The Importance of Typing in Virtual Reality

INDY HOSIER, Colorado State University, USA

Additional Key Words and Phrases: Virtual Reality (VR), Head Mounted Displays (HMDs), Mixed Reality (MR), Words Per Minute (WPM)

ACM Reference Format:
Indy Hosier. 2022. The Importance of Typing in Virtual Reality. 1, 1 (May 2022), 11 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 ABSTRACT

While immersed in a virtual reality setting, such as a head mounted display, it is often times difficult for users to interact with physical objects in the real world, specifically a keyboard. This makes alphanumeric text entry in VR a difficult task. Current implementations of text entry involve the use of large learning curves, expensive equipment, or other mediums to solve the problem of text entry in a VR setting. In this research study, I aim to provide users with a simple, but yet effective method of entering text in a VR application using only a head mounted display and a user's hands, no other medium necessary. I then analyze participants' text entry while using two different keyboard layouts and environments. The first environment contains a split keyboard where users are allowed to use all 10 of their fingers to enter text using this split keyboard. The second environment contains a full keyboard, but users are only allowed to use their index fingers to input text. After introducing users to each of these environments, I then analyze and compare their input in terms of typing efficiency, including words per minute, error rates, and accuracy percentages to determine the best environment for users to implement text entry while being immersed in the VR application.

2 INTRODUCTION

Virtual Reality (VR) offers incredible abilities, some that are even thought to be unachievable in a real world environment. Head mounted displays (HMDs) are a demonstrated example that make it possible for a user to interact with a VR setting. HMDs can provide users with a very unique, immersive, and captivating VR experience. However, oftentimes users must have the ability to utilize real world objects while immersed in virtual reality [8]. One of the primary devices that VR users must have access to is the keyboard [8]. Without a visual sense of the keyboard, it often makes it unusable to users wearing a head mounted display. Instead, many users are forced to use hand held controllers to enter alphanumeric text entries. These hand held controllers are far less efficient and offer less performance value than a keyboard to enter text in many VR applications.

Motivation for this project comes from the fact that it is very difficult to interact with real world objects while operating a VR headset or a head mounted display [5]. The goal of this experiment was to design a VR application where users can both visualize and utilize a keyboard while using an HMD.

Author's address: Indy Hosier, indyh@rams.colostate.edu, Colorado State University, 775 W Lake St, Fort Collins, Colorado, USA, 80521.

XXXX-XXXX/2022/5-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

In addition, users of this application are also able to interact with different virtual environments comprised of both VR and Mixed Reality (MR). The application is centered around a virtual typing environment where users are able to test the efficiency of two different keyboards, using only their hands, no controllers necessary. Contribution of this project aims to eliminate the gap between users and real world objects in virtual reality, specifically a keyboard. Development of this application allows users to interact with a virtual keyboard with their hands while wearing a head mounted display without the need of the HMD's corresponding controllers. Users of this application are also capable of determining the most efficient, comfortable, and immersive environment that best fits their needs. Throughout this research study, participants underwent an experiment, defining an ideal virtual environment to use while utilizing a virtual keyboard.

3 RELATED WORK

Typing has become almost an essential part of our world today. In almost any workplace today, computers are a large part of the infrastructure that makes them successful. Companies today are even trying to move forward with developing a technologically feasible option for employees to work at their office from home. An example of this includes Remote Pair Programming where 2+ programmers are able to work on the same source code while in different locations [2]. A study in Clemson was conducted using the VirtualDesk application where developers are able to meet in a virtual reality environment to solve source code bugs [2]. The VirtualDesk application provides users of a head mounted display with a virtual office space containing the user's computer screen. Users of this application are still able to access a physical keyboard and can utilize their work space in a virtual environment, being the key components of such an efficient application [8]. The Clemson study indicated benefits of greater knowledge transfer, higher code quality, and even fewer defects in the computer code [2].

