A 2-by-6-Button Japanese Software Keyboard for Tablets

Kei Takei and Hiroshi Hosobe
Faculty of Computer and Information Sciences, Hosei University, Tokyo, Japan

Keywords: Text Entry, Touch Screen, Touch Typing.

Abstract: Increasingly more people are using tablets. When they enter text, they usually use software keyboards. However, such software keyboards are not specialized in tablets. In this paper, we propose a software keyboard for tablets. Tailored to its user's use of the index, middle, and third fingers on a tablet screen, the keyboard consists of 2-by-6 buttons. It is focused on Japanese kana characters, and enables the user to enter a character usually with two strokes. We present the results of the experiment that we performed to evaluate the proposed keyboard by comparing it with a standard QWERTY software keyboard. The results indicate a trade-off between the average time lengths for entering a character and the average error rates for entering a word; the proposed keyboard resulted in a lower average error rate but in a longer average time length.

1 INTRODUCTION

Increasingly more people are using tablets. When they enter text with tablets, they usually use software keyboards (also known as virtual keyboards). However, such software keyboards are not specialized in tablets; they are typically simple modifications of software keyboards that have been used for personal computers and smartphones.

Software keyboards commonly used for tablets are based on the QWERTY keyboard. Since tablets have larger displays than smartphones, their software keyboards provide sufficiently large keys. However, even these software keyboards are not as easy to use as ordinary hardware keyboards. Since the software keyboards do not provide physical keys, they do not give clear tactile feedback. Therefore, the users often need to watch their hands and the tablet screens, which makes it difficult for them to do touch typing.

One effective approach to tackling this problem is to simplify the key layout by reducing the number of keys. This approach is adopted by Japanese software keyboards for smartphones. They mainly use a 4-by-3 grid of buttons, on which the users perform touch and flick operations, usually by using their thumbs, to choose characters. This 4-by-3-button software keyboard has become popular among Japanese smartphone users. However, this keyboard is not appropriate for tablets; the users often place tablets on desks, which makes it more difficult for them to perform touch and flick operations by using their thumbs. Therefore, many Japanese tablet users still use the QWERTY software keyboard.

Figure 1: Proposed 2-by-6-button keyboard.

In this paper, we propose a software keyboard for tablets. Tailored to its user's use of the index, middle, and third fingers on a tablet screen, the keyboard consists of 2-by-6 buttons as shown in Figure 1. It is focused on Japanese kana characters (specifically, 87 characters in total). It enables the user to enter a character usually with two strokes: the first stroke selects the group to which the character belongs; then the second stroke determines the character.

We also present the results of the experiment that we performed to evaluate the proposed keyboard. In this experiment, we conducted user study to compare it with the QWERTY software keyboard. The results indicate a trade-off between the average time lengths for entering a character and the average error rates for

Takei, K. and Hosobe, H.
A 2-by-6-Button Japanese Software Keyboard for Tablets.
DOI: 10.5220/0006652701470154
In *Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2018) - Volume 2: HUCAPP*, pages 147-154
ISBN: 978-989-758-288-2

entering a word; the proposed keyboard resulted in a lower average error rate, although it resulted in a longer average time length.

The rest of this paper is organized as follows. After presenting related work in Section 2, we briefly explain Japanese kana characters in Section 3. Next, in Section 4, we present the results of the preliminary experiments that we conducted to make the basic design of our software keyboard. In Section 5, we propose our Japanese software keyboard for tablets, and describe its implementation in Section 6. Then, in Section 7, we report the result of the experiment that we performed to evaluate our keyboard. After discussing our work in Section 8, we describe conclusions and future work in Section 9.

2 RELATED WORK

There has been research on problems with QWERTY software keyboards. (Hasegawa et al., 2012) experimentally compared a QWERTY software keyboard and a QWERTY hardware keyboard used for typing on a tablet, and showed that the software keyboard had resulted in a higher error rate. (Kim et al., 2014) reported that the QWERTY software keyboard was difficult to use because it caused chronic static loads to the users' shoulders.

