LipText: Lip Tracking Based Text Entry in VR

Jiaye Leng[1], Zijun Wang[2], Jian Wu[2*], and Lili Wang[2]

[1] School of Creative Media, City University of Hong Kong
jiayeleng2-c@my.cityu.edu.hk
[2] State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing, 100191
892710638@qq.com, {lanayawj, wanglily}@buaa.edu.cn

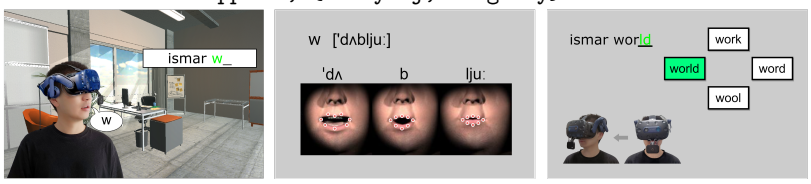


Fig. 1: When the user enters a word, he reads the letter silently (left), and the facial tracker captures the lip shape in a sequence (middle). The head motion of the user captured by VR HMDs is used for the auxiliary selection of a word from the four predicted candidates (right).

Abstract. Text entry is an important task in virtual reality (VR), and most existing methods require hand involvement, while hands-free typing has great potential for applications in mobile scenarios. Existing hands-free text entry methods are usually implemented by combining the head and eyes with techniques such as Dwell, Blink and Gesture, which can easily fatigue the user. In this paper, we propose LipText, a lip-tracking-based text entry method in VR. We use a neural network to perform letter-level prediction on the lip data captured by the facial tracker and use head-based selection as an auxiliary to improve the accuracy. We conduct a user study to evaluate our method, the results show a typing speed of 8.63 WPM for the novice group, 9.81 WPM for the potential expert group, and the highest recorded typing speed is 11.13 WPM achieved by a potential expert. Our method is also novice-friendly, and their typing speed increased by 64.38% over a six-day practice.

Keywords: Virtual reality · Text entry · Hands-free.

1 Introduction

Text entry is a common application in virtual reality (VR), where users need to communicate with each other and record information. Existing text entry methods usually require hand involvement and use devices such as physical keyboards, touch screens, sensors, and handles for input. While in mobile scenarios, these devices impose an additional burden, and the user's hands may be occupied, so it makes sense to explore hand-free text entry techniques. It is also beneficial for those users with hand motion deficits.

Several existing works have explored hands-free input. RingText [23] allows the user to use head motions to control a cursor for selection, eliminating the need for the user to hold a specialized input device to select letters. iText [12] is a technique for text entry in augmented reality (AR) systems based on an imaginary keyboard, and the keyboard area is transparent. Although both methods achieve efficient typing speeds (13.24 WPM for RingText and 13.76 WPM for iText), they both require the user to focus on the keyboard interface, which tends to make the user feel fatigued, and it is also not easy to select a target letter from the 26 letters. We think that introducing new interaction methods may provide a new solution for hands-free typing and provide a more natural interaction experience, and narrowing the range of letters to select when typing can also reduce the user's fatigue.

In this paper, we propose LipText, a lip tracking-based text entry method in VR. Firstly, we set up a facial tracker on the VR HMDs to capture the user's lip shapes. After that, the lip shape sequences with 37 feature points are obtained, which we segment and denoise to generate the lip shape features of the input letters. Secondly, we use a neural network to classify the lip shape features to recognize the input letters. Thirdly, we adopt a head motion-based auxiliary selection method to improve the correctness of the text entry. We design a user study to evaluate our LipText. The results show that the typing speed is about 9.8 WPM, and a potential expert can type at 11.13 WPM. The average NCER and TER are 2.43% and 6.66% respectively. Our method also has good learnability, for the novices, the typing speed can be raised by 64.38% through a six-day practice. Figure 1 shows the diagram of inputting the letter 'W' using our lip tracking-based text entry method.

In summary, the contributions of this paper are as follows:

- We propose a lip tracking-based text entry pipeline in VR. We introduce a new interaction method of silent reading into the hands-free text entry techniques for the first time.
- We introduce a neural network-based letter recognition method for the lip tracking data captured by the facial tracker.
- We design a user study to evaluate the performance of our LipText.

2 Related work

In this section, we review the existing text entry methods in VR and hands-free text entry techniques.

2.1 Text Entry in VR

Unlike input text in reality, in VR users need to wear HMDs and type in a virtual environment (VE), which creates visual and interactive differences to text entry. One possible solution is to integrate the physical keyboard directly to the VR system, which can achieve typing efficiency similar to that in the real world [9]. The above methods requires the user to sit down and type, while Pham et al.

[17] proposed the HawKey application for mobile scenarios, where the user wore a tray in front of him to place the keyboard on it.

