Micro Touch Board Specially Designed for SliT that Is the Japanese Character Input Method for Smartwatches

Toshimitsu Tanaka(✉), Koutaro Saka, Kohei Akita, and Yuji Sagawa

Department of Information Engineering, Faculty of Science and Technology, Meijo University, Shiogamaguchi 1-501, Tenpaku-ku, Nagoya 468-8502, Japan
toshitnk@meijo-u.ac.jp

Abstract. We developed a touch board to solve the problem of smartwatches that screen is hidden by the finger during character input. It is used attached to the watchband under the screen. The board is optimized for the character input method named SliT (***Sl**ide-**i**n and **T**ap method*). Advantage of SliT is that the input speed of the novice is fast and the screen occupancy rate is low. Specifically, the speed is 28.7 [CPM (Characters Per Minute)] and the rate is 26.4%.

In SliT, Japanese hiragana characters are input by combining 12 kinds of strokes and 8 tapping points. The board is designed to detect only those strokes and the taps. Eight touch sensors are arranged so as to surround the circular area placed at the center of the touch board. Between the sensors, numbers from 1 to 8 are allocated clockwise. Only when two adjacent sensors are touched at the same time, the number assigned to their boundary is entered. In stroke input, a fingertip is slid until passing through three boundaries, and three continuous numbers are entered. However, in rare cases, the last number may be lost. So, the stroke is judged from the first two numbers. The stroke recognition rate by this method is 98.7%. In tap input, users touch only one boundary. However, in a very few cases, an unnecessary number is added. In this case, the tap is judged from the first number. The tap recognition rate by this method is 99.4%.

Keywords: Micro touch board · Touch sensor · Smartwatch · Character input device · Japanese language

1 Introduction

The flick keyboard and the software QWERTY keyboard are standard methods for entering characters to smartphones. However, the smartwatch screen is too small to display the QWERTY keyboard. Although it is possible to display the flick keyboard, it occupies most of the screen, so the area for displaying entered characters is very small. In addition, the flick keyboard places more than 20 keys on a small screen. Therefore, each key is too small compared to the fingertip. As a result, the key adjacent to the target key is often unintentionally entered. This is called fat finger error [1].

In order to solve these problems, the character input method SliT (***Sl**ide-**i**n and **T**ap method*) [2] was developed. The main purpose of SliT is to reduce screen occupancy in input wait state. Its screen occupancy rate is only 26.4%, which is the lowest among

M. Kurosu (Ed.): HCII 2019, LNCS 11567, pp. 49–61, 2019.
https://doi.org/10.1007/978-3-030-22643-5_5

competitive methods. However, since it is a method of inputting characters by touching the screen, it is inevitable that the finger hides the screen during the operation. To solve this problem completely, an input device independent of the screen is necessary. Therefore, we developed a micro touch board specially designed for SliT. This board is attached to the watchband under the screen. Its touch sensors detect the strokes and taps necessary for SliT operation. This paper introduces the design and performance of the touch board.

2 Related Researches

Many devices have been developed for entering characters under mobile environment.

By using the glove type device [3] developed by Fujitsu Ltd. or the ring type device [4] such as Ring ZERO [5], it is possible to enter characters by drawing them in the air. However, since it is difficult to draw complicated characters like kanji correctly in 3D space, the input error rate is high. In addition, as the arm must be moved widely, the input speed is slow.

There is a method of using the user's hands and arms as a touch pad or a virtual keyboard. In one such method, SkinTrack [6], it is necessary to touch the naked skin with a finger wearing a ring-shaped transmitter. This is because to detect the position of the finger with high frequency current flowing on the surface of the skin. ARmKeypad [7] projects the keyboard onto user's arm using AR technology. Therefore, it is necessary to wear a see-through type HMD. In FingersT9 [8] and DigiTouch [9], it is necessary to attach one touch sensor to each node of the fingers in order to use the user's fingers as a keyboard. Finger tactile [10] also uses a finger as a keyboard, but this projects the virtual keyboard from the device attached to the user's wrist onto the hand. Haier's Asu Smartwatch [12] builds a projector in to project a virtual keyboard onto the back of the hand. With these two methods, if the angle of the wrist is not properly maintained, the shape of the projected keyboard will be distorted.