Researchers have been developing alternatives to normal QWERTY software keyboards. (Findlater and Wobbrock, 2012) proposed personalizing the layout of a QWERTY software keyboard and integrating multi-touch gestures with the keyboard. (Fukatsu et al., 2013) and (Hakoda et al., 2013) proposed software keyboards for smartphones that enabled eyes-free Japanese kana input. (Hakoda et al., 2014) proposed a portrait-style QWERTY software keyboard for touch screen devices. (Sakurai and Masui, 2013) proposed a QWERTY software keyboard for tablets that enabled Japanese kana input by using flick operations. (Kuno et al., 2013) proposed a software keyboard called Leyboard that could change its key layout to get adapted to its user's hands. In some sense, these studies share the same motivation as our work. However, we explore a different direction; unlike these studies, we mainly investigate the simplification of the key layout for tablets, and also, with the aim of enabling touch typing, we do not adopt complex gestures but use normal touch operations for our software keyboard.

There also has been research on software keyboards tailored to smaller touch screens than tablets and smartphones. For example, (Shibata et al., 2016) proposed a software keyboard called DriftBoard for ultra-small touch screens like smartwatches. Such research suggests that software keyboards should be differently developed for different kinds of devices.

Apart from software keyboards, there has been much research on key layouts other than QWERTY (Noyes, 1983). For example, Dvorak is a popular alternative key layout. Also, there has been research on key layouts for Japanese text entry, e.g., (Shiratori and Obashi, 1987). However, to the authors' knowledge, such research was typically limited to hardware keyboards with similar physical structures to that of QWERTY, and there is no research that showed the effectiveness of such alternative layouts for software keyboards.

Handwritten character recognition has been long used for mobile devices including Japanese-capable tablets and smartphones (Zhu and Nakagawa, 2012). The use of this technology instead of or in combination with a software keyboard might be promising. However, this direction is presently beyond the scope of our research.

3 JAPANESE KANA CHARACTERS

We briefly describe Japanese *kana* characters, which we treat with our software keyboard. In general, Japanese text uses two kinds of characters, i.e., Chinese characters and kana characters. While a Chinese character typically has a meaning, a kana character does not; instead, a kana character is associated with a speech sound. There are two kinds of kana characters called *hiragana* and *katakana*. Although they are used for different purposes, they correspond to each other; for each hiragana character, there is a corresponding katakana character, and vice versa. Our keyboard treats only hiragana characters, and we refer to them just as kana below.

There are approximately 50 basic kana characters, which are further divided into 10 groups that are ordered, each of which typically consists of 5 characters. The first group is special because its 5 characters indicate 5 vowels that are pronounced "a," "i," "u," "e," and "o." The other 9 groups are associated with the basic consonants, "k," "s," "t," "n," "h," "m," "y," "r," and "w." A kana character in these 9 groups forms the sound that combines a consonant and a vowel. For example, the 5 characters of the "k" group are pronounced "ka," "ki," "ku," "ke," and "ko." This grouping of kana characters is basic knowledge of the Japanese language.

The "k," "s," "t," and "h" groups have variants called *dakuon*. Specifically, the dakuon variants of "k," "s," "t," and "h" are "g," "z," "d," and "b" respectively. In addition, the "h" group has another variant called *handakuon*, which is "p." Certain characters have variants that are written in smaller shapes.

Sequences of kana characters can be expressed with the Roman alphabet by using the standard Japanese romanization system (ISO, 1989). This is widely used for computer users to enter Japanese text with alphabet keyboards such as QWERTY.

4 PRELIMINARY EXPERIMENTS

We report the preliminary experiments that we performed before designing our keyboard.

4.1 Overview

Our Japanese software keyboard for tablets aims at enabling users to do touch typing. For this purpose, users need to easily keep the home position without tactile feedback. To make the basic design of this keyboard, we first conducted two preliminary experiments that examined how the numbers and the layouts of buttons on the touch screen affected user performance. The difference of the two preliminary experiments is the visibility of the buttons shown on the screen; the buttons were visible to the participants in the first experiment, but were hidden in the second experiment. To examine user performance, we investigated the speed and the accuracy of users' text entry because these are important and measurable performance criteria.

We developed an iOS application in Objective-C, and used an iPad Air tablet equipped with a 9.7-inch display. The screen of the application displays buttons below and a question panel above, and provides 8 layouts of buttons, i.e., 2-by-2, 2-by-4, 2-by-6, 2-by-8, 2-by-10, 3-by-2, 3-by-4, and 3-by-6 as shown in Figure 2. During the experiments, the tablet was placed on a desk with its shorter sides directed forward, and the participants sat on a chair.

These experiments are similar to those done by (Fukatsu et al., 2013). However, we needed to perform different experiments because our software keyboard is different in the target device, the posture of users, and the level of touch typing.