There are currently implementations being introduced to virtual reality environments to make text input seem more of a smooth and transitional process. An example of implementation includes the Microsoft HoloLens which provides user text input by means of a holographic keyboard and a head rotation controlled pointer, navigated by user worn glasses [7]. A study conducted by Yu et al., [7] reported user text input from the HoloLens reaching performance levels upwards of 25 words per minute (WPM), but falling far short of the USA average of 40 WPM. Another implementation that has been recently trending is the implementation of wearable gloves that detect certain motions of the hand and fingers to represent distinct characters in text entry [6]. Users wearing these gloves are able to receive incredible haptic feedback, such as genuine keyboard clicks and real-time vibrations to generate realistic interactions with a physical keyboard [9]. Relating to wearable objects, the idea of a force-based technologies have also been introduced. An example of a force-based technology is the Shapeshifter, developed by students at the University of California, Merced [3]. By using force-based technologies, users can use opaque surfaces, such as a desk, or even a human body, to interact with a VR environment, and more specifically, enter text through the sensation of touch on a physical surface [3]. From the latest state of the art standpoint, the OptiTrack motion capturing system can precisely pinpoint user finger movements, fast or slow [4]. Using this capturing technology, the implementation of tracing finger movements and auto-text correction can be utilized, providing much faster text input in a VR setting [4]. The usage of smartphone touchscreens and video game controllers have even been widely introduced as an alternative for text input in a VR setting [1]. However, by using smartphones or controllers of any type, a less captivating and immersive experience is introduced to the user.

Relating more to this specific experiment, another way of implementing text input in a VR environment is to use a technique called spacial enhancement [10]. Spacial enhancement is able to create the best, spatially aware keyboard by tracking the user's hands and spacing the keyboard as necessary. The keys' positions are generated based on the bone tracings of a user's hands through the Leap Motion Controller [10]. This hand-in-hand corresponds with my specific experiment design, as the objective is to create a universally accepted spacing of the virtual keyboard tailored to all individuals using the application. Although technologies such as wearable gloves, enhanced motion tracking technology, or even spatial enhancement offer mobility and eyes-free text entry, they require a large amount of learning, expensive equipment, and can limit the use of other input methods, such as a cell phone or a physical keyboard [6].

Complications with many implementations of text entry in VR such as input speed, learning curves, inaccessible equipment, etc., make creating a VR system capable of competing with typing in reality an open challenge. Many of these other implementations aim to tackle text entry in VR with expensive and hard to comprehend technologies, such as finger tracking gloves, enhanced motion capturing cameras, or even with the help of a smart phone. The central objective of the developed application within this research study was to provide users with a simple, but yet effective method of implementing text entry from a keyboard, using only their hands and a head mounted display. However, it can be concluded that the research of text input in VR is a wide area of research and is a very complicated task. The goal of this experiment corresponds to this area of research by offering another solution to the complicated process of text entry in a VR environment.

With great typing skills, time is spent more efficiently and effectively. Hand in hand, efficiency drastically improves productivity, no matter the task at hand. Whether productivity boils down to getting tasks done faster or saving time for other tasks, typing can be considered the root of how effectively computational tasks are completed. It can be argued that typing while being immersed in a virtual reality environment can develop long term benefits. By developing an efficient application within an ever-expanding subject of virtual reality, I can hopefully provide a typing application for others that is easy to comprehend and effective at offering the advantage of a simple but efficient virtual typing environment.

4 METHODOLOGY

To implement this experiment, a within-subject design was used. 10 participants (3 female) were selected to conduct this experiment. Four of the participants had no VR experience and six of which had some background in using VR systems. Nine of the ten participants also had experience typing on a daily basis. All participants had normal or corrected to normal vision and were introduced to every test condition with repeated measures. The technologies that were included in this experiment were the Oculus Quest 2 that the participants used to view the VR application, developed with the Unity Game Engine. This experiment contained one independent variable with 2 different levels and one dependent variable with 4 different levels. The 2 levels of the independent variable were the two different keyboards that the users were introduced to and interacted with. These 2 different keyboard layouts are visualized in Figure 1 and Figure 2, respectfully. The 4 different dependent variables were measures of user input. These 4 different user input measures were gross words per minute (WPM), net WPM, error rate, and accuracy percentage. All of these user input measures and their calculations can also be visualized in the results section. The same physical environment was kept the same the entire experiment to reduce environmental bias. The physical environment was a well lit room that was kept quiet throughout the entire experiment.

