



A New Japanese Input Method for Virtual Reality Applications

Kosuke Komiya and Tatsuo Nakajima 

Department of Computer Science and Engineering, Waseda University, Tokyo, Japan
{kosukekomiya, tatsuo}@dc1.cs.waseda.ac.jp

Abstract. In recent years, virtual reality is becoming popular with the advent of PlayStation VR and mobile VR. However, due to the restrictions of the hardware, it is difficult to reproduce the same thing in the virtual space, as it is in the reality. One of the typical examples is the character input. It is extremely difficult to reproduce convenience and speed equivalent to traditional used input methods such as personal computer keyboard input or smartphone flick input, in the virtual space. Therefore, in this study, we aim to propose a new character input method, focusing on typing Japanese characters, which has an input speed at a certain level, is touch-typable, and requires no controller in a user's hands. The proposed approach uses Leap Motion as an input device. By tracking the movements of the finger of a user, the user selects a pair of a consonant and a vowel with two bending and stretching movements, which makes a Japanese letter. From a preliminary experiment, our method achieved the input speed of 42.1 Characters per Minute. In addition, this paper discusses the duration of the practice needed to use this method.

Keywords: Japanese character input method · Mobile virtual reality
Learning effects

1 Introduction

Due to the advent of PlayStation VR [9], mobile VR, and many other VR devices such as Oculus Rift [10] or HTC Vive [1], virtual reality is becoming popular. However, because of the restrictions of the hardware, it is very difficult to implement an input method with high efficiency.

In this study, we propose a new character input method which focuses on typing Japanese letters for virtual reality applications. Japanese language has three types of characters. Hiragana, Katakana, and Kanji. Input in Japanese is often performed by typing Hiragana first, and then converting it into Kanji. When inputting the word 雨 (ame) which means “rain”, with a PC keyboard for example, a user inputs あめ(ame) first, and then presses the space button to convert the あめ to 雨. This action is needed because あめ can mean not only “rain” (雨), but also “candy” (飴) or many other words. Conversion into Kanji is needed to determine the word the user wants to input.

Our method uses the characteristic of Japanese Hiragana, that one letter is formed from the pair of a consonant and a vowel as shown in Fig. 1. For example, the character

そ is consisted of the combination of the consonant “S” and the vowel “O”. The set of あ,い,う,え,お is called あ行(あ(a) line), the set of か,き,く,け,こ is called the か行(か(ka) line), and so on.

W	R	Y	M	H	N	T	S	K	A	
わ	ら	や	ま	は	な	た	さ	か	あ	A
	り		み	ひ	に	ち	し	き	い	I
を	る	ゆ	む	ふ	ぬ	つ	す	く	う	U
	れ		め	へ	ね	て	せ	け	え	E
ん (n)	ろ	よ	も	ほ	の	と	そ	こ	お	O

Fig. 1. Japanese syllable table

Using this characteristic, we implemented a Japanese input method which satisfies the following three characteristics. The first is to maintain an input speed above a certain level. Because there is no clear standard for the metric, we referred to the results of the similar past researches. The second is that a user can perform touch typing. This is an important feature which both the keyboard input of personal computers and the flick input of smartphones have. As a result, it is possible for a user to notice when the user makes a mistake in the middle of an input, which makes it possible to reduce the amount of corrections. Also, being touch-typable makes it possible to input characters while looking at other things. The third is to be able to input without having a controller in a user’s hand. This allows the user to input, when his/her hands are dirty or with other controllers. We expect the user to be able to text chat at a bearable speed.

From the preliminary experiment, we achieved the input speed of 43.1 CPM (Characters per Minute). In addition, from the experiment, we showed that only 7 days of 15 min practice is needed to reach the input speed of 40 CPM.

The paper is structured as follows. In Sect. 2, we show some related work. This includes some input methods which can be used in a virtual space with relatively high input speed, and input methods which uses the characteristics of Japanese characters, same as the proposed method in this study. Section 3 includes the explanation about our approach. This includes the hardware used for this method, and how the input method works. In Sect. 4, we explain about the preliminary experiment. This section shows the method, results and the discussions of the experiment. Section 5 presents the evaluation of the proposed approach. This section includes the experiment we have performed, the results, and the discussions about the current approach. Finally, in Sect. 6, we conclude this study.