Speech-based techniques are also capable of efficient typing, but due to it being difficult to correct errors [21], it is usually presented in a multimodal form. Adhikary et al. [1] combined hand tracking and speech in VR, allowing the user to speak a sentence and then correct the errors on a mid-air keyboard by avatar hands. Touch screen-based techniques introduces a touch screen, such as tablets and smartphones, to perform text entry. Gugenheimer et al. [6] mounted a multi-touch surface on the back of a VR HMD, which allowed for precise interaction and thus performed text entry in mobile scenarios. Mid-air typing techniques are also solutions for mobile scenarios, mostly requiring sensors to detect the user's typing behavior. Whitmire et al. [22] proposed DigiTouch, which enabled thumb-to-finger touch interaction for text entry through a glove with continuous touch tracking, and similar techniques were available in [10, 24].

Head-based techniques focus on the user using head movements to control the cursor for selections on a virtual keyboard. Yu et al. [25] explored three head-based text entry techniques: Tap, Dwell, and Gesture, with Gesture outperforming the other two. Xu et al. [23] proposed RingText, where the user used dwell-free technique on a circular layout with two concentric circles for input. The handle-based techniques mainly use the handle controllers provided in the existing VR systems for input. Yu et al. [26] proposed PizzaText, which chunked 26 letters and used a handle with two joysticks for two-step selection. Jiang et al. [8] proposed HiPad, which used a handle with a circular touchpad, and a circular virtual keyboard to support single-hand input.

The existing techniques in VR usually use people's hands, heads, voices, and eyes to perform text input, while the mouth has not been explored as a potential input source. Silent reading avoids to some extent the problems associated with speech-based techniques and, similar to speech techniques, it may be able to support letter-level, word-level, and sentence-level input as well.

2.2 Hands-free Text Entry Techniques

Hands-free text entry techniques are being explored due to the possible limitations of devices, or people's hands being occupied. Speech-based techniques are possible to achieve this, transcribing people's speech into text through speech recognition. Ruan et al. [18] used a deep learning-based speech recognition system and compared it to a smartphone's default keyboard, showing that transcribing text using speech was nearly three times faster than on a touchscreen keyboard and that it had a higher uncorrected error rate. Although speech-based techniques are fast to type, their effectiveness suffers in noisy environments [19], and people avoid using them in public for the security of privacy. Some techniques that utilize the head and eyes are also capable of hands-free input, Dwell [25], Dwell-free, Gesture, and Blink are common solutions. E. Mott et al. [16] proposed cascaded gaze typing, which dynamically adjusted the dwell time of keys in the on-screen keyboard based on the likelihood of the next key being selected and the position of the key on the keyboard.

Compared with the dwell techniques, the dwell-free techniques eliminate the cost of dwell time and improve the selection efficiency. RingText [23] adopted hands-free technique where the user used head movements to control the cursor to type on a circular keyboard. Typing with blink was also a dwell-free technique, and the findings of [13] showed it was superior to the dwell technique. Gesture was also a hands-free technique that used head movements to draw a path on the keyboard through letters of a word in order, it had been found to work well on both visible [25] and invisible [12] keyboards.

The above methods usually require the user to focus on the virtual keyboard (the invisible keyboard still has an input area) and control the cursor for selection, which can lead to user fatigue, and it is attractive to free the user's attention from the virtual keyboard.

3 Method

Our lip tracking-based text entry method takes the lip shape data stream captured by the facial tracker and head motion captured by VR HMDs as inputs. The output is the letter or word the user intends to enter. Our method has three main steps: lip shape sequences generation, letter recognition, and auxiliary selection. The pipeline is shown in Figure 2.

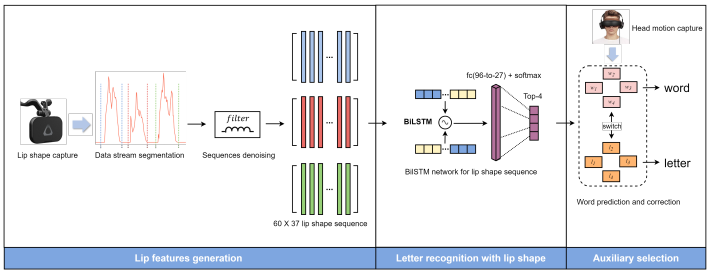


Fig. 2: The pipeline of our lip tracking-based text entry method.

3.1 Lip shape sequences generation

In this section, we capture lip shape with the facial tracker, segment the data stream into the sequences of a single letter with a normalized length, and denoise the sequences.

Lip shape feature capture We use the HTC VIVE facial tracker to capture lip shape features when users read the letters silently. It tracks 37 blend shapes related to the lip shape, including the points on the lip, jaw, teeth, tongue, chin, and cheeks. Its tracking refresh rate is 60Hz, and latency is less than 10ms. The device is also able to access the VR HMDs quickly. Therefore, our method can easily be integrated into the existing VR HMDs-based systems. Our method can achieve high accuracy and low latency lip shape tracking even in low light environments since the device uses an infrared camera. Figure 1 left shows the user wearing the VR HMD with a facial tracker mounted on the bottom of it. When the user reads a letter of words, the device outputs the 37-dimension vector stream that presents the changes of lip shapes.