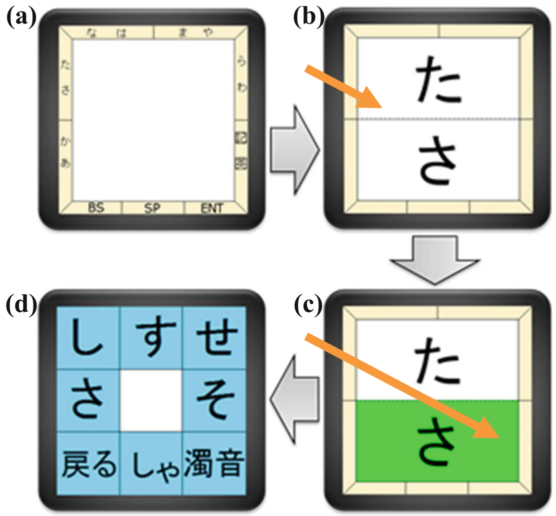
TAP [11] is a plate-shaped device to be used with five fingers. A character is selected by a combination of fingers to be tapped. Therefore, it is necessary to learn patterns assigned to each character. In Float [13], one character is selected by combination of wrist angle measured with acceleration sensor built in smart watch. In WristWhirl [14], one letter is selected by the movement of the wrist. In WrisText [15], the key of the One Line keyboard [16] is selected by the angle of the wrist measured by the sensor attached to the watch band. In every method, it is necessary to learn the wrist angle and/or the wrist motion assigned to each character.

Speech recognition technology allows the user to enter sentences by speaking them to a smartwatch. However, it is difficult to correct wrong input by voice only. In addition, there are disadvantages that the people around can listen to the utterance.

3 Character Input Method SliT

The touch board proposed in this research is optimized to the character input method SliT. The first feature of SliT is that the screen occupancy rate is low. To leave a large area for displaying sentences, the keys of the SliT are allocated on narrow area on the surrounding the screen. The second feature is that beginners will be able to enter characters in a short time. A smartwatch is usually linked to a smartphone or a tablet. In this circumstance, the smartwatch will be assumed to use to enter a few short messages, because the large screen of the smartphone or the tablet is convenient for entering long sentences. Therefore, it is designed to be able to enter characters quickly even if there are few opportunities to use. A summary of SliT is shown below.

In SliT, one Japanese hiragana character is entered by choosing a Gyou (the rows of the table of Japanese syllabary), then choosing a Dan (the columns of the table). Figure 1(a) shows the initial display of the user interface. Each of the left edge, the upper edge, and the right edge of the screen is divided into two segments, and two Gyous are assigned to each segment.



(a) This is the initial display. The yellow narrow area is the keyboard. One segment is selected with slide-in.

(b) When the finger crosses the keyboard, the screen is split and the names of Gyou assigned to the intersected segment are displayed one by one.

(c) When the finger enters either area, its background turns green. The Gyou written in the area is selected by releasing the finger.

(d) The characters belonging to the selected Gyou is displayed. One is selected by tapping.

Fig. 1. Changes in screen display of SliT during entering a hiragana character. (Color figure online)

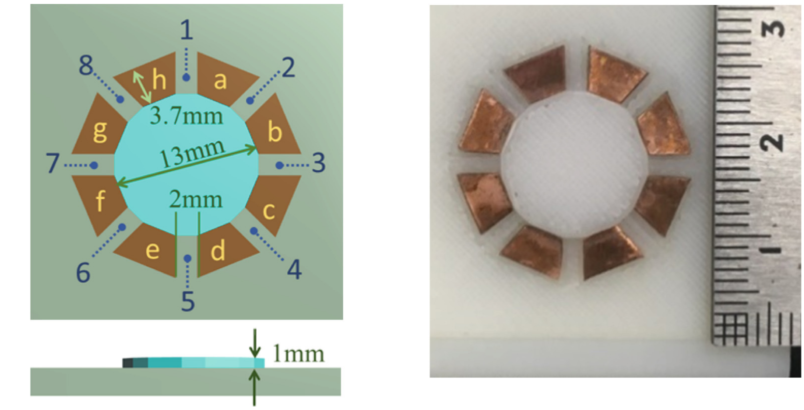
One Gyou is selected with one slide-in. Slide-in is an operation of sliding a finger touched outside the screen toward the inside. When the slide-in passes through a segment, the screen is split into two areas as shown in Fig. 1(b). Two Gyous assigned to the intersected segment are indicated one for each area. When the finger moves into one of the areas, its background turns green as shown in Fig. 1(c). By leaving the finger from the screen, the Gyou written in the area is selected. Subsequently, as shown in

Fig. 1(d), five hiragana characters belonging to the Gyou are displayed, so one is select by tapping. It is possible to select voiced, semi-voiced, and lowercase characters by tapping the bottom right button and changing the display.

