Text Typing in VR Using Smartphones Touchscreen and HMD

Sabah Boustila* University of Toronto, Canada

Thomas Guégan† Tohoku University, Japan

Kazuki Takashima‡ Tohoku University, Japan

Yoshifumi Kitamura § Tohoku University, Japan

Abstract

In this work, we were interested in using smartphone touchscreen keyboard for text typing in virtual environments (VEs) with head-mounted display. We carried out an experiment comparing the smartphone to the ordinary devices: gamepad and HTC Vive Controllers. We represented the touchscreen keyboard in the VE with a virtual interface and the fingertips with tracked green circles. A confirm-on-release paradigm was employed for text typing. Results showed that the smartphone did not fully outperformed the other devices. However, unlike the other devices, smartphones users tended to correct progressively their error while typing thanks to their familiarity with the device.

Index Terms: Human-centered computing—Virtual reality—Touch screens—; Human-centered computing—User interface design—

1 Introduction and Related Work

Smartphones have been propelled on top of the most used devices for both masses and professionals. Several works suggested using these devices to interact with virtual reality (VR) applications using Head-Mouted Displays (HMDs). Thanks to a virtual representation of the device inside the virtual environment (VE), users can operate their smartphone while wearing the HMD [1].

In the context of text typing in VR, a virtual keyboard is displayed to reproduce touchscreen keyboard. For this purpose, the hands are as well represented in different ways; full tracked hand, semi-transparent hands or fingertips representation [3, 5]. HoVR-type uses a smartphone that possesses hovering functionality to provide a tactile-based text input [5]. Only the fingertips were represented with tracked circles to minimize visual clutter. Previous work compared touchscreen keyboard performance to physical keyboard [2]. touchscreen keyboard showed significantly slower performance than the physical one. The authors assumed that it is related to the usage of only one finger to write text, participants held the device with one hand and use the other hand for text typing. FaceTouch overcomes this problem with the principle of free-hand text typing [4]. This work suggested a smartphone placed on the front of a HMD. Users write with both hands however, for long usage, users mentioned the preference of having the option of switching to a hand-held smartphone after a period of interaction.

In our work we were interested in comparing smartphone to the most common VR interaction devices: gamepad and HTC VIVE hand controllers in the context of text typing. We suggested using a hand-held smartphone where users can hold the smartphone and can use both thumbs to write characters. The usage of this conventional method could help users to skip an initial learning phase for typing. We named our prototype Smartphone as a Controller (SaaC).

*e-mail: s.boustila@utoronto.ca
†e-mail: guethomas3@gmail.com
‡e-mail: takashima@riec.tohoku.ac.jp
§e-mail: kitamura@riec.tohoku.ac.jp

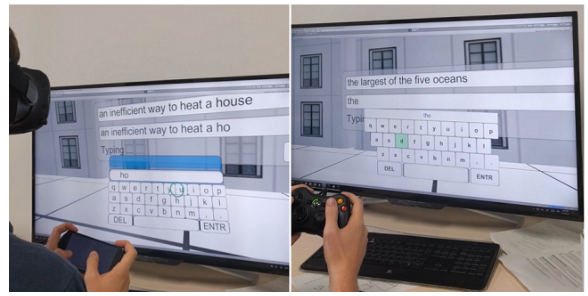

Figure 1: The image on the left shows a user writing a sentence in the VE using SaaC. The image on the right shows a user typing text using the Gamepad.

2 Smartphone as a Controller Prototype

Our proposal main concept is the redirection of every touch gestures tracked on a real smartphone device into a virtual interface hosted in the VR application.

The prototype is mainly based on a virtual representation (virtual interface) of the smartphone in the VE. We reproduced the same proportions leading to 1:1 mapping from the touchscreen to the virtual interface (VI). Therefore, the user knows where to interact with the real device according to this visual information. Fingertips were represented with green circles for better performance [3]. The smartphone orientation was fixed in landscape mode. The bottom two-thirds of the VI represent the keyboard while the top third represents a text area. Since users usually hold the smartphone at the waist level and look down when using it, our VI was placed on the bottom of the user's field of view and then calibrated for each participants according to his preferences.

2.1 Text Typing Principle

Text typing is based on the selection of a key representing a character. In our prototype we used *confirm-on-release* paradigm since the user needs to locate his finger before writing a character. The user presses the screen on the position he thinks the closest to his target (a character). Then, with the circle marker information, he verifies his position, and drag his finger if necessary. To confirm, he releases his finger triggering a writing action, see Fig. 1. This approach provides enough precision to perform tasks required using our prototype. Indeed, it has already shown the best performance in comparison to *confirm-on-touch* paradigm [5]. While writing, the text appears on the text zone on the virtual screen, allowing the user to check his word in the same sight (see Fig. 1).

3 Experiment

The purpose of our work was to investigate the performance of SaaC regarding HTC VIVE Controllers and gamepad in text typing. The experiment consisted of copying sentences using all these methods one by one.

2019 IEEE Conference on Virtual Reality and 3D User Interfaces
23-27 March, Osaka, Japan
978-1-7281-1377-7/19/$31.00 ©2019 IEEE

3.1 Method

We used a within-subject design with one independent variable, three typing methods: SaaC, Raycasting and Gamepad. In Raycasting condition, the participant used both HTC Vive Controllers, one for each hand. To type a character, he pointed the ray casted from the controller toward the target character (on a virtual keyboard) and then pressed the trigger. The participant could type text using both hands. In Gamepad condition, the participant used the joystick to choose the character on the virtual keyboard, and pressed a button to select it. The order of the typing methods was counterbalanced between subjects to avoid any order effects. We measured error rate (Minimum String Distance (MSD)), efficiency (Key Strokes Per Character (KSPC)) and typing speed (words-per-minute (wpm)).