4.2 Using Visible Buttons

In the first preliminary experiment, we examined how different numbers and layouts of buttons would affect user performance when buttons were visible. We asked the participants to touch a button corresponding to the location on the question panel that got highlighted. Such highlighted locations were chosen randomly. We formed one set of the experiment from a sequence of 20 occurrences of highlighting.

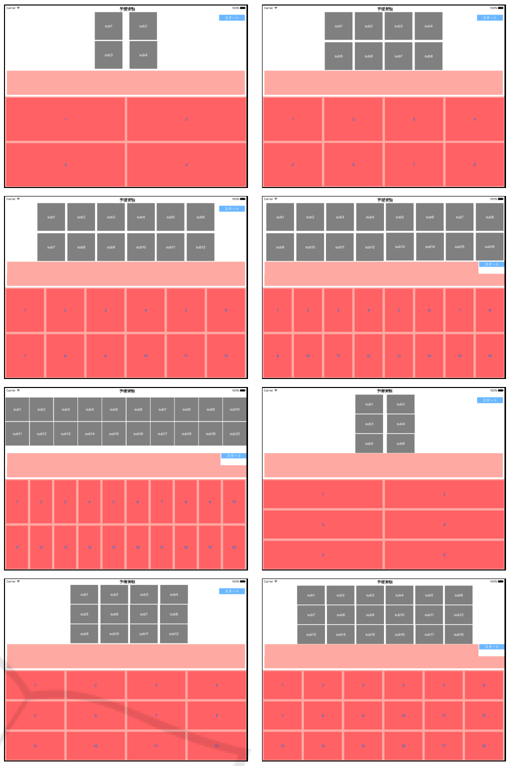

Figure 2: Application for the preliminary experiments.

We recruited 6 participants ranging from 20 to 55 years in age. All of them were right-handed, and they had the experience of using touch screens for 4.0 years on average. Each participant performed 5 sets of the experiment for each button layout (i.e., $8 \times 5 \times 20 = 800$ touches in total).

We measured average time lengths and average error rates. As already mentioned, the speed and the accuracy of users' text entry are important performance criteria. A shorter average time length indicates higher speed, and a lower average error rate indicates higher accuracy. The average time length for a layout is the average time from a touch to the next touch. The average error rate for a layout was the average division of the number of errors by the total number of touches. The first set of the experiment among the 5 sets for a layout was regarded as a practice, and therefore was excluded from the calculation of the average time length and the average error rate, about which we did not inform the participants. The first touch of one set of the experiment was not used to calculate the average time length.

4.3 Using Hidden Buttons

In the second preliminary experiment, we examined how different numbers and layouts of buttons would affect user performance when buttons were hidden from participants. For this purpose, the lower half of the screen was covered with a plastic box. The participants performed touch inputs similar to those in the first preliminary experiment, without seeing their hands and the lower half of the screen.

We recruited 5 participants ranging from 20 to 55 years in age. All of them were right-handed, and they had the experience of using touch screens for 4.2 years on average. All of them had participated also in the first preliminary experiment. This experiment measured average time lengths and average error rates in the same way as the first preliminary experiment.

4.4 Results

Figures 3(a) and 3(b) show the average time lengths and the average error rates respectively that we obtained in the preliminary experiments. Whether the buttons were visible did not cause large differences in the average time lengths. In both cases, they gradually increased as the numbers of buttons increased.

By contrast, the visibility of the buttons caused large differences in the average error rates, which suggests that the participants largely depended on their sights when typing on the touch screen. Especially, when there were 16 buttons or more, the average error rates were large.

(a)
(b)

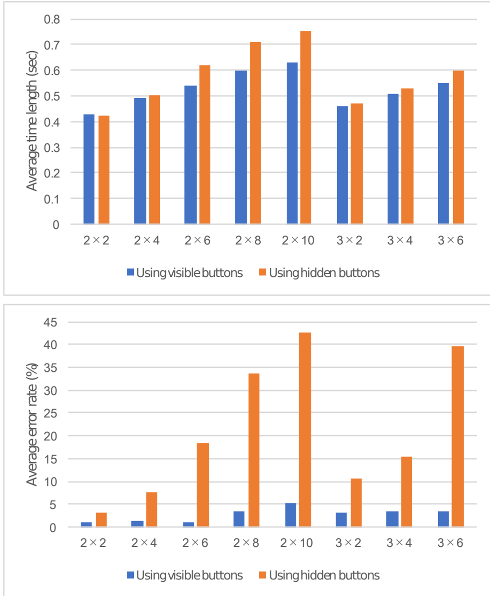

Figure 3: (a) Average time lengths and (b) average error rates resulted from the preliminary experiments.