At the beginning of the experiment, 10 participants were all asked to wear the Oculus 2 Quest VR Headset. Once inside of the VR application, users were presented with the first typing environment (Figure 1). This environment contained a virtual keyboard that was split into two different keyboards of left and right sections, a pair of virtual hands and five fingers on each hand that were tracked by the VR headset, an input dialogue box for users to type into, and three phrases of text that users were instructed to type as input into the input box. The virtual keyboard in the first environment

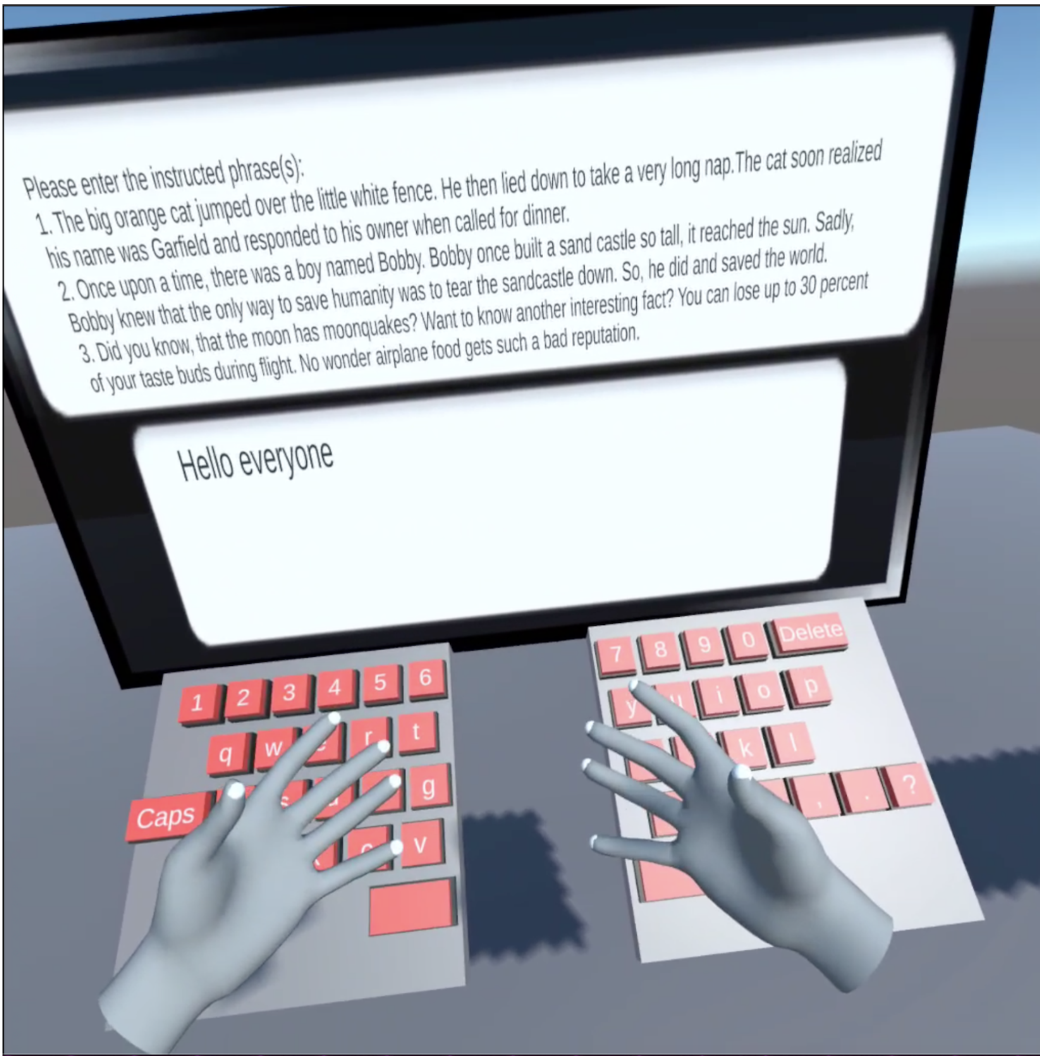

Fig. 1. First Virtual Typing Environment.