Ten volunteer participants (3 females and 7 males) performed the experiment, all aged between 20 and 26 years old and had normal or corrected vision. The participants were all familiar with smartphones and its QWERTY keyboard.

3.2 Stimuli and Apparatus

We used an HTC VIVE HMD as a display device. For the SaaC, we used a Sony Xperia Z5 compact (4.6 inches, 128x72 mm). The Gamepad was an Xbox 360 controller, two HTC VIVE hand controllers were used for Raycasting method. The experiment was run using Unity 5 and Unity Mobile Android API on a Windows computer with an Intel Core i7-3930X CPU with 6 3.20 GHz cores and a NVIDIA GeForce GTX TITAN Black.

In the virtual scene, two text bars were displayed: one contained the sentence to copy and the other, placed below, was filled with the participant input. The virtual interface was displayed bellow the two text bars. This took place in a virtual small city, which was only here to avoid having an empty VE, see Fig. 1.

3.3 Procedure

The main task was to copy a set of sentences, one by one, using one of the three typing method. Each set was composed of 10 sentences. Three instructions were provided: (1) to memorize each sentence before typing it (although the sentence was always visible), (2) to type as fast as possible, but still take care of precision and (3) to correct a mistake if the participant realized it instantly (but not after).

Participants first assisted to a sentence typing demonstration by the experimenter. Then, they performed a calibration phase to adjust text scale and VI position. During this phase, they were free to write some sentences for 5 minutes. The experiment started, participant typed a sentence and confirm by pressing the ENTER key, the next sentence appeared after 2 seconds. Depending on the typing speed, this phase took between 5 and 8 minutes to complete. The three conditions were separated by a total 5 minutes break.

4 Results and Discussion

The non-parametric Kruskal-Wallis test revealed that typing method had significant effect on all dependent variables; typing speed (*p-value* = 2.2e-16), efficiency and error rate (*p-value*= 0.0003677 , *p-value* = 0.01006, respectively). All results are reported in table 1. In terms of text typing speed, raycasting method was the fastest method 21.42 wpm, followed by SaaC 12.10 wpm then gamepad 8.98 wpm. Our SaaC' wpm mean value is in accordance with the results of Grubert *et al.* [2], reporting 11.6 wpm when typing with touchscreen keyboard.

In terms of efficiency, Raycasting method performed with an efficiency of 91.84%, SaaC with 93.30 % and Gamepad with 97.89%. We can notice a trade-off between speed and efficiency. The Gamepad method was the slower method however, it was the most efficient according to the KSPC. Participants wrote slowly but with less error in comparison to SaaC and Raycasting. Inversely, Raycasting method was the fastest, however the great speed has consequences on both efficiency and error rates. Participants typed

Method	Wpm	Efficiency	Error rate
Raycasting	21.42	91.84	2.30
SaaC	12.10	93.30	1.12
GamePad	8.98	97.89	1.90
p-value	$2.2e-16$	0.0003677	0.01006

Table 1: Text typing results according to the typing method. All mean, *p-values* showing the influence of typing methods on wpm, efficiency and error rate.

text quickly, but they also made many mistakes and corrected their errors less frequently.

As for error rate, participants with SaaC had significantly less error than the other methods, participants considered their errors while typing and corrected them progressively. We think that this behavior when using SaaC is because that SaaC is intuitive and more natural method for text typing. In fact, all participants are already familiar with text typing using smartphones. Therefore, they had the reflex to correct their errors while text typing, leading to more correct sentences at the end.

Our text typing method was implemented without any automated error-correction this allowed us in one hand, to avoid the development of an auto-corrector (which is not our aim) and in another hand to avoid dependency to the auto-correction method. Integrating auto-corrections could improve typing speed for the smartphone.

5 Conclusion

In this work we were interested in the use of smartphone as interaction device with HMD. For this purpose, we conducted an experiment comparing smartphone to gamepad and HTC VIVE controllers in text typing. The smartphone was represented in the virtual environment with a virtual interface having same dimensions than the real touchscreen. The aim was to allow users to use their own smartphones to interact with virtual environments and without any additional tracker.

Results revealed that HTC VIVE controllers were the fastest in text typing whereas, SaaC led to writing correctly. Indeed, users considered their errors and correct them progressively. In the future, we would like to suggest SaaC as a general controller covering diverse interactions. Therefore, we will investigate SaaC in several other user cases such as selecting and manipulating objects.

References

[1] A. P. Desai, L. Pea-Castillo, and O. Meruvia-Pastor. A window to your smartphone: Exploring interaction and communication in immersive vr with augmented virtuality. In *2017 14th Conference on Computer and Robot Vision (CRV)*, pp. 217–224, May 2017. doi: 10.1109/CRV.2017.16

[2] J. Grubert, L. Witzani, E. Ofek, M. Pahud, M. Kranz, and P. O. Kristensson. Effects of hand representations for typing in virtual reality. *CoRR*, abs/1802.00613, 2018.

[3] J. Grubert, L. Witzani, E. Ofek, M. Pahud, M. Kranz, and P. O. Kristensson. Text entry in immersive head-mounted display-based virtual reality using standard keyboards. *CoRR*, abs/1802.00626, 2018.

[4] J. Gugenheimer, D. Dobbelstein, C. Winkler, G. Haas, and E. Rukzio. Facetouch: Enabling touch interaction in display fixed uis for mobile virtual reality. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, UIST '16, pp. 49–60. ACM, 2016. doi: 10.1145/2984511.2984576

[5] 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, Jan 2017. doi: 10.1109/ICCE.2017.7889285