5 PROPOSED KEYBOARD

We propose a Japanese software keyboard for tablets.

5.1 Basic Design

We design a new Japanese software keyboard for tablets as an alternative to the QWERTY software keyboard. For this purpose, we exploit the results of the preliminary experiments shown in the previous section. Since the visibility of the buttons did not cause large differences in average time lengths, we emphasize average error rates.

We design our keyboard by using normal touch operations. Since there are approximately 50 basic kana characters, it is basically better to choose a layout with more buttons. In particular, the 2-by-6-button layout is noticeable; it exhibited excellent user performance when the buttons were visible, and also it resulted in the average error rate of less than 20 % when the buttons were hidden. Therefore, we use this layout to design our keyboard. The user of this keyboard is supposed to use the index, middle, and third fingers of both hands, each of which is assigned to two vertically laid-out buttons. This allows the user to type without moving their arms, and thus the user can keep the home position on the touch screen.

We use two strokes for one basic kana character. The first stroke is used to select one from the 10 groups of basic kana characters, and the second stroke is used to select the basic kana character. As shown in Figure 1, the 2-by-5 white buttons on the left are assigned to this task, and the 2-by-1 gray buttons on the right are assigned to the special keys called “back” and “other.”

5.2 Entering Characters

We now describe details about how to enter characters. The current design of our keyboard supports 85 graphic characters (i.e., 81 kana characters, 3 Japanese special characters, and “space”) and 2 control characters (“backspace” and “enter”). The 80 characters shown in the “Upper” and “Lower 1” columns in Table 1 (which include “enter”), called *2-stroke characters* below, can be entered with two strokes. The 6 characters shown in the “Lower 2” column in Table 1, called *3-stroke characters* below, can be entered with three strokes. The remaining character, “backspace,” can be entered with one stroke using the “back” button.

Table 1: Supported characters (except “backspace”).

Group	Upper	Lower 1	Lower 2
あ	あいうえお	ぁぃぅぇぉ	
か	かきくけこ	がぎぐげご	
さ	さしすせそ	ざじずぜぞ	
た	たちつてと	だぢづでど	っ
な	なにぬねの		
は	はひふへほ	ばびぶべぼ	ぱぴぷぺぽ
ま	まみむめも		
や	や ゆ よ	ゃ ゅ ょ	
ら	らりるれろ		
わ	わ を ん	ゎ	
other	enter 、。ー space		

To enter a character X, the user performs the following operations.

1. If X is “backspace,” the user touches the “back” button.
2. Otherwise, the user touches the button corresponding to the group of X, and next does the following.
   (a) If X is a 2-stroke character, the user touches the button corresponding to X.
   (b) Otherwise (i.e., X is a 3-stroke character), the user touches the “other” button and then the button corresponding to X.

Right after the user performs the first stroke for a group, the upper and the lower buttons will be changed to the “Upper” and the “Lower 1” characters in the corresponding row of Table 1. When the user further touches the “other” key, the lower buttons will be changed to the “Lower 2” characters.

If the user makes a mistake of touching a wrong button for the first stroke of a 2- or 3-stroke character, the user can cancel this first stroke by touching the “back” button. If the user makes a mistake of touching the “other” button for a 2-stroke character, the user can cancel this second stroke by touching the “other” button again.[1]

5.3 Examples

Let us show three examples of entering characters. The first example is to enter the kana character corresponding to “ku.” Since it belongs to the “k” group, the user first touches the button corresponding to this group (which is indicated by the red circle in Figure 4(a1)), which then displays the “ku” character on an upper button (Figure 4(a2)). Touching this button, the user finishes the entry of “ku.”

The next example is to enter the 3-stroke character corresponding to “po.” Since it belongs to a variant of the “h” group, the user first touches the button corresponding to this group (Figure 4(b1)). Since it is not shown on the buttons (Figure 4(b2)), the user touches the “other” button, which then displays the “po” character on a lower button (Figure 4(b3)). Touching this button, the user finishes the entry of “po.”

