

Vitty: Virtual Touch Typing Interface with Added Finger Buttons

Yongjae Lee and Gerard J. Kim^(✉)

Digital Experience Laboratory, Korea University, Seoul, Korea
lyj930303@gmail.com, gjkim@korea.ac.kr

Abstract. In this paper, we propose to add individual finger buttons to a nominal interaction controller to realize a QWERTY like touch typing experience in virtual reality. The method is called Vitty, and the intuition behind it is real life typing is emulated by mapping the fingers/buttons to the appropriate sectional rows in the virtual QWERTY layout. Vitty is expected to reduce the mental and physical fatigue in the usual aiming and selecting of each small alphanumeric keys. We compare the performance and usability of Vitty to those of the typing interfaces as available with the current popular VR interaction controllers, and found that with minimal training, Vitty can be an inexpensive yet viable alternative for text input for VR.

Keywords: Virtual reality · Alphanumeric/Text input · Interaction · Controller · Performance evaluation · Usability

1 Introduction

With the innovations for affordable head mounted displays (HMD), stable sensing and high-end computer graphics, immersive virtual reality (VR) has attracted much attention lately. Aside from the reality and immersion, the virtual experience is often contingent on natural and usable interaction as well. In fact, interactive techniques for VR have long been studied, e.g. particularly for the generic tasks of navigation, object selection and manipulation [1]. However, there has not been a satisfactory solution to the task of text entry in VR due to the difficulty in tracking individual fingers and providing even a minimal haptic/tactile feedback. Moreover, its utility had been relatively low and overlooked, while today, its importance has risen significantly by the prevalence of social networking.

In this paper, we propose a simple solution: adding buttons, for each finger (except for the thumb), to the conventional interaction controller as an interface for virtual touch typing for a QWERTY style keyboard (hence, named “Vitty”). In the usual touch typing, one uses the muscle memory to reach and locate the wanted key with the eight fingers (the index to the pinky) from the home middle row. With Vitty, the same muscle memory (aided by the visual feedback) is applied to the VR interaction controller in a similar way to match the fingers to the desired keys. This way, Vitty emulates the usual method of typing while providing the important sense of haptic/tactile feedback through the

individual buttons (See Fig. 1). We compare Vitty to the typing interface as available with the current popular VR interaction controllers.

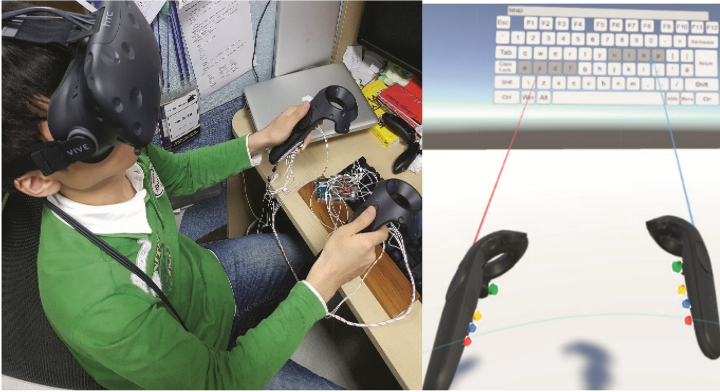


Fig. 1. Text input for VR using Vitty: selecting the keyboard section using the ray casting with the interaction controllers then entering the individual letter with finger mapped buttons.

2 Related Work

There is a large amount of pervious work on various VR interaction techniques [2–4]. We only outline notable works in text entry for VR. The most popular and conventional way of text entry in VR setting is the “aim and shoot” style, in which a hand-held device or hand-mounted sensor is used to cast a virtual ray and select a particular key and making the final confirmation using a button (or other discrete input method) [5, 6]. A more direct method is to use a glove like device that attempts to sense individual finger movements and map them into the virtual space to realize virtual QWERTY style typing [7–9]. A slight variant is the hand-mounted but restricted set of buttons, each corresponding to an alphabetic key [10, 11]. Such a non-QWERTY method would require extensive training, however. Recently, improved external finger tracking and sensing technologies have allowed the use of bare hands, relieving the user from having to use the cumbersome glove-like hand worn device [12]. However, these sensors are still not accurate enough, often need to be installed in the environment (making the input system not self-contained) and has a limited operating range. Combined with the lack of haptic feedback, such a scheme generally has low usability.