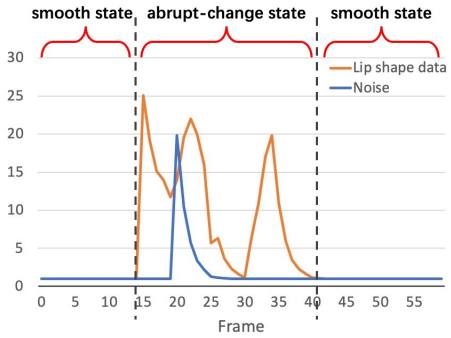


Fig. 3: The one-dimensional data in the lip shape 37-dimensional data when the user reads a is given by the orange line, which can be divided into smooth and abrupt-change states. The blue line represents the noise data.

Data stream segmentation The 37-dimensional vector stream captured is continuous and must be segmented into normalized sequences for each letter before letter recognition. The orange line in Figure 3 shows the change in one of the 37 dimensions as the user reads the letter 'a', where the lips move from closed to open and then closed again. Our approach to segmenting the continuous data stream is based on detecting abrupt-change points. The state where the lips remain closed is referred to as the "smooth state" (frames 0-14), and the point where the data sharply fluctuates from this smooth state is called the "abrupt-change point" (frame 15). We designed a segmentation algorithm 1 to capture the sequence of fluctuating data as the lip shape data for each letter, starting from the abrupt-change point until the smooth state is restored (frames 15-40, abrupt-change state). The details of the algorithm are as follows.

The algorithm inputs three consecutive frames of lip features: LF_{i-2}, LF_{i-1}, LF_i; the current state S_i, which can be "smooth state" (S) or "abrupt-change state" (AC), initialized as "S"; and the current frame count Cnt for the "AC" state. The output is the segmentation flag F_i, indicating a segmented frame. First, we initialize $SumStd$ (the sum of the standard deviations of the three frames) to 0 and F_i to False (lines 1-2). We then compute $SumStd$ as the sum of the standard deviations in each dimension across LF_{i-2}, LF_{i-1}, and LF_i (lines 3-7).

If S_i is "S" and $SumStd$ exceeds a predefined threshold $MINSTD$, this indicates an "abrupt-change point", so S_i switches to "AC" and F_i is set to True (lines 8-11). During silent reading, the lips remain active, so the standard deviation stays above the threshold until the user finishes reading and closes their mouth.

If S_i is "AC" and it ends at this frame, we reset S_i to "S" and set F_i to True (lines 12-15). The function Check determines if the "AC" state should end: if $SumStd$ falls below $MINSTD$, S_i switches to "S". Two special cases may occur: 1) Subtle lip movements may cause $SumStd$ to drop prematurely, exiting "AC" too early. 2) Fast reading may result in consecutive letters being treated as one sequence since the "S" state is not detected. To address this,

Algorithm 1: Data Stream Segmentation

Input : previous lip features LF_{i-2}, LF_{i-1}, current lip feature LF_i, current state S_i, current written frames Cnt

Output : segmentation flag F_i

$SumStd = 0$
$F_i =$ **False**
for $j = 0 \rightarrow 37$ **do**
  $m = (LF_{i-2}[j] + LF_{i-1}[j] + LF_i[j])/3$
  $std = (LF_{i-2}[j] - m)^2 + (LF_{i-1}[j] - m)^2 + (LF_i[j] - m)^2$
  $SumStd = SumStd + \sqrt{std/3}$
end for
if $(S_i == S$ && $SumStd > MINSTD)$
then
  $S_i = AC$
  $F_i =$ **True**
if $(S_i == AC$ && Check$(SumStd, Cnt) ==$ **True**$)$
then
  $S_i = S$
  $F_i =$ **True**

we set $MINACFRAME$ to 15 and $MAXACFRAME$ to 60, based on the observations of 26 letters' temporal data. The function Check returns True if Cnt exceeds $MINACFRAME$ and $SumStd$ is below $MINSTD$, or if Cnt exceeds $MAXACFRAME$. Otherwise, it returns False. This prevents exiting "AC" too early or recording noisy data. However, we advise users to ensure clear lip movements and avoid reading too quickly.

Sequences denoising Noise significantly impacts the performance of our method, as the segmentation algorithm relies on detecting "abrupt-change points" for mode switching, making it sensitive to noise. In the "S" state, even slight facial jitters can trigger an abrupt change, causing the algorithm to switch to the "AC" state and record noisy data. Since it's unrealistic to expect users to keep their lips perfectly still when not reading letters, an effective denoising algorithm is necessary. As shown in Figure 3, noise data typically exhibit low frequency and short duration, with jitter lasting less than ten frames. Based on this, our denoising algorithm calculates the sum of standard deviations for each dimension from the 10th frame onward. If this sum falls below a threshold, the data is considered noisy and discarded. We then normalize sequences shorter than 60 frames by padding them with zeros.