The final example is to input the control character “enter.” Since it is a special character that belongs to the “other” group, the user first touches the button corresponding to this group (Figure 4(c1)), which then displays “enter” on an upper button (Figure 4(c2)). Touching this button, the user finishes the input of “enter.”

6 IMPLEMENTATION

We implemented the proposed keyboard (Figure 1). To evaluate this, we also implemented a QWERTY software keyboard (Figure 5). We implemented them in Objective-C as part of an iOS application for iPad Air. We made the QWERTY keyboard almost the same as the standard one equipped with iPad Air; one exception was that the user always needed to touch “n” twice to enter the kana character indicating “n.”[2]

7 EXPERIMENT

We report the experimental comparison of the proposed keyboard with the QWERTY software keyboard.

7.1 Procedure

We recruited 6 participants ranging from 20 to 22 in age. They had the experience of using touch screens for 4.3 years on average. During the experiment, they sat on a chair, and a tablet was placed on a desk with its shorter sides directed forward.

The participants were asked to do the following: enter 7-character words that would be displayed on the upper part of the screen, touch the “enter” button

[1] More precisely, if the character is the one that appears in the “Upper” column of Table 1, the user does not need to do the cancellation since it is still shown on an upper button.

[2] Although touching “n” twice is a common way of entering the kana character indicating “n,” the QWERTY keyboard equipped with iPad Air additionally provides another way of entering this character.