One other interesting approach is capturing and segmenting out the imagery of a real keyboard and using hands (using a computer vision method), and blending it into the virtual scene [13]. While such an approach makes it possible to use the familiar conventional keyboard, the keyboard (on a fixed desktop location) is not fit for active usage while navigating in the VR space.

3 System Overview

The Vitty QWERTY keyboard is divided in its layout into several sections (see Fig. 2) which are first selected by the standard ray-casting technique (Fig. 1) aimed by the interaction controller, e.g. those that are equipped with sensors and few buttons for orientation/position control and discrete command input (we have used the motion controller from the HTC Vive). Each section contains 4 letters that are mapped to four different fingers as used in the normal QWERTY typing, and thus four different finger buttons attached to the controller (see Fig. 3). For instance, the letter “w” in the normal typing is to be entered by the left ring finger and similarly so by the corresponding finger button in the proposed scheme. Once the section which includes the “w” key (colored in red in Fig. 2) is selected, the individual letter input is made through the corresponding finger button. For instance, to input “w”, the user points the left hand controller to the red section (in Fig. 1) and press the third button with the ring finger.

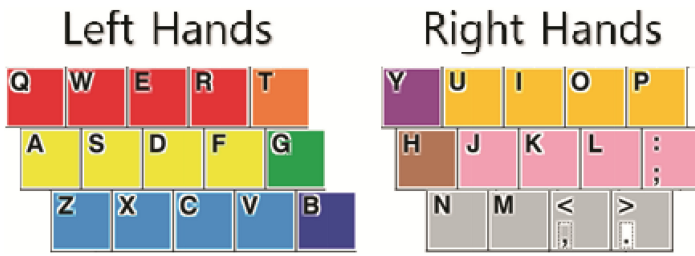


Fig. 2. The sections in the virtual QWERTY keyboard layout. For example, to input “w”, the user points the left hand controller to the red section and press the third button with the ring finger. (Color figure online)

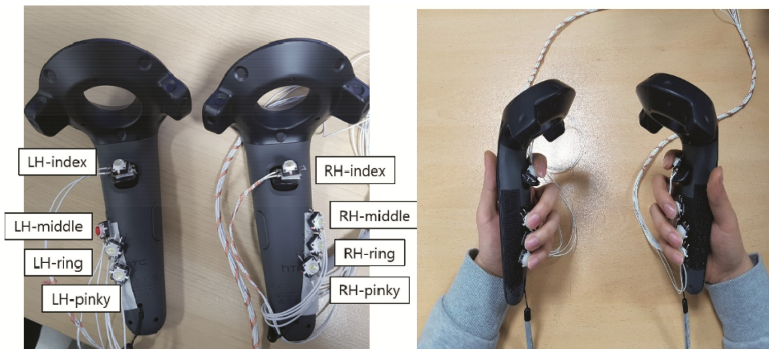


Fig. 3. Finger buttons added to HTC Vive interaction controller for QWERTY style text input.

Note that there are few exceptions. There are sections with only one letter such as those for “t”, “g”, “b”, “y”, “h” and other special keys like the space bar and backspace (see Fig. 2). The typing process is still the same; that is, select the section by ray-casting and entering the letter with the “index finger” (as so in normal typing). Such a nearly

equivalent finger mapping (to QWERTY) makes the proposed method quite natural. Figure 1 shows a screen shot of the virtual keyboard interface within the virtual space. The rationales behind the proposed method are summarized as follows:

1. Using any layout other than QWERTY would require extensive training.
2. With stable tracking performance, the conventional interaction controllers have shown favourable usability for supporting general interactive tasks in the virtual space, e.g. by ray casting, joy stick, buttons and touch-pad. Adding yet another dedicated device for text entry would be prohibitive.
3. Using buttons as key input provides tactility and haptic feedback (vs. e.g. purely with ray-casting).
4. Despite the stable tracking, selecting the individual alphabetic key using ray-casting can still be difficult or tiring in comparison to the selection of the larger keyboard “section.” And the successive button click is very fast, due to the same finger-to-letter mapping as the normal typing.
5. Adding buttons is inexpensive and they can be used for other interactive purposes. Interaction controllers already employ several buttons anyway.

4 Usability Experiment

4.1 Experimental Design

To demonstrate and validate the prospective advantages of Vitty as noted, we conducted a usability experiment to compare Vitty to two other nominal VR text input methods, thus, there are three test conditions as explained in Table 1; (1) ray casting one-handed, (2) ray casting two handed and (3) Vitty. The typing performance and subjective usability were measured as dependent variables. Note that Vitty is designed to be used with two controllers, and it is compared to both the one and two handed RC for fair assessments.