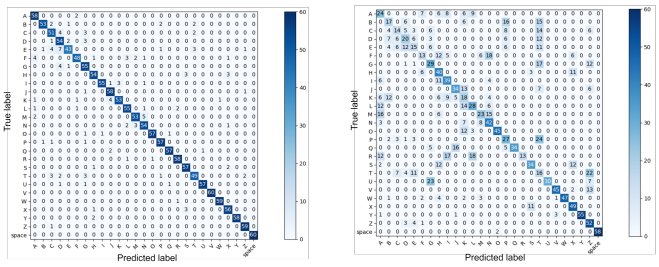
3.2 Letter recognition

We use a neural network-based method to recognize the letters according to the normalized lip shape sequences when the user reads silently.

Dataset We recruited ten participants (five males, five females, aged 22-30) from our university to collect training data. Six had prior experience with VR

headsets. Each participant completed 30 phrases, including randomly generated phrases from the Mackenzie phrase set [15] and semantic-free phrases to ensure balanced data for each letter. For the 26 letters, we used the standard lip shapes from the International Phonetic Alphabet [11]. The space key was represented by a distinct "pouting" lip shape, designed to be easily distinguishable from the letters. Participants were instructed to enter text naturally while ensuring clear and complete lip shapes. In total, we collected 300 phrases (10 participants $\times$ 30 phrases), resulting in 8100 sequences (27 characters $\times$ 300).

Neural network Lip tracking generates a 37-dimensional data sequence, requiring neural networks capable of processing multivariate time series data. We consider three options: Long Short-Term Memory (LSTM) [4], Gated Recurrent Unit (GRU) [5], and Bidirectional LSTM (BiLSTM) [2]. LSTM excels at handling time-series data by using gates (Input, Output, Forget) to manage long-term dependencies and address vanishing gradients. GRU, with two gates (Update and Reset), is simpler and has fewer parameters than LSTM. BiLSTM combines forward and backward LSTMs, allowing the model to capture bidirectional patterns in the sequence. We incorporate a Leaky ReLU activation function to increase sparsity and mitigate overfitting. A fully-connected layer follows, outputting a 1×27 vector, with 27 representing the classification categories. The input to the network is a 60×37 matrix.



(a) (b)

Fig. 4: The confusion matrices for 27 labels (26 letters + space key) on the validation (a) and test (b) sets. The horizontal axis represents the predicted labels, and the vertical axis represents the true labels. Each element of the matrices represents the number of times a given true label was predicted as different labels. The diagonal of the matrix shows the number of correct predictions for each letter, and each row sums to 60.

We used the 8100 collected lip shape sequences, split into a 3:1:1 ratio for training, validating, and testing, to evaluate the classification accuracy of the LSTM, GRU, and BiLSTM models. All three models achieved notable results, with BiLSTM, LSTM, and GRU achieving accuracies of 92.72%, 86.30%, and 78.65%, respectively. We further optimized the BiLSTM model by adjusting training parameters, including learning rate, loss function, batch size, optimizer, and hidden unit dimensions. The best BiLSTM model had a hidden size of 96.

Figure 4 (a) shows the confusion matrix for BiLSTM on the validation set, where letters like 'B', 'C', 'F', 'K', 'M', and 'T' had accuracies between 80% and 90%, with 'E' being the lowest at 71.67%, and the rest above 90%. On the test set (Figure 4 (b)), accuracy dropped for some letters except 'X', 'Y', 'Z', and 'space'. We then calculated the top-4 prediction accuracy (Figure 5), which improved compared to the top-1. Except for 'B', 'C', 'D', and 'E', the accuracy for all other letters exceeded 80%, suggesting that a multi-choice auxiliary selection method may enhance usability.

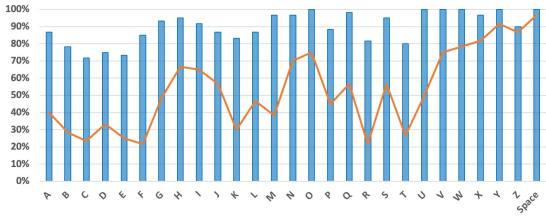


Fig. 5: The probability of the correct predicted results of 27 letters appearing in the top-4 (blue) versus the top-1 (orange).

3.3 Auxiliary selection

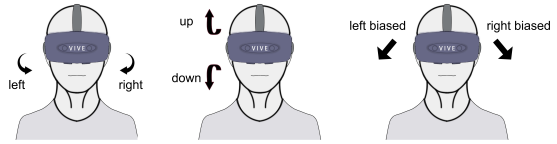


Fig. 6: Head-motion for auxiliary selection.