was split into left and right sections and the rows of the keys were formatted according to a physical keyboard with corresponding indentation of the keys. The tracked hands in this environment contained mappings and interaction of all 10 fingers, five on each hand with the white spheres indicating the process of pressing a key. Participants were given two minutes each to type a single instructed phrase, also shown in Figure 1 above the input field. The participants were instructed to type out their given phrase to the best of their ability. Once the two minutes had passed by, the participants were then asked to type the next instructed phrase while being given another two minutes. This process took place a total of three times for each participant in the first environment. At the end of the third iteration for each participant, user input was recorded for analysis. Once the input was recorded, a new participant was introduced to the virtual typing environment and was

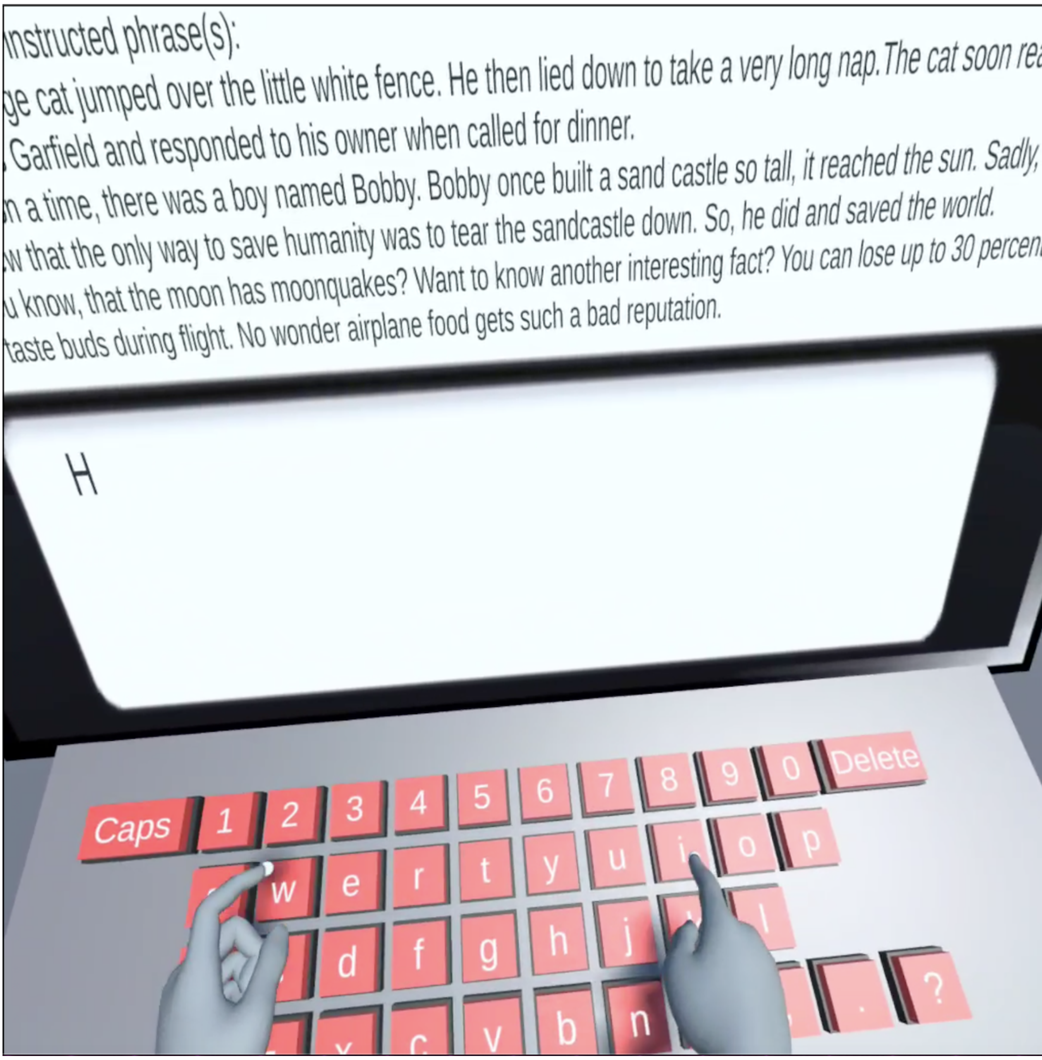

Fig. 2. Second Virtual Typing Environment.

instructed to repeat the same process as the previous participant. Each participant typed a total of three phrases over a total span of six minutes, allocating two minutes to each phrase.