2 Related Work

2.1 VR Text Input for Vive

VR Text Input for Vive shown in Fig. 2 is an input method using HTC Vive. A user selects a consonant at a position touching the circular touch pad of the HTC Vive controller and inputs a character by selecting a vowel by the inclination of the controller. The feature of this input method is similar to the input method proposed in this study, where the pseudonym consists of a combination of 10 consonants and 5 kinds of vowels.

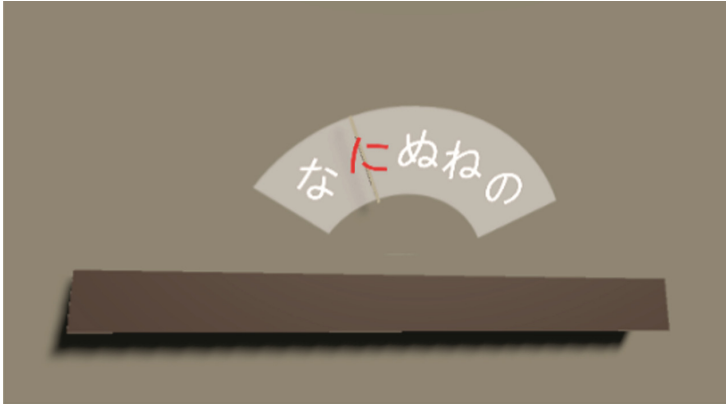


Fig. 2. A screenshot of VR text input for Vive

2.2 Japanese GazeTalk

Japanese GazeTalk introduced in [2] is also an input method which uses the characteristic of Japanese hiragana, that a hiragana is consisted of a combination of 10 consonants and 5 vowels. Figure 3 is the on-screen keys used to input [2].

さ行	か行	あ行 + 「一」	入力 トップメニュー	変換	あ行・わ行・ん
た行	くたに		変換	ち	て
な行	は行	に	ら行、わ行+ん	と	っ (小字)
		だ行 (濁点入力)	バックスペース		

Fig. 3. Layout of the input menu in the Japanese GazeTalk

The user types a Japanese character by selecting a consonant by gazing at the button shown on Fig. 3 (left), and then selecting a vowel by gazing at the button shown on Fig. 3 (right). According to the experiment performed in [2], after certain amount of practice, character input speed was approximately 25 CPM.

2.3 Punch Keyboard

The punch keyboard [11] is a keyboard type input method as shown in Fig. 4 [4]. A user enters each key by pushing in the virtual space. The feature of this input method is that the keyboard is curved, and both keys are designed to be perpendicular to the viewpoint. This approach makes it possible to reduce input errors.

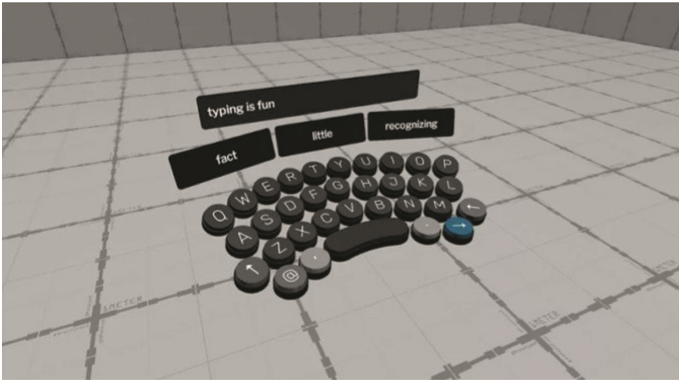


Fig. 4. A screenshot of punch keyboard

2.4 Limitations of the Previous Work

The disadvantages of the above three input methods introduced in this section is that it is necessary to prepare a certain degree of space in front of a user when using the methods in the VR space. Therefore, when these methods are used, it becomes difficult to see other objects while inputting characters.

3 Our Approach

In this study, Leap Motion [6] is used as an input device. Leap Motion is a device that can acquire the positions of the joints of both hands with an installed infrared camera. In addition, by using Unity Core Assets [7] provided on the official site of Leap Motion, it is possible to acquire the state of bending and stretching of each finger. Also, this device can be attached not only to head mounted displays (HMD) such as Oculus Rift or HTC Vive, but also to mobile VR [8]. In addition, since these devices do not hinder the use of other controllers, the input method proposed in this study can be used in many situations. In addition to these merits, it is also true that detecting hand movements with

Leap Motion is not always accurate. It is very difficult to constantly track the joints accurately. From this reason, in the proposed method, we used the bending and stretching movements, which can be relatively detected accurately, to lower the misrecognition rate.