4.2 Test Apparatus

The experimental task was carried out in a VR environment, viewed using the HTC VIVE headset and the input made possible with the VIVE interaction controller, tracked by separate sensing modules. In the case of RC-2 and Vitty, two interaction controllers were used. The interaction controllers for Vitty were added with the finger buttons (see Fig. 3) which were implemented using the Arduino controller [14]. These added finger buttons were attached to the original VIVE controller using the Velcro so that they could be fitted according to a particular user hand comfortably.

4.3 Experimental Procedure

Nine paid subjects participated in the experiment. All the subjects had to possess a typing skill of higher than 38 words-per-minute (in English alphabets). While only subject had the prior experience of using VR equipment, none complained of any discomfort. After

Table 1. The three experimental conditions tested.

No.	Factor level	Explanation	
1	Ray casting One-handed (RC-1)	One interaction controller (held in the dominant hand) is used to select the desired key by ray casting and confirm the input by a dedicated trigger button input.	
2	Ray casting Two-handed (RC-2)	Two interaction controllers are used held in both hands. Each is used to select the desired key in the left or right part in the same manner as the one-handed aim-and-shoot.	
3	Vitty	Subsection of the keyboard is selected by ray casting with the controller, then individual key input by corresponding finger buttons. Two hands/controllers are used.	

collecting their basic background information, the subjects were briefed about the purpose of the experiment and given instructions for the experimental tasks. A short 10-minute training was given, without wearing the HMD to reduce the fatigue as much as possible, so that the subjects could be familiarized with the experimental process and three text input methods.

In the main experiment, the subject sat, wore the HMD, and held the controllers and carried out the typing task. The chair and viewing distance to the virtual keyboard were adjusted for the subject's maximum comfort. The experimental task comprised of a set of sentences (13) from the Mackenzie's "Phrase set for evaluating text entry techniques" [15] for each test treatment. Each treatment was presented in a balanced fashion using the Latin square methodology, and in each treatment the task was repeated three times.

The task started by the user pressing the "start" button, then 13 sentences would appear above the virtual keyboard for which the user was to enter. A single block (13 sentences) was finished by touching the "enter" button after entering all the sentences. Three performance indicators were measured: the task completion time, the time interval between the individual key input and the error rate (number of incorrectly input letters). The user was asked to make the entries correctly as fast as possible. After the session, the subjects filled out a usability questionnaire (see Table 2). The user rested between each treatment and the total experimental session took about 1 h.

Table 2. The usability survey questions.

Q1: Ease of use	Rate how easy it was to enter the text. 1 (very hard) ~ 7 (very easy)
Q2: Physical load	Rate the level of physical fatigue. 1 (very tiring) ~ 7 (not tiring at all)
Q2: Mental load	Rate the level of mental fatigue. 1 (very tiring) ~ 7 (not tiring at all)
Q4: Ease of learning	Rate how easy it was to learn the text entry input method. 1 (very difficult) ~ 7 (very easy)
Q5: General satisfaction	Rate how satisfied you were for the text entry input method. 1 (very unsatisfactory) ~ 7 (very satisfactory)

4.4 Results

Figure 4 shows the comparative results of the task completion time among the three tested condition. The graph shows the performance change over the three blocks so that the learning effect can be taken into consideration. In the first two blocks, there were statistically significant differences among the three input methods, with RC-2 being the fastest, followed by the RC-2, then Vitty. By the third block, however, the statistical differences were reduced to an insignificant level. As for the error rate, no statistically significant differences could be found among the three methods.