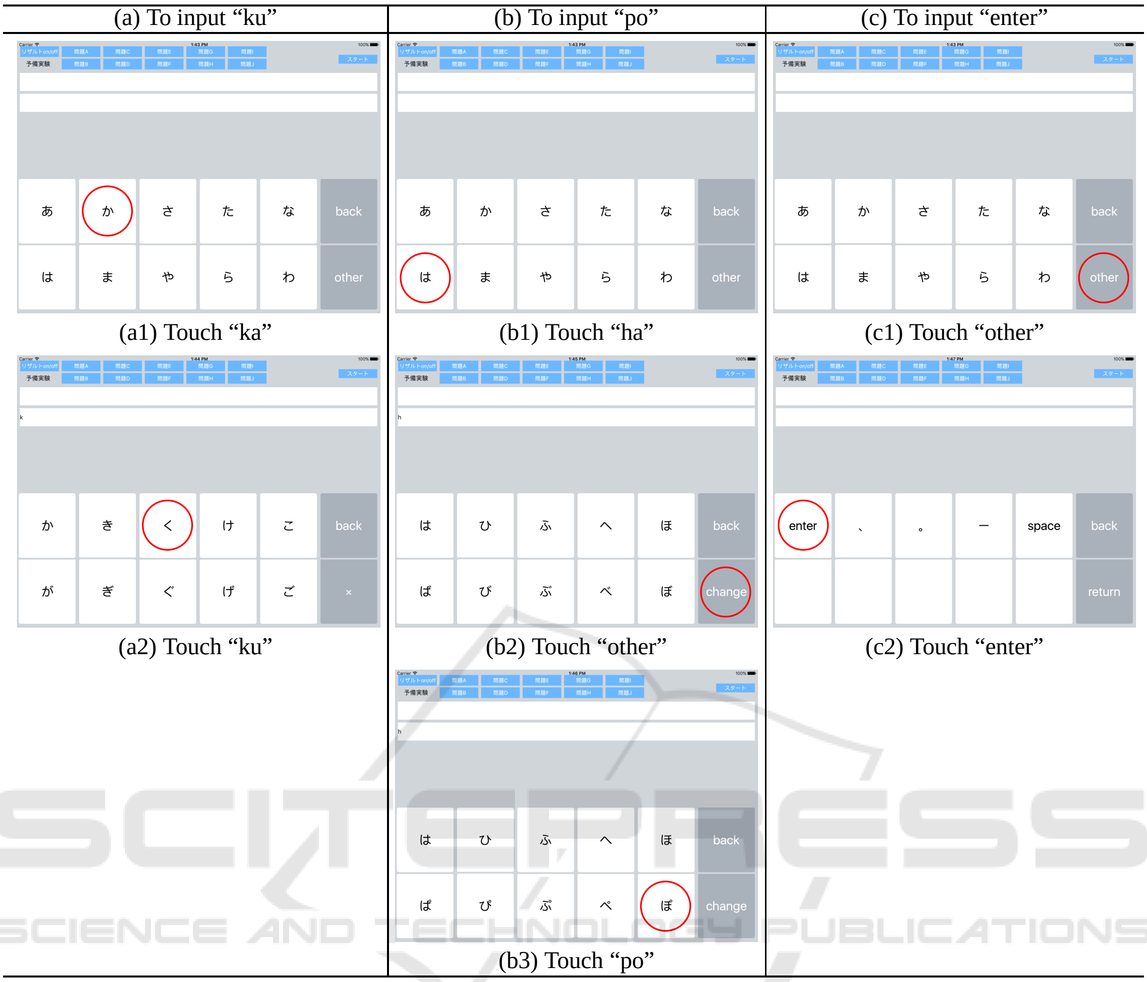

Figure 4: Examples of entering characters with the proposed keyboard.

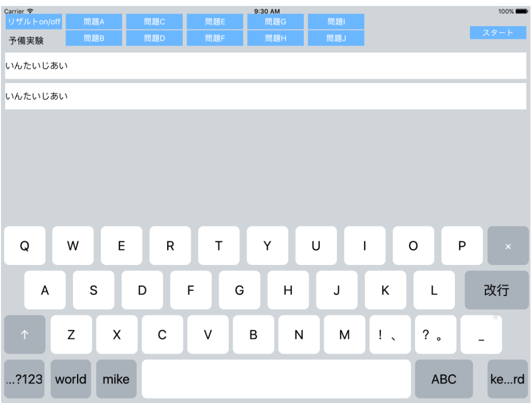

Figure 5: QWERTY keyboard used for comparison.

to move to a next word, and repeat this again. Such words were selected randomly from the sets of words prepared beforehand, each of which consisted of 20 words. Each participant entered 2 sets of words with each keyboard, i.e., 4 sets in total.

The participants were divided into 2 groups: each participant of one group first entered 2 sets *A* of words with the QWERTY keyboard, and next entered other 2 sets *B* of words with the proposed keyboard (i.e., 80 words in total); each participant of the other group first entered *A* with the proposed keyboard, and next *B* with the QWERTY keyboard. The participant's entry of the first set of words with each keyboard was for practice, and was not included in the measurement.

We measured average time lengths for kana characters and average error rates for words. The entries of the “enter” button were included. The entry of the first character of each set of words was not included. In the case of the QWERTY keyboard, a sequence of 2 kana characters for entering *yōon* or *sokuon* (that uses a smaller-shape character) was treated as 1 kana character because it was expressed in a specific way by the standard Japanese romanization system (ISO, 1989). Average error rates were computed in the same way as (Hakoda et al., 2013), which is as follows:

Table 2: Results of the experiment.

	Average time length (sec)	Average error rate (%)	Average # of times "backspace" was used
2-by-6-button	1.78	0.6	8.3
QWERTY	0.78	2.3	14.0

compare the character sequence entered by a participant with the word presented on the screen; count the total of wrong characters, unnecessary characters, and missing characters; compute the average error rate as the average division of this total by the number of the characters in the presented word.

We also collected questionnaires from 17 persons consisting of the 6 above-mentioned participants and other 11 persons who participated in a pilot experiment that we had performed before this final experiment. We performed it in the same way as (Hakoda et al., 2013) by partly using the same questions. The questionnaire consisted of the 5 questions asking the following: (1) the easiness of learning how to enter characters; (2) the easiness of touch typing; (3) promisingness—how easy it seems to use when the user gets accustomed; (4) good points; (5) points that need to be improved. The questions (1), (2), and (3) were quantitative ones asked by using 5-level Likert scales. The questions (4) and (5) were qualitative ones to which the participants answered in sentences.

7.2 Results

Table 2 shows the results of the experiment about the average time lengths and the average error rates together with the average numbers of times when "backspace" was used for one set of words. The results indicate a trade-off between the average time lengths and the average error rates; the proposed keyboard resulted in both a lower average error rate and a smaller average number of times when "backspace" was used than the QWERTY keyboard, although the proposed keyboard resulted in a longer average time length.

Table 3 shows the results of the questionnaires about questions (1)–(3). The easiness of learning and the promisingness were agreed to by many participants, although the easiness of touch typing resulted in rather low evaluation.

Concerning questions (4) and (5), there were positive answers about the large size of the buttons (4 participants) and the small number of the buttons (3 participants), although there were negative answers especially about the difficulty of entering special characters such as "enter" (6 participants) and the difficulty of entering 3-stroke characters (2 participants).

Table 3: Results of the questionnaires.

