficacy is contingent on varying participants and potential populations, the comparison outcomes could be biased. Subsequent endeavors could involve comprehensive user studies to delve into the intrinsic disparities among these methods.

Upon a user's touch on the touchpad, our method highlights the customized contact position and the calculated key to touch, offering visual feedback regarding the selected key. Some of the latest head-mounted displays provide built-in gesture-tracking modules, enabling users to visualize finger positions once sensors detect hand movements (e.g., in the form of virtual hands). Future work could integrate such visual representations into our method, enhancing the realism of interaction, augmenting user perception of fingers and touchpads, and potentially benefiting the method's performance.

Acknowledgments

This work is supported by the National Natural Science Foundation of China through Project 61932003, 62372026, by Beijing Science and Technology Plan Project Z221100007722004, and by National Key R&D plan 2019YFC1521102.

References

[1] J. Adhikary and K. Vertanen. Text Entry in Virtual Environments using Speech and a Midair Keyboard. *IEEE Transactions on Visualization and Computer Graphics*, 27(5):2648–2658, May 2021. doi: 10.1109/TVCG.2021.3067776

[2] M. A. Bakar, Y.-T. Tsai, H.-H. Hsueh, and E. C. Li. CrowbarLimbs: A Fatigue-Reducing Virtual Reality Text Entry Metaphor. *IEEE Transactions on Visualization and Computer Graphics*, 29(5):2806–2815, May 2023. Conference Name: IEEE Transactions on Visualization and Computer Graphics. doi: 10.1109/TVCG.2023.3247060

[3] J. Bergstrom-Lehtovirta and A. Oulasvirta. Modeling the functional area of the thumb on mobile touchscreen surfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, p. 1991–2000. Association for Computing Machinery, New York, NY, USA, 2014. doi: 10.1145/2556288.2557354

[4] C. Boletsis and S. Kongsvik. Text Input in Virtual Reality: A Preliminary Evaluation of the Drum-Like VR Keyboard. *Technologies*, 7:31, Apr. 2019. doi: 10.3390/technologies7020031

[5] 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*, vol. 46, pp. 2154–2158. SAGE Publications Sage CA: Los Angeles, CA, 2002.

[6] W. Cui, R. Liu, Z. Li, Y. Wang, A. Wang, X. Zhao, S. Rashidian, F. Baig, I. Ramakrishnan, F. Wang, and X. Bi. GlanceWriter: Writing Text by Glancing Over Letters with Gaze. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pp. 1–13. ACM, Hamburg Germany, Apr. 2023. doi: 10.1145/3544548.3581269

[7] V. Dhakal, A. M. Feit, P. O. Kristensson, and A. Oulasvirta. Observations on typing from 136 million keystrokes. In *Proceedings of the 2018 CHI conference on human factors in computing systems*, pp. 1–12, 2018.

[8] T. J. Dube and A. S. Arif. Text Entry in Virtual Reality: A Comprehensive Review of the Literature. In M. Kurosu, ed., *Human-Computer Interaction. Recognition and Interaction Technologies*, vol. 11567, pp. 419–437. Springer International Publishing, Cham, 2019. Series Title: Lecture Notes in Computer Science. doi: 10.1007/978-3-030-22643-5_33

[9] T. J. Dube, K. Johnson, and A. S. Arif. Shapeshifter: Gesture Typing in Virtual Reality with a Force-based Digital Thimble. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts*, pp. 1–9. ACM, New Orleans LA USA, Apr. 2022. doi: 10.1145/3491101.3519679

[10] W. Garbe. (2012) symspell [computer software], https://github.com/wolfgarbe/SymSpell.

[11] Z. Gu, X. Xu, C. Chu, and Y. Zhang. To Write not Select, a New Text Entry Method Using Joystick. In M. Kurosu, ed., *Human-Computer Interaction: Interaction Technologies*, Lecture Notes in Computer Science, pp. 35–43. Springer International Publishing, Cham, 2015. doi: 10.1007/978-3-319-20916-6_4

[12] 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*, pp. 49–60. ACM, Tokyo Japan, Oct. 2016. doi: 10.1145/2984511.2984576

[13] A. Gupta, C. Ji, H.-S. Yeo, A. Quigley, and D. Vogel. RotoSwype: Word-Gesture Typing using a Ring. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pp. 1–12. ACM, Glasgow Scotland Uk, May 2019. doi: 10.1145/3290605.3300244

[14] Z. He, C. Lutteroth, and K. Perlin. TapGazer: Text Entry with Finger Tapping and Gaze-directed Word Selection. In *CHI Conference on Human Factors in Computing Systems*, pp. 1–16. ACM, New Orleans LA USA, Apr. 2022. doi: 10.1145/3491102.3501838

[15] J. Huang, M. Sun, J. Qin, B. Gao, and G. Qin. Climbing keyboard: A tilt-based selection keyboard entry for virtual reality. *International Journal of Human–Computer Interaction*, 2022.