Head-based selection method To improve text entry accuracy, we allow users to select from the recognition results. Since our approach is hands-free, we consider integrating head-based selection methods to minimize user learning effort. To further enhance accuracy and speed, we integrate a word correction technique using SymSpell [3], combined with the head-based selection approach, following similar methods from previous works [8, 23]. Common head selection methods, such as Dwell and Dwell-free [23], often lead to misselection. Therefore, we propose a head motion-based selection method to choose between the top four predictions. As shown in Figure 6, this method detects six head movements: up, down, left, right, left bias (for Switch), and right bias (for Backspace). The first four are used to select from the top-4 predictions, while Switch toggles between letter and word selection.

4 Pilot User Study

Referring to the now popular approach of head selection, we considered adding the auxiliary selection strategies mentioned in Section 3.3 to our method. In this pilot user study, we evaluated three potential head-based selection methods: Dwell, Dwell-free, and ours. There are top-4 candidates of the prediction as well as the Switch and Backspace keys in the dwell layout.

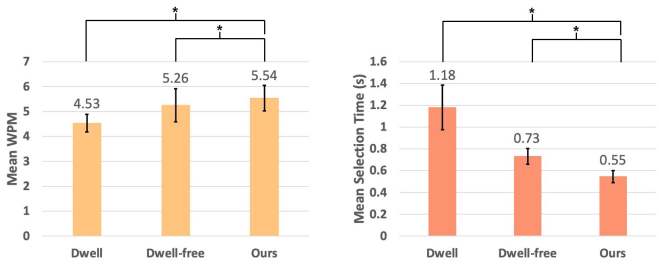
4.1 Pilot User Study Design

Participants and Hardware Setup Eighteen participants (thirteen males and five females, aged between 22-30) from our university participated in this study. We used a VIVE Pro 2 headset to provide an immersive experience and a VIVE facial tracker to capture lip shapes. Our computer configuration was the Intel Core i7 processor with an NVIDIA GeForce RTX 1060 graphics card. The system was developed with Unity 2021.2.

Task and Procedure This study used a within-subject design with one independent variable. Session 1-3 represented Dwell (CC1), Dwell-free (CC2) and our selection method (EC) respectively, where the dwell time for Dwell was set to 400ms. Each session required participants to input ten randomly generated phrases from the Mackenzie phrase set [15], and the order of the sessions was also randomly assigned. Before starting the experiment, we gave participants approximately 3 minutes to familiarize themselves with these methods. The two metrics for this experiment are typing speed and single letter selection time, where single-letter selection time is the time between the model predicting a candidate letter and the participant completing the selection. We kept correctly selected data and excluded incorrectly selected data (the target letter was not in top-4). Participants were required to fill out a NASA-TLX questionnaire [7] at the end of each session. A total of 3 (methods) $\times$ 10 (phrases) $\times$ 18 (participants) $=$ 540 phrases were collected. The Words Per Minute (WPM) was calculated following the equation in [14].

4.2 Results

We used a one-way repeated ANOVA to analyze the results of the experiment and used Bonferroni correction in pair-wise comparisons.



(a) (b)

Fig. 7: Mean typing speed (a) and selection time (b) for three head-based selection methods. Error bars indicate standard deviation. Asterisks denote statistical significance (same for all figures below).

Typing Speed & Selection Time Figure 7 shows the average typing speed and the average selection time of a single letter for the three methods, with details

Table 1: The typing speed, in seconds.

Condition	Avg ± std. dev.	(EC-CC_i) / CC_i	p	Cohen's d	Effect size
CC_1	4.53 ± 0.36	22.30%	$< 0.001^*$	2.56	Huge
CC_2	5.26 ± 0.66	5.32%	$= 0.009^*$	0.47	Small
EC	5.54 ± 0.52				

of pair-wise comparisons given in Table 1 and 2. The results of the ANOVA showed that the different methods had a significant effect on both typing speed ($F_{1.39,23.63} = 94.06$, $p < 0.001$, $\eta_p^2 = .85$) and selection time ($F_{1.01,22.24} = 312.01$, $p < 0.001$, $\eta_p^2 = .93$). For typing speed, pair-wise comparisons showed that our selection method had a significant improvement compared to Dwell ($p < 0.001$) and Dwell-free ($p = .009$), and there was also a significant difference between Dwell and Dwell-free ($p = .009$). The effect size of our selection method compared with Dwell and Dwell-free were 'Huge' and 'Small', respectively. For selection time, pair-wise comparisons showed that our selection method had a significant reduction compared to Dwell ($p < 0.001$) and Dwell-free ($p < 0.001$), and there was also a significant difference between Dwell and Dwell-free ($p < 0.001$). The effect size of our selection method compared with other two were 'Huge'.

Table 2: The selection time, in seconds.

Condition	Avg ± std. dev.	(CC_i-EC) / CC_i	p	Cohen's d	Effect size
CC_1	1.18 ± 0.20	53.39%	$< 0.001^*$	4.24	Huge
CC_2	0.73 ± 0.07	24.66%	$< 0.001^*$	2.86	Huge
EC	0.55 ± 0.05				

Workload The average scores for the six questions of the NASA-TLX questionnaire are shown in Figure 8. Over all six questions, the ANOVA results showed significantly different workloads between the three methods ($F_{2,44} = 77.95$, $p < 0.001$, $\eta_p^2 = .780$). The pair-wise comparisons on overall score showed that our selection method had less workload on the user, compared to Dwell ($p < 0.001$) and Dwell-free ($p < 0.001$), and there was also a significant difference between Dwell and Dwell-free ($p < 0.001$).