After the first set of 10 participants had typed their inputs in the first environment, the same participants were then introduced to the second virtual typing environment shown in Figure 2. This environment was similar to the first environment, however, the virtual keyboard in the second environment was not formatted with any indentation of a physical keyboard and the board was not split into left and right sections. The hand mapping interactions were also different. Participants in this environment were restricted to only using one finger of each hand to type out each instructed phrase. The white spheres in Figure 2 again indicate the process required to press a key. The participants in the second environment were again given an instructed phrase to type within two minutes to the best of their ability. Once finished with the first phrase, the participants then moved on to the next phrase with another allotted two minutes. This process again took place a total of three times for each participant in the second virtual typing environment. At the end of the third iteration for each participant, user input was recorded for analysis. Once each participant had completed all three iterations in each environment, the participant was dismissed.

The goal of this experiment was to test which environment was more efficient in terms of user input measurements such as gross words per minute (WPM), which calculated overall speed of character entry, Net WPM, which calculated speed of character entry while taking errors into account, error rate, measured by how many errors per minute the user recorded, and lastly, accuracy, measured by percentage of correct keys pressed when entering input. More can be seen about these measurements in the results section. Both environments provided users with an unobstructed view of the keyboard without the wrist interfering in both VR environments. These environments also allowed the user to implement text entry, using only their hands and the head mounted display, no other medium was necessary. A hypothesis that was formed for this experiment was that if the participants' net words per minute in each environment were compared against each other, the second environment, containing the full keyboard with only the index fingers allowed to type, will have a higher average of net WPM than the first environment containing a split keyboard and all 10 fingers allowed to type. This hypothesis was formed because the second environment is a more universally recognized keyboard layout and with only the index fingers allowed to type, there will be less room for error when typing.

5 RESULTS

Once each participant had finished all three of their two minute iterations in the environments, their inputs were recorded for analysis. The three phrases were built randomly, but kept the same throughout the experiment for each participant to reduce bias in the difficulty of the phrases. The three phrases that were given were:

1. The big orange cat jumped over the little white fence. He then lied down to take a very long nap. The cat soon realized his name was Garfield and responded to his owner when called for dinner.

2. Once upon a time, there was a boy named Bobby. Bobby once built a sand castle so tall, it reached the sun. Sadly, Bobby knew that the only way to save humanity was to tear the sand castle down. So, he did and saved the world.

3. Did you know, that the moon has moonquakes? Want to know another interesting fact? You can lose up to 30 percent of your taste buds during flight. No wonder airplane food gets such a bad reputation.

These three phrases were given a long length, designed so that the participants would ideally not finish the phrase in under two minutes. However, if they did complete their phrase under two minutes, they were asked to continue to the next phrase until their two minutes were up. None of the participants in this experiment were able to complete a phrase in under the allotted two minutes.

User input was recorded to measure efficiency in terms of typing. To provide what user input was recorded, the best iteration of the entire experiment by Participant 7 in the second virtual typing environment is shown below:

1. The big orange cat jumped over the little white fence. He then lied down to take a very long nap. The cat soon realized his name was Garfield and responded to his owner (168 characters)

2. Once upon a time, there was a boy named Bobby. Bobby once built a sand castle so tall, it reached the sun. Sadly, Bobby kknew that the only way to save humanity was t (166 characters)

3. Did you know, that the moon has moonquakes? Want to know another interesting fact? You can lose up to 30 percent of your taste buds during flight. No (150 characters)

The first measure of user input taken was gross words per minute. This measurement provides the speed a user was able to type in words per minute, regardless of errors. In typing, a word is recognized universally as five characters. In order to calculate gross words per minute, the equation is: Gross WPM = (number of characters typed/5) / number of minutes. For example, Participant 7 was able to type 168 characters total in two minutes for their first instructed phrase. To calculate their gross WPM for phrase 1, the equation is as follows: Gross WPM = (168/5)/2 minutes = 16.8 WPM for the first instructed phrase. All participant's gross WPM were calculated for each instructed phrase to calculate their average gross WPM for all three phrases, shown in Table 1.