[16] B. Jackson, L. B. Caraco, and Z. M. Spilka. Arc-type and tilt-type: Pen-based immersive text input for room-scale vr. In *Proceedings of the 2020 ACM Symposium on Spatial User Interaction*, SUI '20. Association for Computing Machinery, New York, NY, USA, 2020. doi: 10.1145/3385959.3418454

[17] H. Jiang and D. Weng. HiPad: Text entry for Head-Mounted Displays Using Circular Touchpad. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 692–703. IEEE, Atlanta, GA, USA, Mar. 2020. doi: 10.1109/VR46266.2020.00092

[18] H. Jiang, D. Weng, Z. Zhang, Y. Bao, Y. Jia, and M. Nie. HiKeyb: High-Efficiency Mixed Reality System for Text Entry. In *2018 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, pp. 132–137. IEEE, Munich, Germany, Oct. 2018. doi: 10.1109/ISMAR-Adjunct.2018.00051

[19] H. Jiang, D. Weng, Z. Zhang, and F. Chen. HiFinger: One-Handed Text Entry Technique for Virtual Environments Based on Touches between Fingers. *Sensors*, 19(14):3063, July 2019. doi: 10.3390/s19143063

[20] F. Kern, F. Niebling, and M. E. Latoschik. Text Input for Non-Stationary XR Workspaces: Investigating Tap and Word-Gesture Keyboards in Virtual and Augmented Reality. *IEEE Transactions on Visualization and Computer Graphics*, 29(5):2658–2669, May 2023. Conference Name: IEEE Transactions on Visualization and Computer Graphics. doi: 10.1109/TVCG.2023.3247098

[21] J. Kim, W. Delamare, and P. Irani. ThumbText: Text Entry for Wearable Devices Using a Miniature Ring. In *Graphics Interface*. Toronto, Canada, May 2018.

[22] Y. R. Kim and G. J. Kim. HoVR-Type: Smartphone as a typing interface in VR using hovering. In *2017 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 200–203. IEEE, Las Vegas, NV, 2017. doi: 10.1109/ICCE.2017.7889285

[23] N. Kimura, T. Gemicioglu, J. Womack, R. Li, Y. Zhao, A. Bedri, Z. Su, A. Olwal, J. Rekimoto, and T. Starner. Silentspeller: Towards mobile, hands-free, silent speech text entry using electropalatography. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pp. 1–19, 2022.

[24] C. Kumar, R. Hedeshy, I. S. MacKenzie, and S. Staab. TAGSwipe: Touch Assisted Gaze Swipe for Text Entry. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–12. ACM, Honolulu HI USA, Apr. 2020. doi: 10.1145/3313831.3376317

[25] J. Leng, L. Wang, X. Liu, X. Shi, and M. Wang. Efficient flower text entry in virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 28(11):3662–3672, 2022.

[26] Y. Li, S. Sarcar, Y. Zheng, and X. Ren. Exploring Text Revision with Backspace and Caret in Virtual Reality. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–12. ACM, Yokohama Japan, May 2021. doi: 10.1145/3411764.3445474

[27] Y. Lu, C. Yu, S. Fan, X. Bi, and Y. Shi. Typing on split keyboards with peripheral vision. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, p. 1–12. Association for Computing Machinery, New York, NY, USA, 2019. doi: 10.1145/3290605.3300430

[28] 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. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(2):1–24, June 2017. doi: 10.1145/3090083

[29] I. S. MacKenzie. A note on calculating text entry speed. *Unpublished work. Available online at http://www. yorku. ca/mack/RN-TextEntrySpeed. html*, 2002.

[30] 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*, CHI EA '03, p. 754–755. Association for Computing Machinery, New York, NY, USA, 2003. doi: 10.1145/765891.765971

[31] F. Matulic, A. Ganeshan, H. Fujiwara, and D. Vogel. Phonetroller: Visual Representations of Fingers for Precise Touch Input with Mobile Phones in VR. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–13. ACM, Yokohama Japan, May 2021. doi: 10.1145/3411764.3445583

[32] M. McGill, D. Boland, R. Murray-Smith, and S. Brewster. A Dose of Reality: Overcoming Usability Challenges in VR Head-Mounted Displays. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 2143–2152. ACM, Seoul Republic of Korea, Apr. 2015. doi: 10.1145/2702123.2702382

[33] A. Nguyen, S. Bittman, and M. Zank. Text input methods in virtual reality using radial layouts. In *Proceedings of the 26th ACM Symposium on Virtual Reality Software and Technology*, VRST '20. Association for Computing Machinery, New York, NY, USA, 2020. doi: 10.1145/3385956.3422114

[34] D.-M. Pham and W. Stuerzlinger. HawKEY: Efficient and Versatile Text Entry for Virtual Reality. In *25th ACM Symposium on Virtual Reality Software and Technology*, pp. 1–11. ACM, Parramatta NSW Australia, Nov. 2019. doi: 10.1145/3359996.3364265

[35] 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 *2016 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 109–112. IEEE, Greenville, SC, USA, Mar. 2016. doi: 10.1109/3DUI.2016.7460039