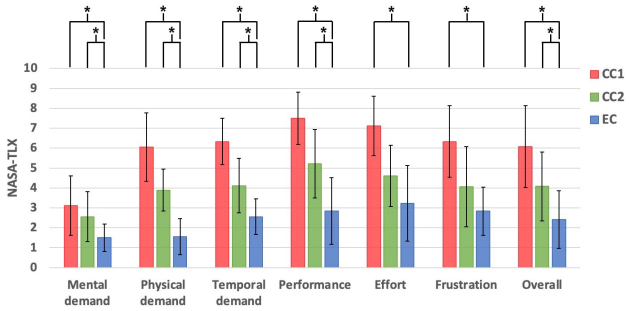


Fig. 8: Mean scores for the six individual questions and overall in NASA-TLX for three head-based selection methods (smaller value is better, from 0 to 10).

4.3 Discussion

The experimental results support our two hypotheses. Firstly, our head selection method has a faster typing speed and less selection time for a single letter. The Dwell and Dwell-free method require the user to control a cursor with the head and use the head motion to select keys on the layout. This is an unnatural way of selection because the user cannot freely deflect the head as they are used to. When using the two methods, we observed that participants moved their heads slowly at first when making a selection and then sped up as the cursor approached the target key, while this process was usually absent when using our head selection method. In addition, the Dwell method requires an additional period of dwell for selection determination, which results in a longer selection process. As reflected by the experimental results, the Dwell method had the slowest typing speed and the longest selection time.

Secondly, our head selection method has a lower workload. The Dwell and Dwell-free methods require the user to focus on a layout and use the head control a cursor to make the selection. Some participants reported that this kind of selection method made them tired quickly, while the orientation-based selection was simpler and easier, and they could make the right choice without hesitation. We decided to use our head motion-based selection method as the auxiliary selection method based on the above analysis.

5 User study

A six-day user study was conducted to evaluate the performance of LipText, including typing speed and error rates. We divided the participants into a novice group and a potential expert group to explore how their performance would improve with the increased use of LipText. The hardware setup for this user study was the same as the previous pilot user study.

5.1 User Study Design

Participants We recruited ten participants (eight males and two females, aged between 22-28) to participate in the user study, who formed a potential expert group and a novice group. The potential expert group participants were the top five performers from the previous pilot user study. The novice group was five participants who had not used our LipText, but had previous experience with VR head-mounted display devices.

Task and Procedure The entire experiment was divided into six sessions, with each user required to complete one session per day, and in each session the user was required to complete ten randomly generated phrases from the Mackenzie phrase set [15]. In each session, participants were asked to complete each session as 'quickly and accurately' as possible. In total, we collected 5 (participants) $\times$ 2 (groups) $\times$ 6 (sessions) $\times$ 10 (phrases) = 600 phrases. The error rates were calculated according to [20], Total Error Rate (TER) = Not Corrected Error Rate (NCER) + Corrected Error Rate (CER).

5.2 Results

We conducted a mixed-design ANOVA on the experimental results, where 'session' (session 1-6) was the within-subject factors and 'group' (novice group and potential expert group) was the between-subject factors.

Typing Speed We found that both 'session' ($F_{1.81,14.44} = 202.26$, $p < 0.001$, $\eta_p^2 = .96$) and 'session' × 'group' ($F_{1.81,14.44} = 4.870$, $p = .027$, $\eta_p^2 = .38$) had a significant effect on typing speed. This indicated that after a period of practice, participants in both groups experienced a significant increase in typing speed. And 'group' ($F_{1,8} = 24.71$, $p = .001$, $\eta_p^2 = .76$) was also found to have a significant effect on typing speed. In the pair-wise comparisons, significant differences were found between all session pairs (all $p < .01$) except for pair $3vs4$. These results indicated that there was still an upward trend in the typing speed of the participants after six days of practice.

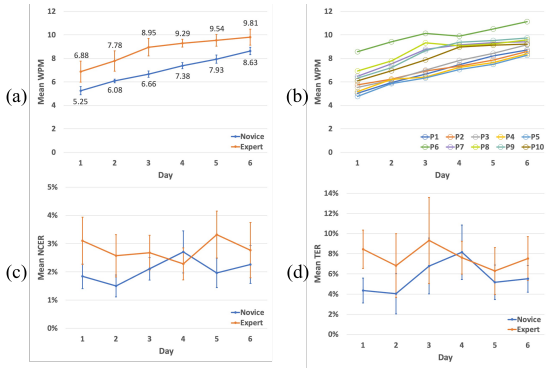


Fig. 9: Mean typing speed of novice group and potential expert group (a), mean typing speed of 10 participants (b), mean NCER (c) and TER (d) of novice group and potential expert group of 6 sessions.