The screen share of the keyboard in Fig. 1(a) is 26.4%. It is lower than the competitive software keyboards for smartwatches such as 5-TILES [17], TouchOne Keyboard [18], and ZoomBoard [19]. In the experiment, the speed after inputting 125 characters from the beginning of use was 28.7 [CPM (Characters Per Minute)]. Since this speed is reached in 320-s usage on average, it can be said that inputs at practical speeds are possible even for short-time use. The rate of erroneous input was 4.7%.

4 Design of the Touch Board

Figure 2(a) is a design drawing of the touch board. Eight brown trapezoids from “a” to “h” are touch sensors. They are equally spaced around the circular area placed at the center of the touch board. When two adjacent sensors are touched at the same time, the number assigned between them is output. So that, it is possible to judge the finger position from the output value. Figure 2(b) is a photo of the touch board. Eight touch sensors are made by cutting a thin copper plate. They are fitted in a board made with a 3D printer. The material of the board is ABS (acrylonitrile butadiene styrene) copolymer.



(a) Sensor layout and number assignment (b) Prototype of the touch board

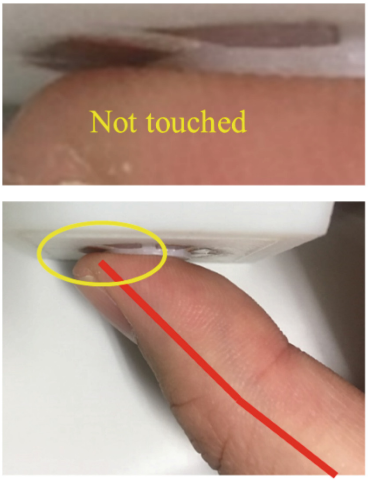
Fig. 2. Design of the touch board.

In Gyou selection of SliT, 12 kinds of slide-in (2 directions from each of 6 segments) are used. Each is specified by a sequence of three positions where the fingertip passes. For example, “た” Gyou is selected with a stroke from upper left to upper right, which is the motion from the sensor “g” to “b” passing “h” and “a”. When a finger passes the border of two sensors, the number assigned between them is output, because the finger contacts with both sensors at the same time. Using such numbers, the stroke of "た" Gyou is defined as sequence 8, 1, 2. Other strokes are defined as the sequences shown in Table 1. In Dan selection of SliT, the eight keys displayed in light blue in Fig. 1(d) are used. Those keys are replaced by the positions 1 to 8 shown in Fig. 2(a). Each of which is selected by tapping.

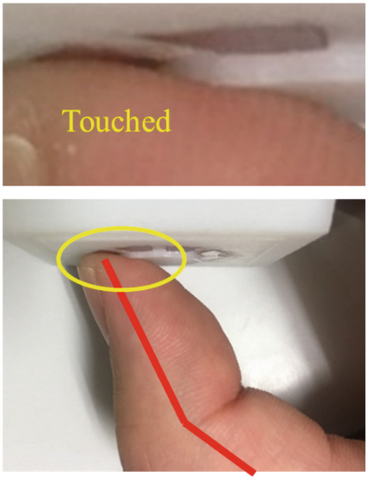
Table 1. Definition of the order of three positions for selecting each Gyou.

Gyou	あ	か	さ	た	な	Symbol
Sequence	6, 5, 4	7, 8, 1	7, 6, 5	8, 1, 2	8, 7, 6	3, 2, 1
Gyou	は	ま	や	ら	わ	Mode change
Sequence	1, 2, 3	1, 8, 7	2, 3, 4	2, 1, 8	3, 4, 5	4, 5, 6

The circular area at the center of the board in Fig. 2 is intended to avoid touching extra sensors. This area is elevated 1 [mm] from the sensors. As shown in Fig. 3(a), if the angle of the finger to the touch board is small, the fingertip can not reach the sensor. However, as shown in Fig. 3(b), by placing the finger at an angle close to the vertical to the board, the fingertip touches the board. This angle reduces the contact area of the fingertip, so reducing input errors due to touching surrounding sensors.



(a) The case that the angle of the finger to the touch board is small.



(b) The case that the finger close to the vertical to the board.

Fig. 3. The angle of the finger to the touch board.