Participant	Avg. Gross WPM Environment 1	Avg. Gross WPM Environment 2
1	9.46 WPM	10.20 WPM
2	4.10 WPM	13.30 WPM
3	8.23 WPM	7.03 WPM
4	4.93 WPM	11.13 WPM
5	5.50 WPM	12.17 WPM
6	7.37 WPM	14.13 WPM
7	3.97 WPM	16.13 WPM
8	3.90 WPM	8.97 WPM
9	2.80 WPM	10.37 WPM
10	5.03 WPM	10.33 WPM

Table 1. Average Participant Gross Words Per Minute in Each Typing Environment.

The second measure of user input taken was net words per minute. This measurement provides the speed a user was able to type in words per minute, with errors being calculated in the equation. Again, a word is recognized universally as five characters. In order to calculate net words per minute, the equation is: Net WPM = (Gross WPM) - (uncorrected errors/number of minutes). An uncorrected error is an error that the user made while typing that was not corrected. For example, Participant 7 was able to type 166 characters total in two minutes for their second instructed phrase. However, the user made an error in typing the word "kknew" in phrase 2. This example has 1 error throughout the entire phrase, which is the duplicate character "k." To calculate their net WPM for phrase 2, the equation is as follows: Net WPM = 16.8 WPM - (1 uncorrected error/2 minutes) = 16.1 WPM for the second instructed phrase. All participant's net WPM were calculated for each instructed phrase to calculate their average net WPM for all three phrases, shown in Table 2.

The third measure of user input taken was error rate. This measurement provides the amount of errors a user made per minute. In order to calculate error rate, the equation is: Error rate = number of errors/number of minutes. Again, an uncorrected error is an error that the user made while typing that was not corrected. For example, Participant 7 made an error in typing the word "kknew" in phrase 2. This example has 1 error throughout the entire phrase, which is the duplicate character "k." To calculate their error rate for phrase 2, the equation is as follows: Errors per minute = (1 uncorrected error/2 minutes) = 0.5 EPM for the second instructed phrase. All participant's error rate were calculated for each instructed phrase to calculate their average error rate for all three phrases, shown in Table 3.

The fourth and final measure of user input taken was accuracy. This measurement provides the percent of correct entries typed of the total entries typed. In order to calculate accuracy, the equation is: Accuracy percentage = (number of correct entries/number of total entries) * 100. Again, an uncorrected error is an error that the user made while typing that was not corrected. For example, Participant 7 was able to type 166 characters total in two minutes for their second instructed phrase.

Participant	Avg. Net WPM Environment 1	Avg. Net WPM Environment 2
1	9.13 WPM	8.34 WPM
2	2.27 WPM	13.30 WPM
3	7.90 WPM	7.03 WPM
4	4.60 WPM	10.8 WPM
5	3.50 WPM	11.33 WPM
6	7.03 WPM	13.30 WPM
7	3.47 WPM	15.97 WPM
8	3.40 WPM	8.47 WPM
9	2.8 WPM	9.03 WPM
10	4.03 WPM	10.00 WPM

Table 2. Average Participant Net Words Per Minute in Each Typing Environment.

Participant	Avg. Error Rate Environment 1	Avg. Error Rate Environment 2
1	0.33 EPM	1.67 EPM
2	1.67 EPM	4.17 EPM
3	0.33 EPM	0.33 EPM
4	0.33 EPM	0.33 EPM
5	2.0 EPM	0.83 EPM
6	0.33 EPM	0.66 EPM
7	0.83 EPM	0.17 EPM
8	0.50 EPM	0.5 EPM
9	0.0 EPM	1.17 EPM
10	1.0 EPM	0.33 EPM

Table 3. Average Participant Error Rate in Each Typing Environment.

Participant 10 made one error in typing the word "kknew" in phrase 2. This example has 1 error throughout the entire phrase, which is the duplicate character "k." To calculate their accuracy for phrase 2, the equation is as follows: Accuracy percentage = (165 correct entries typed/166 total entries typed) * 100 = 99.40% of correct entries typed for the second instructed phrase. All participant's accuracy were calculated for each instructed phrase to calculate their average accuracy for all three phrases, shown below in Table 4 in the discussion section.