Figure 9 (a) shows the daily mean typing speed of two groups. The mean typing speed of the novice group was 6.99 WPM (s.e. = 0.25), and their typing speed increased from 5.25 WPM (s.e. = 0.34) on the first day to 8.63 WPM (s.e. = 0.26) on the last day, raising by 64.38%. The mean typing speed of the potential expert group was 8.71 WPM (s.e. = 0.25), and their typing speed increased from 6.88 WPM (s.e. = 0.34) on the first day to 9.81 WPM (s.e. = 0.26) on the last day, raising by 42.59%. Figure 9 (b) shows the daily mean typing speed of all ten participants, where the fastest typing speed was recorded as 11.13 WPM achieved by one potential expert participant on the last day.

Error Rates For NCER, the results of ANOVA showed that 'session' ($F_{5,40} = .92$, $p = .48$, $\eta_p^2 = .103$) and 'session' × 'group' ($F_{5,40} = 2.35$, $p = .058$, $\eta_p^2 = .227$) had no significant effects on it, while 'group' ($F_{1,8} = 7.62$, $p = .025$, $\eta_p^2 = .488$) had a significant effect on it, and the novice group had a significantly lower NCER than the potential expert group ($p = .025$) in the pair-wise comparisons. For TER, no significant effects were found on 'session' ($F_{1.96,15.69} = 2.36$, $p = .128$,

$\eta_p^2 = .228$), 'session' × 'group' ($F_{1.96,15.69} = 1.25$, $p = .312$, $\eta_p^2 = .135$), and 'group' ($F_{1,8} = 3.20$, $p = .112$, $\eta_p^2 = .286$), and no significant differences were found in the pair-wise comparisons. The results indicated that multi-day training did not sacrifice accuracy. Figure 9 (c) and (d) show the mean NCER and TER for ten participants over six days. The mean NCER and TER of the novice group for six days were 2.06% (s.e. = 0.19%) and 5.66% (s.e. = 0.79%) respectively. The mean NCER and TER of the potential expert group for six days were 2.79% (s.e. = 0.19%) and 7.66% (s.e. = 0.79%).

5.3 Discussion

In term of efficiency, our LipText can reach 11.13 WPM, which is comparable to the state-of-the-art hands-free methods, such as RingText (13.24 WPM) [23] and iText (13.76 WPM) [12]. And as we can see in Figure 9 (a) and (b), the learning curve of the participants is still on an upward trend, and we believe that the typing speed can still be improved after more time of practice. In terms of accuracy, the six-day average NCER and TER for the novice group and the potential expert group are 2.06%, 5.66% and 2.79%, 7.66%, respectively. It can be found that the TER is higher relative to NCER, and the error rates of the novice group are lower than that of the expert group. The statistical analysis results show that the NCER of the novice group is significantly smaller than that of the potential expert group. We find that the reason why TER is higher than NCER is that the model is still not accurate enough to predict the letters, such as 'B', 'C', 'D', 'E'. The reason for the lower error rates of the novice group than the potential expert group may be that reading silently at a slower speed gives a more complete lip shape and thus a more accurate prediction. In terms of learnability, after six days of practice, the average typing speed of all ten participants improved by 52.15%, with the novice group improving by 64.38% and the potential expert group improving by 42.59%. This shows that our LipText is very novice-friendly and users can master it after a short period of practice.

6 Conclusions, limitations and future work

We have proposed LipText, a lip tracking-based text entry method in VR. Lip shape features are captured using a facial tracker and analyzed by a neural network to obtain the letters that the user reads silently. We also used the user's head motion to execute auxiliary selection to improve the correctness of text entry. Our method achieves 8.63 WPM for the novice group and 9.81 WPM for the potential expert group in typing speed and has the highest typing speed of 11.13 WPM. The error rates are 2.43% and 6.66% for NCER and TER respectively. Our method also has good learnability.

However, our method still has some limitations: 1) Limited dataset: We were unable to recruit enough participants, resulting in a smaller dataset, which prevented us from training a robust model. Additionally, the task is complex: users' speech rates are difficult to standardize, and several letters have similar lip shapes. These issues lead to limited model accuracy and a relatively high TER. 2) Model selection: We tested only three neural network models, which may not be the most suitable options for our task. 3) Inefficient selection process: Our method

uses a two-step selection process, which is not the most efficient. 4) Letter-level input only: Our method currently supports only letter-level input, and the typing speed is constrained by the time overhead of the user's silent reading. 5) Sample size and bias: The user study had a limited number of participants, with more males than females, which may introduce bias in the experimental results.