The state of each sensor is detected using Arduino Uno, and transmitted to the host every millisecond. When the finger touches two adjacent sensors, Arduino outputs numbers from 1 to 8. Otherwise, it outputs 0. While the finger is moving, it may float from the board momentarily and/or touch only one sensor for a short time. Thus, zeros inserted in the numbers to be send. For example, in the stroke of “た” Gyou, zeros are inserted between the series of 8, 1 and 2, as shown in Fig. 4(a). So that, short zero sections are deleted. In practice, consecutive zeros less than 100 in length are deleted. This processing removes the red zero in Fig. 4(b) and corrects the output to Fig. 4(c). After that, by consolidating consecutive same numbers into one, the output will be the sequence of 0, 8, 1, 2, 0. The first and last zeros indicate that the finger is away the touch board.

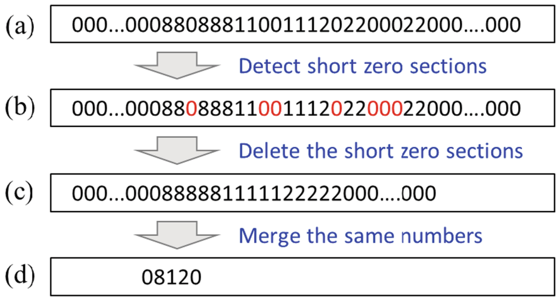


Fig. 4. Signal processing of the touch board. (Color figure online)

If the entered numbers are not consecutive (under the definition that the next number of 8 is 1) and the difference is 2, complement the numbers between them. For example, sequence 1, 3 is corrected to 1, 2, 3.

If the position to release the finger shifts in the stroke input, the last position of the sequence will be gone or the next position will be added. Similarly, if the position to start the stroke shifts, the first position disappears or the previous position is inserted. In order to correct such sequences, two methods were conceivable. One method judges a stroke with only two positions from the first. Another determines from the last two positions. Which is better is evaluated in the experiment.

5 Evaluation Experiment

5.1 Experimental Procedure

For this experiment, the smartwatch shown in Fig. 5 was used. The band of it was created with a 3D printer, and our touch board and SONY Smartwatch 3 is embedded. The subject wears it on the wrist and operates it with the thumb of the dominant hand. The position of the fingers and hands other than the thumb was free as long as they did not touch the board.

Prior to the experiment, how to use the touch board and the experiment procedure are explained in about 5 min in total. Next, the subject experiences to enter the strokes and learns the angle needed to touch the thumb to sensors. Here, one task of the experiment is executed as practice. When all stokes of the task are correctly entered, the experiment is started. If incorrect input occurs, after briefly teaching about the angle of

Fig. 5. Equipment used for the experiments (the left) and a sample of animation displayed to control stroke input speed (the right).

the thumb necessary to touch the sensors, an additional task is executed and this verification process is repeated. However, five minutes after the start of the practice, the experiment is started regardless of the status. For the experiment entering taps, "stroke" in this procedure is changed to "tap".

In the stroke experiment, the subject enters the 12 kind strokes, which were shown in Table 1. Each stroke is a movement through three consecutive positions. In one task, each of 12 kinds is entered once in random order. This task is repeated 15 times with a 10 [s] break.

According to the research by Akita et al. [2], a person skilled in SliT can enter one character at about 1 [s]. Since one character is specified by a pair of one stroke and one tap, the time spent for the stroke is considered to be about 0.5 [s].

Therefore, we induced the subjects to execute strokes at this speed using animation. As shown in Fig. 5, a blue dot is displayed on the screen. The dot moves from the start position of the stroke to the end at 0.5 [s] in constant velocity. We asked the subjects to track the dot with their thumbs. However, even if the speed of the fingertip differs from the speed of the dot, input is accepted.

In the tap experiment, each of the eight positions is entered once per each task. This task is repeated 15 time with a 10 [s] break. The order of the touch positions is randomly changed each task. In this experiment, the subjects tap the blue dot immediately after it is displayed. The position of the fingertip is judged by output from the board up to 0.5 [s] after the dot is displayed.

We conducted this experiment with 10 subjects. Their age is 20 to 24 years old. All subjects are right-handed. They have never used our touch board before this experiment. In all tasks, subjects are not notified whether the input is correct or not.

5.2 Stroke Input

In Table 2, the row entitled "Complete" shows the rate that the strokes are completely entered, that is, the three positions of each stroke are correctly input without excess or deficiency. The value at the right end of the table is the average of the 10 subjects. The rate at which a complete stroke is input is 94.7%.

Table 2. Percentage of the strokes recognized correctly.