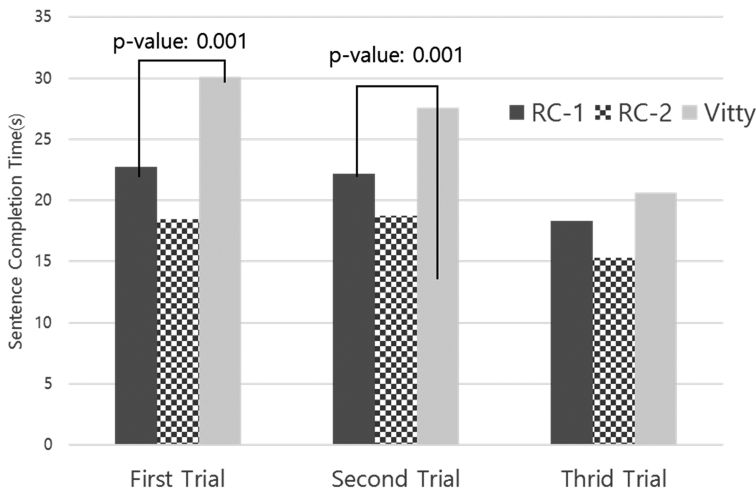


Fig. 4. Task completion times in three trial blocks for the three tested conditions (RC-1, RC-2 and Vitty). Unlike the first two blocks, the third block shows no significant difference in the performance between the conventional and Vitty.

Figure 5 shows two examples of the time taken to input two successive key inputs; one for two keys that belong to different sections (e.g. “a” then “n”) and other for those that belong to the same section (e.g. “e” then “r”). For the former, we see that RC-1

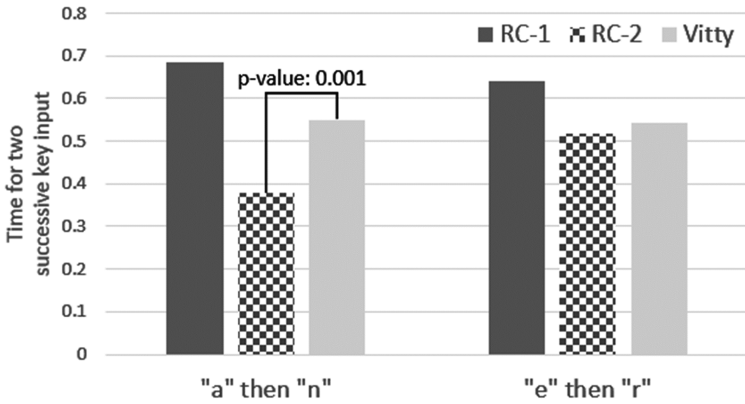


Fig. 5. Time taken for two successive key inputs, for those belonging to: (1) different sections, e.g. “a” then “n” (left) and (2) same section, e.g. “e” then “r”. In the former, the RC-1 performs the best, while for the latter, RC-1 and Vitty performs at a similar level.

generally performs better than RC-1 (because of the less movement using two hands) and also Vitty (because Vitty requires a button press after ray casting to the desired section). As for the latter, Vitty shows an improvement statistically equaling the performance of RC-2.

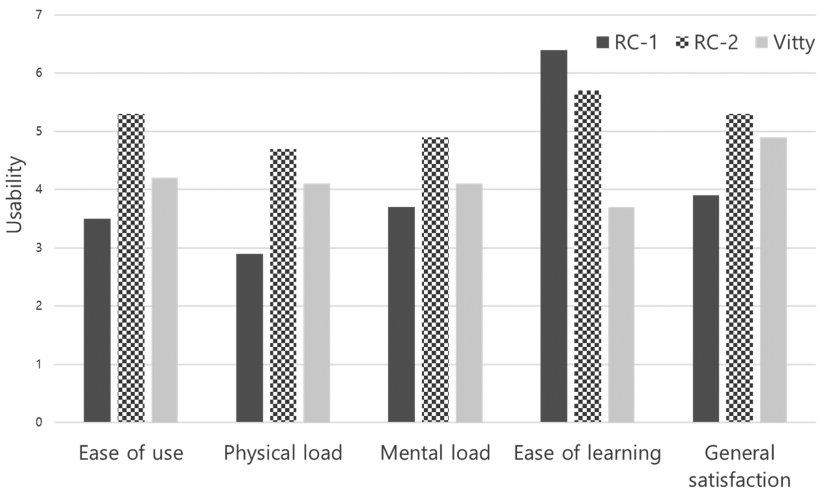


Fig. 6. Responses to the usability survey across the three text input methods. RC-2 generally showed the highest usability.

Figure 6 shows the usability survey results. In most categories, the RC-1 showed the worst usability particularly with respect to the fatigue factor (having to manage all input with just one hand). RC-2 was the rated the most usable among the three due to its familiarity and directness. One intuition behind Vitty was that real life typing could be

emulated by mapping the fingers/buttons to the appropriate sectional rows in the virtual QWERTY layout. Vitty was expected to reduce the mental and physical fatigue in the usual aiming and selecting of each small alphanumeric keys, with the current less-than-perfect sensing and tracking technologies. However, while not exactly QWERTY style, using ray casting was still much simpler and familiar to the average user. Ironically, sectional row selection to emulate QWERTY like experience seem to have required some amount of training and getting used to. For this reason, its expected advantage was not sufficient to supersede the performance of RC-2.