In the method proposed in this study, as shown in Fig. 5 (left), a bending and stretching operation is performed once with a finger assigned to a consonant of a character to be inputted. For example, if a user wants to enter a character in the “か(ka)” line, he/she bends and stretches the index finger of the right hand, and he/she bends and stretches the thumb of the left hand if he/she wants to enter the characters of the “は(ha)” line. As shown in Fig. 5 (right), the bending and stretching operation is performed with the finger assigned to the vowel of the input character. Characters are inputted by these two bending and stretching operations. By performing the operation as shown in the figure, it is possible to input characters of “け(ke)”. For erasing a character, a user bends and extends the index finger, middle finger, and the ring finger at the same time. Though conversion between Hiragana and Kanji is necessary for Japanese input (explained in Sect. 1), the conversion was not yet implemented in this method.

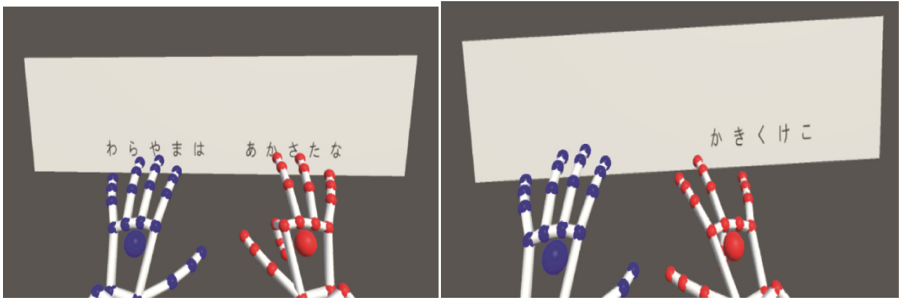


Fig. 5. Determining the consonant & the vowel

4 Preliminary Experiment

As a preliminary experiment shown in [5], we compared the proposed input method with VR Text Input for Vive introduced in Sect. 2. The reason for choosing a comparison with VR Text Input for Vive is that the input method emphasizes the input of Japanese syllabary like the proposed method in this study. In performing this experiment, a participant in the experiment actually practiced using respective input methods until he/she learned the character arrangement of both input methods.

For the word input, the participant inputs 6 words randomly selected from 60 kinds of preliminarily prepared 6 characters words, and measured the time taken to input, and the number of input errors. This series of flows was set as one set, and 10 sets were performed in succession.

For the character input, the participant inputs 20 randomly chosen pseudonyms from “あ” to “ん”, and measured the time taken to input, and the number of input errors. The number of input mistakes in this experiment is the number of times the action of erasing

one character was performed. This series of flows was set as one set, and 10 sets were performed in succession.

The experiment result for word input is as shown in Fig. 6 (up). From this result, we can see that the input speed of the proposed method is equivalent to VR Text Input for Vive in word input. The average of 10 sets is 42.9 CPM (Characters per Minute) for VR Text Input for Vive, 43.1 CPM for the proposed method. The average number of input errors is 4.4 times for Text Input for Vive and 1.8 times for the proposed method.

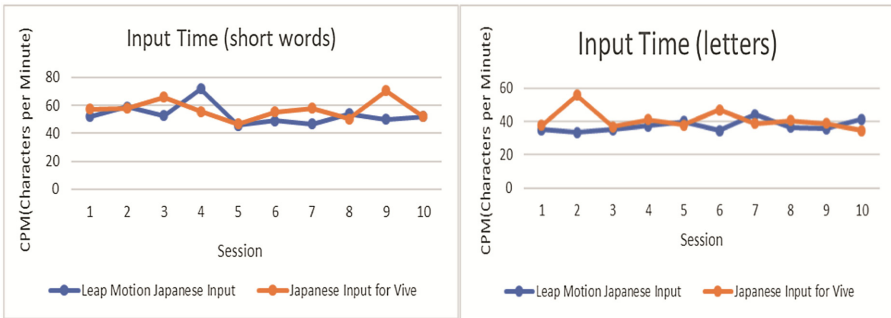


Fig. 6. Input speed when writing short words and random letters

The experiment result for character input is as shown in Fig. 6 (down). From this result, it is understood that input speed of the proposed method is equivalent to that of VR Text Input for Vive in the case of inputting a random character. The average of 10 sets is 33.7 CPM for VR Text Input for Vive and 36.2 CPM for the proposed method. The average number of input errors is 4.4 times for Text Input for Vive and 1.8 times for the proposed method.