[36] M. Pratorius, U. Burgbacher, D. Valkov, and K. Hinrichs. Sensing Thumb-to-Finger Taps for Symbolic Input in VR/AR Environments. *IEEE Computer Graphics and Applications*, 35(5):42–54, Sept. 2015. doi: 10.1109/MCG.2015.106

[37] R. Qin, S. Zhu, Y.-H. Lin, Y.-J. Ko, and X. Bi. Optimal-T9: An Optimized T9-like Keyboard for Small Touchscreen Devices. In *Proceedings of the 2018 ACM International Conference on Interactive Surfaces and Spaces*, pp. 137–146. ACM, Tokyo Japan, Nov. 2018. doi: 10.1145/3279778.3279786

[38] K. Sengupta, S. Bhattarai, S. Sarcar, I. S. MacKenzie, and S. Staab. Leveraging Error Correction in Voice-based Text Entry by Talk-and-Gaze. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–11. ACM, Honolulu HI USA, Apr. 2020. doi: 10.1145/3313831.3376579

[39] Y. Singhal, R. H. Noeske, A. Bhardwaj, and J. R. Kim. Improving Finger Stroke Recognition Rate for Eyes-Free Mid-Air Typing in VR. In *CHI Conference on Human Factors in Computing Systems*, pp. 1–9. ACM, New Orleans LA USA, Apr. 2022. doi: 10.1145/3491102.3502100

[40] 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*, pp. 113–120, 2003.

[41] 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*, pp. 1–13. ACM, Montreal QC Canada, Apr. 2018. doi: 10.1145/3173574.3174221

[42] Y. Tian, H. Bai, S. Zhao, C.-W. Fu, C. Yu, H. Qin, Q. Wang, and P.-A. Heng. Kine-Appendage: Enhancing Freehand VR Interaction Through Transformations of Virtual Appendages. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–17, 2022. doi: 10.1109/TVCG.2022.3230746

[43] J. Walker, S. Kuhl, and K. Vertanen. Decoder-assisted typing using an hmd and a physical keyboard. In *CHI '16: Extended Abstracts of the the ACM Conference on Human Factors in Computing Systems*, May 2016.

[44] J. Walker, B. Li, K. Vertanen, and S. Kuhl. Efficient Typing on a Visually Occluded Physical Keyboard. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pp. 5457–5461. ACM, Denver Colorado USA, May 2017. doi: 10.1145/3025453.3025783

[45] E. Whitmire, M. Jain, D. Jain, G. Nelson, R. Karkar, S. Patel, and M. Goel. DigiTouch: Reconfigurable Thumb-to-Finger Input and Text Entry on Head-mounted Displays. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(3):1–21, Sept. 2017. doi: 10.1145/3130978

[46] P. C. Wong, K. Zhu, and H. Fu. FingerT9: Leveraging Thumb-to-finger Interaction for Same-side-hand Text Entry on Smartwatches. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pp. 1–10. ACM, Montreal QC Canada, Apr. 2018. doi: 10.1145/3173574.3173752

[47] W. Xu, H.-N. Liang, Y. Zhao, T. Zhang, D. Yu, and D. Monteiro. RingText: Dwell-free and hands-free Text Entry for Mobile Head-Mounted Displays using Head Motions. *IEEE Transactions on Visualization and Computer Graphics*, 25(5):1991–2001, May 2019. doi: 10.1109/TVCG.2019.2898736

[48] Z. Xu, W. Chen, D. Zhao, J. Luo, T.-Y. Wu, J. Gong, S. Yin, J. Zhai, and X.-D. Yang. BiTipText: Bimanual Eyes-Free Text Entry on a Fingertip Keyboard. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–13. ACM, Honolulu HI USA, Apr. 2020. doi: 10.1145/3313831.3376306

[49] Z. Xu, P. C. Wong, J. Gong, T.-Y. Wu, A. S. Nittala, X. Bi, J. Steimle, H. Fu, K. Zhu, and X.-D. Yang. TipText: Eyes-Free Text Entry on a Fingertip Keyboard. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*, pp. 883–899. ACM, New Orleans LA USA, Oct. 2019. doi: 10.1145/3332165.3347865

[50] 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*, pp. 539–548. ACM, Charlotte NC USA, Nov. 2015. doi: 10.1145/2807442.2807504

[51] 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*, pp. 4479–4488. ACM, Denver Colorado USA, May 2017. doi: 10.1145/3025453.3025964

[52] D. Yu, K. Fan, H. Zhang, D. Monteiro, W. Xu, and H.-N. Liang. PizzaText: Text Entry for Virtual Reality Systems Using Dual Thumbsticks. *IEEE Transactions on Visualization and Computer Graphics*, 24(11):2927–2935, Nov. 2018. doi: 10.1109/TVCG.2018.2868581

[53] Z. Zhang, M. Sun, B. Gao, and L. Wang. 2-thumbs typing: A novel bimanual text entry method in virtual reality environments. In *2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pp. 530–531, 2021. doi: 10.1109/VRW52623.2021.00147