	(1) Easiness of learning	(2) Easiness of touch typing	(3) Promisingness
5: Strongly agree	3	0	3
4: Agree	8	3	7
3: Neither agree nor disagree	4	3	3
2: Disagree	2	8	2
1: Strongly disagree	0	3	2
Average	3.7	2.4	3.4

8 DISCUSSION

The proposed keyboard resulted in a lower error rate than the QWERTY software keyboard. We think that this was mainly because of the large size and the small number of the buttons of the proposed keyboard, but also because it changed the screen whenever the user touched a button, which worked as clear feedback.

The proposed keyboard resulted in a longer average time length. We think that, as the results of the questionnaires showed, this was especially because of the difficulty of entering special and 3-stroke characters.

The proposed keyboard does not support Chinese characters. To support them, we need to add a facility of converting kana to Chinese characters. This would not be easy because such a facility requires the user's selection of an appropriate conversion from a candidate list; this typically needs the user to move the user's fingers from the home position, which would degrade the advantage of the proposed keyboard.

9 CONCLUSIONS AND FUTURE WORK

We proposed a 2-by-6-button software keyboard for tablets. It enabled its user to enter a Japanese kana character usually with two strokes. We also experimentally compared it with a standard QWERTY software keyboard. It resulted in a lower average error rate but in a longer average time length.

To improve the performance of the proposed keyboard, we need to solve the current problems with the entry of special and 3-stroke characters. For this purpose, we are exploring the possibility of introducing a limited form of flick operations by still keeping the low error rate of the proposed keyboard.

ACKNOWLEDGEMENT

This work was partly supported by JSPS KAKENHI Grant Number JP15KK0016.

REFERENCES

Findlater, L. and Wobbrock, J. O. (2012). From plastic to pixels: In search of touch-typing touchscreen keyboards. *Interactions*, 19(3):44–49.

Fukatsu, Y., Shizuki, B., and Tanaka, J. (2013). No-look Flick: Single-handed and eyes-free Japanese text input system on touch screens of mobile devices. In *Proc. MobileHCI*, pages 161–170. ACM.

Hakoda, H., Fukatsu, Y., Shizuki, B., and Tanaka, J. (2013). An eyes-free kana input method using two fingers for touch-panel devices. *IPSJ SIG Tech. Rep.*, 2013-HCI-154(6):1–8. In Japanese.

Hakoda, H., Shizuki, B., and Tanaka, J. (2014). Evaluating the performance of a portrait-style QWERTY keyboard for touch-panel devices. *IPSJ SIG Tech. Rep.*, 2014-HCI-156(18):1–7. In Japanese.

Hasegawa, A., Hasegawa, S., and Miyao, M. (2012). Characteristics of the input on software keyboard of tablet devices: Aging effects and differences between the dominant and non-dominant hands for input. *J. Mobile Interact.*, 2(1):23–28. In Japanese.

ISO (1989). Documentation—Romanization of Japanese (kana script). ISO 3602:1989.

Kim, J. H., Aulck, L., Bartha, M. C., Harper, C. A., and Johnson, P. W. (2014). Differences in typing forces, muscle activity, comfort, and typing performance among virtual, notebook, and desktop keyboards. *Applied Ergon.*, 45(6):1406–1413.

Kuno, Y., Shizuki, B., and Tanaka, J. (2013). Long-term study of a software keyboard that places keys at positions of fingers and their surroundings. In *Proc. HCII*, volume 8008 of *LNCS*, pages 72–81.

Noyes, J. (1983). The QWERTY keyboard: A review. *Int. J. Man-Machine Studies*, 18(3):265–281.

Sakurai, Y. and Masui, T. (2013). A flick-based Japanese input system for a QWERTY software keyboard. *IPSJ SIG Tech. Rep.*, 2013-HCI-154(5):1–4. In Japanese.

Shibata, T., Afergan, D., Kong, D., and Yuksel, B. F. (2016). DriftBoard: A panning-based text entry technique for ultra-small touchscreens. In *Proc. ACM UIST*, pages 575–582.

Shiratori, Y. and Obashi, F. (1987). Design of a key layout for the Roman alphabet based Japanese input system and its evaluation. *IPSJ J.*, 28(6):658–668. In Japanese.

Zhu, B. and Nakagawa, M. (2012). Recent trends in online handwritten character recognition. *J. IEICE*, 95(4):335–340. In Japanese.