In future work, we plan to improve our method in the following areas: 1) Build a more comprehensive and generalizable dataset. Additionally, we will refine the neural network design to achieve more accurate and robust results, enabling single-step selection or reducing the need for two-step selections. 2) Redesign letters with similar lip shapes. As observed in the experimental results, 'space' achieved high accuracy both in the validation and test sets due to its distinct lip shape. Therefore, we may redesign letters with similar lip shapes to make them more distinguishable, improving model accuracy. 3) Incorporate word lip shape data into the dataset for word-level input, which could significantly increase the typing speed of our method. 4) Conduct broader user experiments with a larger and more diverse participant pool to further strengthen the validity and generalizability of our findings.

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

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)
2. Fu, R., Zhang, Z., Li, L.: Using lstm and gru neural network methods for traffic flow prediction. In: 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC). pp. 324–328. IEEE (2016)
3. Garbe, W.: {SymSpell} (6 2012), hhttps://github.com/wolfgarbe/SymSpell
4. Gers, F.A., Schraudolph, N.N., Schmidhuber, J.: Learning precise timing with lstm recurrent networks. Journal of machine learning research **3**(Aug), 115–143 (2002)
5. Graves, A., Schmidhuber, J.: Framewise phoneme classification with bidirectional lstm and other neural network architectures. Neural networks **18**(5-6), 602–610 (2005)
6. Gugenheimer, J., Dobbelstein, D., Winkler, C., Haas, G., Rukzio, E.: 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 (2016)
7. Hart, S.G.: Nasa-task load index (nasa-tlx); 20 years later. In: Proceedings of the human factors and ergonomics society annual meeting. vol. 50, pp. 904–908. Sage publications Sage CA: Los Angeles, CA (2006)
8. Jiang, H., Weng, D.: 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 (2020). https://doi.org/10.1109/VR46266.2020.00092
9. Jiang, H., Weng, D., Zhang, Z., Bao, Y., Jia, Y., Nie, M.: 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 (2018). https://doi.org/10.1109/ISMAR-Adjunct.2018.00051

10. Jiang, H., Weng, D., Zhang, Z., Chen, F.: Hifinger: One-handed text entry technique for virtual environments based on touches between fingers. Sensors **19**(14), 3063 (2019)
11. Ladefoged, P.: A course in phonetics. Thomson Wadsworth **86** (2006)
12. Lu, X., Yu, D., Liang, H.N., Goncalves, J.: itext: Hands-free text entry on an imaginary keyboard for augmented reality systems. In: The 34th Annual ACM Symposium on User Interface Software and Technology. pp. 815–825 (2021)
13. Lu, X., Yu, D., Liang, H.N., Xu, W., Chen, Y., Li, X., Hasan, K.: Exploration of hands-free text entry techniques for virtual reality. In: 2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR). pp. 344–349. IEEE (2020)
14. MacKenzie, I.S.: A note on calculating text entry speed. Unpublished work. Available online at http://www. yorku. ca/mack/RN-TextEntrySpeed. html (2002)
15. MacKenzie, I.S., Soukoreff, R.W.: Phrase sets for evaluating text entry techniques. In: CHI'03 extended abstracts on Human factors in computing systems. pp. 754–755 (2003)
16. Mott, M.E., Williams, S., Wobbrock, J.O., Morris, M.R.: Improving dwell-based gaze typing with dynamic, cascading dwell times. In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems. pp. 2558–2570 (2017)
17. Pham, D.M., Stuerzlinger, W.: Hawkey: Efficient and versatile text entry for virtual reality. In: 25th ACM Symposium on Virtual Reality Software and Technology. pp. 1–11 (2019)
18. Ruan, S., Wobbrock, J.O., Liou, K., Ng, A., Landay, J.A.: Comparing speech and keyboard text entry for short messages in two languages on touchscreen phones. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies **1**(4), 1–23 (2018)
19. Shneiderman, B.: The limits of speech recognition. Communications of the ACM **43**(9), 63–65 (2000)
20. Soukoreff, R.W., MacKenzie, I.S.: 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)
21. Vertanen, K.: Efficient correction interfaces for speech recognition. Ph.D. thesis, Citeseer (2009)
22. Whitmire, E., Jain, M., Jain, D., Nelson, G., Karkar, R., Patel, S., Goel, M.: 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 (2017)
23. Xu, W., Liang, H.N., Zhao, Y., Zhang, T., Yu, D., Monteiro, D.: 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 (2019). https://doi.org/10.1109/TVCG.2019.2898736
24. Xu, Z., Chen, W., Zhao, D., Luo, J., Wu, T.Y., Gong, J., Yin, S., Zhai, J., Yang, X.D.: 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 (2020)
25. 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 2017 CHI Conference on Human Factors in Computing Systems. pp. 4479–4488 (2017)
26. Yu, D., Fan, K., Zhang, H., Monteiro, D., Xu, W., Liang, H.N.: Pizzatext: Text entry for virtual reality systems using dual thumbsticks. IEEE Transactions on Visualization and Computer Graphics **24**(11), 2927–2935 (2018). https://doi.org/10.1109/TVCG.2018.2868581