6 DISCUSSION

Of the four user statistics that were measured, perhaps the most important statistic is the participants' Net WPM. When calculating a typist's words per minute, the net words per minute statistic is universally recognized as the most efficient at calculating a typist's speed. Net WPM essentially calculates a typist's speed of how many correct words they are able to type in a minute. In this experiment, the majority of participants' net words per minute were raised significantly in the second environment containing the full keyboard as shown in Figure 3 containing a comparison of the participants' average net WPM in each environment. This provides support for the previously claimed hypothesis that the second environment would contain a higher average of participants' net words per minute.

Participant	Avg. Accuracy Environment 1	Avg. Accuracy Environment 2
1	99.31%	96.53%
2	91.82%	94.48%
3	99.10%	99.31%
4	97.73%	99.42%
5	92.14%	98.63%
6	99.24%	98.84%
7	97.76%	99.80%
8	97.92%	98.92%
9	100%	97.34%
10	96.16%	99.31%

Table 4. Average Participant Accuracy in Each Typing Environment.

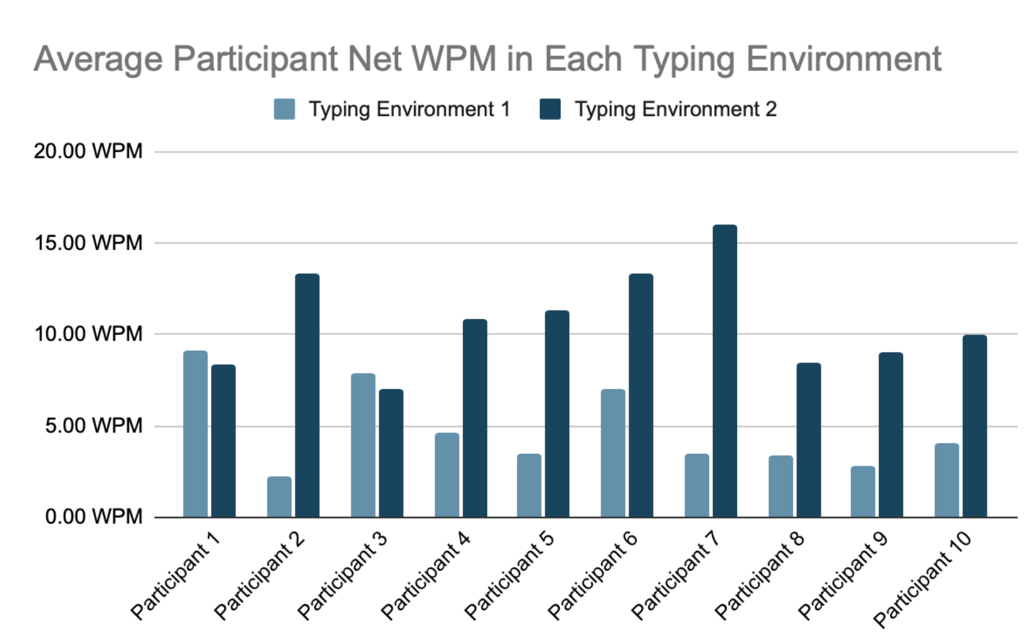

Fig. 3. Comparing Participant Net WPM in Each Environment.

The results of participants' net WPM logically make sense. By having a more recognized environment of a regular keyboard, users in this environment most likely contain a legacy bias from a realistic keyboard. Users in this environment are also only allowed to use their index fingers to type out characters. By only using their index fingers, there is less room for error if there were other fingers that were wrongfully recognized for typing an unwanted entry on the keyboard.

Although net WPM saw a significant increase in the second typing environment, so did gross WPM which did not calculate errors. This gave rise to a majority increase in participants' errors per minute. However, there is also a logical explanation for this as well. If users are able to type at a faster rate, there is more of a chance for an error upon entry of text. Even though users in the second typing environment are generally typing faster and leaving more of a chance for errors, their accuracy percentage saw an increase in the second environment. This can be explained by the amount of character entries there were in the second typing environment compared to the first. If users were able to type more correct characters at a faster rate, their accuracy of correct entries raises higher in response.