In any of the input methods, input speed when inputting random characters was slower than the input speed when inputting a word. It is considered that the reason is that the time from recognition of the input word (character) to actual input is presented as overhead. In the experiment of inputting a word, it suffices to recognize six times of words, but it seems that the above results were obtained because it was necessary to recognize 20 times in the experiment of inputting characters.

5 Evaluation

As Evaluation, we performed an experiment with five participants (all right-handed) to see the length of the practice needed to use this method. Each participant was asked to input words for 15 min to practice the usage of the proposed method. The input words were randomly selected three characters words. The participants were allowed to take a break during the 15 min practice (without stopping the timer) to reproduce the expected actual usage in everyday life. After the 15 min practice, the participant was asked to input six of the six characters words, and measured the time took to input the words, and the number of input errors. The set of practicing and time measuring was performed

seven times on different days contiguous as possible. After performing the set seven times, the participants were asked the following questions.

1. **Fatigue**

- Did you become fatigued during the experiment?
- Was there any difference between the first day and the last day?
- What did you do to prevent being fatigued?

2. **Usage**

- Is the input speed bearable for text chatting?
- Which part of the method was hard to use?

3. **Touch-typing**

- Were you able to touch-type on the last day?

4. **Other**

- Do you have any comments about the method?

The experiment results are shown in Figs. 7 and 8 below.

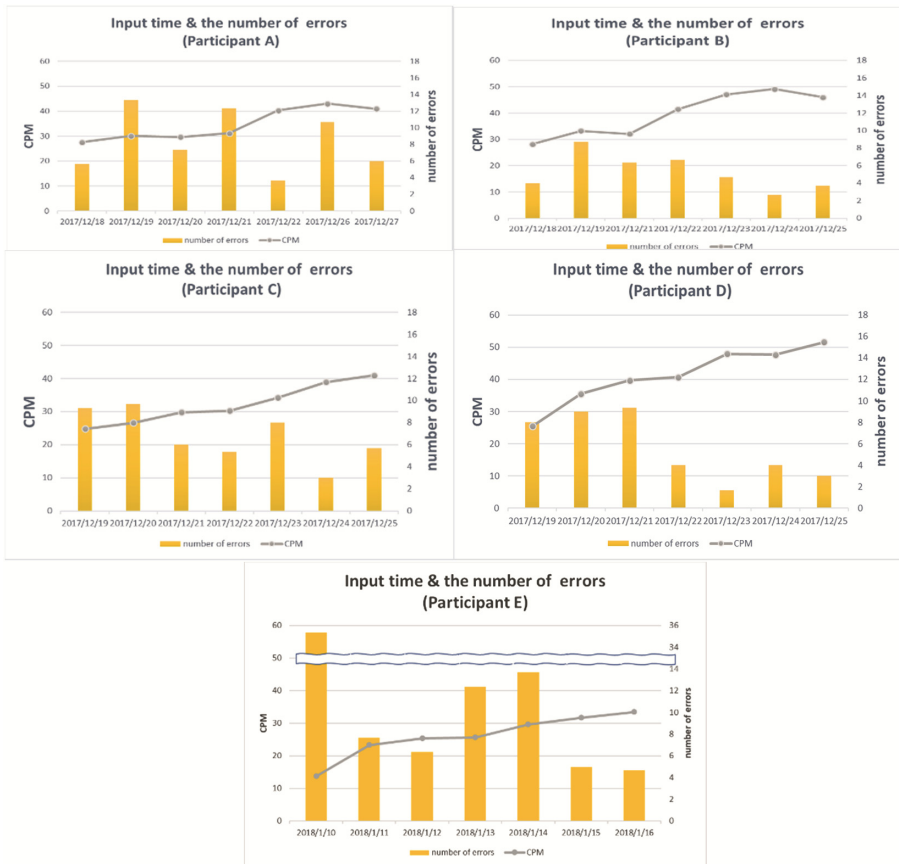


Fig. 7. Input time and the number of errors

Figure 8 shows the input speed scored by each participant on the last day, and the comparison between the proposed method and the PC keyboard. The input speed of the PC keyboard is measured using [3]. The number shown on the chart is the number of English alphabets typed in 30 s.