We found that the QWERTY style touch typing was not followed all the time in actuality, and the way such users change the typing rules on their own could not be re-enacted with Vitty as in the real world. Post-briefing revealed that many subjects felt the added buttons to be less than natural. The stance of the hands with respect to the keyboard was also not totally natural because it had to be aiming toward the desired sectional row. Subjects also complained of the ray obscuring the sectional rows and made the initial selection difficult. Despite these little overlooked factors which contributed to a non-ideally recreated QWERTY typing experience, the task performance after few blocks of trials showed a promising improvements.

5 Conclusion and Future Work

In this work, we have proposed a text entry method for VR, called Vitty, which attempted to emulate the QWERTY style touch typing by adding individual finger buttons to a nominal interaction controller. Despite the less than perfect implementation, after just minimal training, Vitty showed comparable performance to the conventional ray casting based text input method. Thus, with a more professional and ergonomic design, we believe Vitty can offer a viable way to realize easy text input by only extending the controller in an inexpensive way. We see an opportunity to transfer such familiar keyboard oriented interface to the virtual space, e.g. as used in many desktop games (“f” for move forward, spacebar for “shoot”) and software interfaces (copy and paste using control-C and control-V).

Acknowledgment. This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (No. 2011-0030079) and also by the MISP (Ministry of Science, ICT & Future Planning), Korea, under the National Program for Excellence in SW (R2215-16-1007) supervised by the IITP (Institute for Information & communications Technology Promotion)

References

1. Bowman, D., Kruijff, E., LaViola Jr., J., Poupyrev, I.: 3D User Interfaces: Theory and Practice, CourseSmart eTextbook. Addison-Wesley (2004)
2. Feiner, A., Olwal, S.: The flexible pointer: an interaction technique for selection in augmented and virtual reality. In: Proceedings of the UIST 2003, pp. 81–82 (2003)

3. Poupyrev, I., Billinghurst, M., Weghorst, S., Ichikawa, T.: The go-go interaction technique: non-linear mapping for direct manipulation in VR. In: Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology, pp. 79–80. ACM (1996)
4. Thomas, B., Piekarski, W.: Glove based user interaction techniques for augmented reality in an outdoor environment. *Virtual Reality* **6**(3), 167–180 (2002)
5. HTC VIVE User guide. http://dl4.htc.com/web_materials/Manual/Vive/Vive_User_Guide.pdf?_ga=1.64742511.1367646068.1462853716. Accessed 10 May 2016
6. PlayStation4 User’s Guide, Entering characters. <http://manuals.playstation.net/document/en/ps4/basic/osk.html>. Accessed 10 May 2016
7. Bowman, D., Wingrave, C., Campbell, J.M., Ly, V., Rhoton, C.: Novel uses of Pinch Gloves™ for virtual environment interaction techniques. *Virtual Reality* **6**(3), 122–129 (2002)
8. Howard, B., Howard, S.: Lightglove: wrist-worn virtual typing and pointing. In: Proceedings of the Fifth International Symposium on Wearable Computers, pp. 172–173. IEEE (2001)
9. Rosenberg, R., Slater, M.: The chording glove: a glove-based text input device. *IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.)* **29**(2), 186–191 (1999)
10. Alphagrip. <http://www.alphagrips.com/>. Accessed 10 May 2016
11. Trewgrip. <http://www.trewgrip.com/>. Accessed 10 May 2016
12. Leap Motion Developer, VR Keyboard. <https://developer.leapmotion.com/gallery/vr-keyboard>. Accessed 10 May 2016
13. McGill, M., Boland, D., Murray-Smith, R., Brewster, S.: A dose of reality: overcoming usability challenges in VR head-mounted displays. In: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, pp. 2143–2152. ACM (2015)
14. Banzi, M., Shiloh, M.: Getting Started with Arduino: The Open Source Electronics Prototyping Platform. Maker Media, Inc. (2014)
15. MacKenzie, I.S., Soukoreff, R.W.: Phrase sets for evaluating text entry techniques. In: CHI 2003 Extended Abstracts on Human Factors in Computing Systems. ACM (2003)