Overall, the comparison of the two typing environments did a good job at capturing just how well a physical keyboard is recognized universally. When users were prompted to type an instructed phrase using a full keyboard and only their index fingers instead of a split keyboard and all 10 fingers, a general increase of words per minute and accuracy percentage was still shown. Although the averages of participants' net WPM was far from the national average of 40 WPM, the highest net average WPM was still 15.97 WPM. In hindsight, it would have been a more accurate experiment if both typing environments were introduced and measured with all 10 fingers being allowed to enter text. In further work, an experiment that created both of these environments with every finger allowed to enter text would have done a better job at determining and capturing the most efficient way of text entry in a VR setting. Additionally, the Oculus Quest that was used in the experiment was limited by capturing hand movements due to shadows, quicker hand movements, and even knuckle and joint tracking. By using a data glove that was able to track finger tip movements, this experiment would have better been able to capture user hand movement and more accurately define user typing speed in words per minute. However, the goal of the experiment was met by introducing a typing environment that allowed users to use text entry while in a VR setting using only the head mounted display and their hands. With some spacial enhancement and recognition tweaks, I believe both of the typing environments in this experiment can provide a very efficient way of using text entry in a VR setting, with no other medium necessary than a virtual reality head mounted display.

REFERENCES

[1] Sabah Boustila, Thomas Guégan, Kazuki Takashima, and Yoshifumi Kitamura. 2019. Text Typing in VR Using Smartphones Touchscreen and HMD. *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)* (2019). https://doi.org/10.1109/VR.2019.8798238

[2] James Dominic, Brock Tubre, Charles Ritter, Jada Houser, Colton Smith, and Paige Rodeghero. 2020. Remote Pair Programming in Virtual Reality. *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)* (2020). https://doi.org/10.1109/ICSME46990.2020.00046

[3] Tafadzwa Joseph Dube, Kevin Johnson, and Ahmed Sabbir Arif. 2022. Shapeshifter: Gesture Typing in Virtual Reality with a Force-based Digital Thimble. *Association for Computing Machinery* (2022). https://doi.org/10.1145/3491101.3519679

[4] John Dudley, Hrvoje Benko, Daniel Wigdor, and Per Ola Kristensson. 2019. Performance Envelopes of Virtual Keyboard Text Input Strategies in Virtual Reality. *2019 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)* (2019). https://doi.org/10.1109/ISMAR.2019.00027

[5] Jens Grubert, Lukas Witzani, Eyal Ofek, Michel Pahud, Matthias Kranz, and Kristensson. 2018. Effects of Hand Representations for Typing in Virtual Reality. *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)* (2018), 151–158. https://doi.org/10.1109/VR.2018.8446250

[6] Jens Grubert, Lukas Witzani, Eyal Ofek, Michel Pahud, Matthias Kranz, and Per Ola Kristensson. 2018. Text Entry in Immersive Head-Mounted Display-Based Virtual Reality Using Standard Keyboards. *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)* (2018). https://doi.org/10.1109/VR.2018.8446059

[7] Pascal Knierim, Valentin Schwind, Anna Maria Feit, Florian Nieuwenhuizen, and Niels Henze. 2018. Physical Keyboards in Virtual Reality: Analysis of Typing Performance and Effects of Avatar Hands. *Association for Computing Machinery* (2018). https://doi.org/10.1145/3173574.3173919

[8] Jia-Wei Lin, Ping-Hsuan Han, Jiun-Yu Lee, Yang-Sheng Chen, Ting-Wei Chang, Kuan-Wen Chen, and Yi-Ping Hung. 2017. Visualizing the Keyboard in Virtual Reality for Enhancing Immersive Experience. *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)* (2017). https://doi.org/10.1145/3102163.3102175

[9] Chien-Min Wu, Chih-Wen Hsu, Tzu-Kuei Lee, and Shana Smith. 2016. A virtual reality keyboard with realistic haptic feedback in a fully immersive virtual environment. *2017 Virtual Reality* (2016). https://doi.org/10.1007/s10055-016-0296-6

[10] Yang Zhen, Chen Cheng, Lin Yuqing, Wang Duming, Li Hongting, and Zu Weidan. 2019. Effect of spatial enhancement technology on input through the keyboard in virtual reality environment. *Applied Ergonomics* (2019). https://doi.org/10.1016/j.apergo.2019.03.006