4 out of 5 participants achieved the input speed over 40 CPM. Considering the fact that in the preliminary experiment, the input speed achieved was 43.1, we can say that 7 days of practice was enough for most of the participants to achieve the expected input speed. In addition, from the graph on Fig. 8, We can anticipate that the input speed using the proposed method and the speed using PC keyboard are correlated.

From Fig. 7, we can see the improvement of the participants considering the input speed, while for most of the participants, number of errors seems to have no clear difference between the first day and the last day.

Participant	"Leap Motion Japanese Input" input speed	PC keyboard input speed
A	40.94	131
B	46.00	125
C	40.97	136
D	51.63	155
E	33.45	90

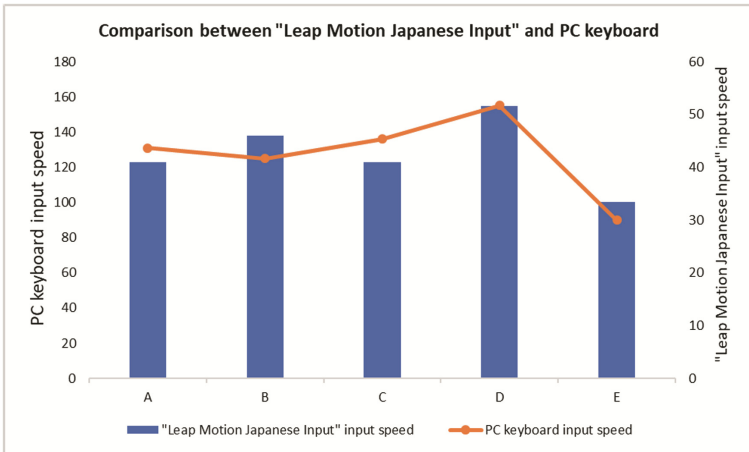


Fig. 8. Comparison between “Leap Motion Japanese Input” and PC keyboard

The figures below show the answers of each participant to the questions asked after the experiment.

participant	comment
A	using for 15 minutes is tiring. No fatigue if used with rest.
B	Felt pain on the ring finger. No fatigue if used with rest
C	Text chat with frequent typing (communications which requires immediate response) will be tiring. No pain if not frequent.
D	No fatigue.
E	No fatigue.

Fig. 9. Answers to the question “*Did you become fatigued during the experiment?*”

Figure 9 shows the answers to the question “*Did you become fatigued during the experiment?*”. As we can see from the figure, every participant answered that the method can be used for non-frequent text chat without being tired. However, most of the participants answered that the method is tiring for contiguous usages.

participant	comment
A	No difference.
B	No difference.
C	Less fatigue on the last day.
D	Less fatigue on the last day. Learned the hand position and ways of bending fingers to prevent fatigue
E	Less fatigue on the last day. Learned the hand position and ways of bending fingers to prevent fatigue

Fig. 10. Answers to the question “*Was there any difference between the first day and the last day?*”

Figure 10 is the answers to the question “*Was there any difference between the first day and the last day, considering fatigue?*”. Two participants answered there were no difference between the first day and the last day. However, three participants answered that they felt less fatigue in the last day. Considering the comment of participant D and

participant E, we can see that fatigue can be reduced by changing way of using the method. This can be seen from Fig. 11 below.

participant	comment
A	Use an arm rest
B	Don't use contiguously.
C	Use an arm rest. Don't raise the arm too high.
D	Use an arm rest. Don't raise the arm too high. Face lower to do so.
E	Learned the appropriate hand position

Fig. 11. Answers to the question “*What did you do to prevent being fatigued?*”

Three participants answered that using an arm rest is a way to prevent being tired. Two participants mentioned about the arm height. Since holding the arms high in the air is one of the main reasons that makes the user tired, keeping the arms low can prevent being tired.

participant	comment
A	Fast enough
B	Fast enough.
C	Fast enough for typing only Hiragana.
D	Fast enough for typing only Hiragana.
E	Fast enough.

Fig. 12. Answers to the question “*Is the input speed bearable for text chatting?*”

Figure 12 is the answers to the question “*Is the input speed bearable for text chatting?*”. All participants answered that the method was fast enough to use for text chatting. However, some participants mentioned about the conversion to Kanji.

Figure 13 is the answers to the question “*Which part of the method was hard to use?*”. 4 out of 5 participants mentioned about the recognition error.

participant	comment
A	Low recognition accuracy of Leap Motion.
B	Recognition error between middle finger and ring finger.
C	Recognition error between ring finger and little finger.
D	Recognition error between middle finger and ring finger.
E	Nothing.

Fig. 13. Answers to the question “Which part of the method was hard to use?”

Figure 14 is the answers to the question “Were you able to touch-type on the last day?”. Three participants answered that they were able to touch-type after seven days of practice. However, two participants needed more practice for the low frequently used characters.

participant	comment
A	Yes.
B	Yes, except the low frequently used characters.
C	60~70%, except the low frequently used characters.
D	Yes.
E	Yes.

Fig. 14. Answers to the question “Were you able to touch-type on the last day?”

Figure 15 is the answers to the question “Do you have any comments about the method?”. 3 participants mentioned about the feedback when they made the bending and stretching movement. They commented that it would be better to have a feedback with a sound, for example, so that the user can notice if the system actually switched between the consonants and the vowels.

participant	comment
A	Better to have a feed back when switching between consonants and vowels.
B	Better to have a feed back when switching between consonants and vowels.
C	Right hand was mainly used. There may be problems for left-handed people.
D	No comments.
E	Better to have a feed back when switching between consonants and vowels.

Fig. 15. Answers to the question “*Do you have any comments about the method?*”

6 Conclusion and Future Work

6.1 Conclusion

From the results of the experiment and the answers to the questions, we showed that 7 days of 15 min practice was enough to achieve the input speed, fast enough for text chatting. In addition, we showed that the fatigue would not be a problem when this method is used for text chatting. On the other hand, for contiguous usages like writing a report, this method would not be useful considering the fatigue.

However, conversion between Hiragana and Kanji remains to be implemented. In addition, considering the comment of some participants, adding a feedback when the system switches between the consonants and the vowels would also be needed for better usage of the method.

6.2 Future Work

As all experiment participants mentioned, recognition errors occur when using the method. Since this is mainly caused by the recognition skill of Leap Motion, using other hand tracking devices would be needed as future work. In addition, doing the same experiment with left-handed participants would be needed to compare the difference of the usage between left-handed users and right-handed users.

References

1. HTC Corporation, Vive. <https://www.vive.com/jp/>. Accessed 7 Jan 2018
2. Itoh, K., Aoki, H., Hansen, J.P.: A comparative usability study of two Japanese Gaze typing systems. In: Proceedings of the 2006 symposium on Eye tracking research & applications, pp. 59–66 (2006)

3. InfoVision Incorporation, タイピング 速度測定 Ver 2.22. <http://sl.infovision.co.jp/sl/typing/default.html>
4. KITAI, VR comfortably enter text! VR keyboard appeared. <http://www.moguravr.com/punchkeyboard/>. Accessed 7 Jan 2018
5. Komiya, K., Nakajima, T.: A Japanese input method using leap motion in virtual reality. In: Proceedings of The Tenth International Conference on Mobile Computing and Ubiquitous Networking (2017)
6. Leap Motion Inc., Leap Motion. <https://www.leapmotion.com/>. Accessed 7 Jan 2018
7. Leap Motion Inc., Leap Motion Core Asset. <https://developer.leapmotion.com/unity#116>
8. Leap Motion Inc., Technology-Leap Motion. <https://www.leapmotion.com/product/vr/#11>. Accessed 7 Jan 2018
9. Nakagawa, Sony Interactive Entertainment Inc., PlayStation VR. <http://www.jp.playstation.com/psvr/>. Accessed 7 Jan 2018
10. Oculus VR, LLC., Oculus Rift/Oculus. <https://www.oculus.com/rift/>. Accessed 7 Jan 2018
11. Ravasz, J.: Punchkeyboard. <https://github.com/rjth/Punchkeyboard>. Accessed 7 Jan 2018