Subject	A	B	C	D	E	F	G	H	I	J	av.
Complete (%)	96.7	97.8	91.1	98.3	93.9	86.1	95.0	96.1	94.4	97.2	94.7
First two (%)	**98.9**	**99.9**	**97.8**	**100**	**96.1**	**97.8**	**100**	**98.9**	**97.8**	**100**	**98.7**
Last two (%)	97.2	98.3	92.8	98.3	**96.1**	87.8	95.0	97.2	95.6	97.2	95.6

* The "Complete" means all three positions of the stroke ware entered without excess or deficiency. The "First two" mean that at least 2 positions from the first is correct. The "Last two" means that at least 2 positions from the last is correct. Therefore, they include the complete strokes.

The row of "First two" shows the rate that both the first and the second positions of each stroke were correctly entered. This includes the case where the third position is lost or where the fourth position is added, in addition the complete strokes. In opposition, the "Last two" row shows the rate that two or more positions from the last are correct. About all subjects in Table 2, the value of the "First two" row is higher than or equal to that of "Last two". By the calculation of the t-test using the values of those two rows, $t = 2.18$ and $P = 0.013$. This P value ensures with a significance level of 5% that stroke judgement with the first two positions is more reliable than that with the last two. It is the same judgment result in Mann-Whitney U test. By judging from the first two positions, the average input rate rises to 98.7%. This rate is sufficiently high.

Of the total 1,800 trials, 96 incomplete inputs occurred. Table 3 shows the five strokes with many incomplete inputs. Since inputs lost the first position of the stroke are fewer than the input lost the last position, it is more reliable to judge using the first positions in the input sequence.

Table 3. Number of incomplete strokes.

Sequence	1,8,7	1,2,3	3,4,5	2,1,8	8,1,2	Others	Total
Lost the first position	0	1	**8**	0	**7**	0	16
Lost the last position	**15**	**18**	**8**	**11**	0	**9**	**61**
Others	9	1	0	1	4	4	19
Total	24	20	16	12	11	13	96

Almost half of the incorrect inputs occurred on two strokes starting at position 1. In these strokes, the thumb moves from the center top of the board to the middle of the right end or left end. In our touch board, it is necessary to move the thumb along the curved edge of the center raised area. Because the speed increases in the movement from top to bottom, it is expected that the thumb may be off from the sensor at the last position.

The stroke 3, 4, 5 is performed at the lower right part of the board. Since the sensors must be touched by bending the thumb, position control may be difficult. The last position of the stroke 2, 1, 8 is 8 and the first position of 8, 1, 2 is also 8. That is, mistakes occurred at the position 8. To touch that position, the thumb must be extended. The angle of the thumb to the board decreases. The belly of the thumb contacts the lifted area. As the result, the contact point shifts out of the sensors.

5.3 Tap Input

Table 4 shows the percentage of taps. The "Complete tap" means that only the correct position is entered. The average is 97% that is higher than that of strokes shown in Table 2. The "First is correct" row indicates the percentage of the case at least the first position is correct. This case includes both when only the correct position is entered and when a position is added after the correct position. The "Last is correct" shows the percentage at least the last position is correct. In the calculation of the t-test with these rows, t = 2.11 and P = 0.001. This shows that the first position is more reliable than the last when the significance level is 5%. Therefore, we judge the position that was input first as the position of the fingertip. As the result, the input rate of taps becomes 99.4%.

Table 4. Percentage of the taps recognized correctly.

Subject	A	B	C	D	E	F	G	H	I	J	av.
Complete tap (%)	97.8	97.8	97.8	98.9	95.6	93.9	97.8	95.6	98.9	96.1	97.0
First is correct (%)	**100**	**100**	**98.9**	**100**	**98.3**	**97.8**	**100**	**100**	**100**	**98.9**	**99.4**
Last is correct (%)	97.8	97.8	98.9	98.9	97.2	96.1	97.8	95.6	98.9	97.2	97.6

* The "Complete tap" means that only the correct positions were entered. The "First is correct" and the "Last is correct" mean that even if the extra position is added the first and the last position is the correct position, respectively.

Table 5 shows the number of erroneous inputs for each position in 1200 taps in total. No errors occur except for the four positions shown in the table. The row titled "Add after" indicates the number of errors that occurred because an unnecessary position was added after the correct position. The row "Add before" indicates the error caused by having been added before. Since the former is more than the latter, judging with the first position increases the recognition rate of the tap.

Table 5. Incorrect input patterns of taps and the number of them.

Position	8	6	4	1	total
Add after: [sequence] times	[8,**1**] 23	[6,**5**] 18	[4,**5**] 2		**43**
Add before: [sequence] times			[**5**,4] 7	[2,1] 4	11
Total of errors	23	18	9	4	54

* "Add after" is a case where an unnecessary position was added after the target position. "Add before" is the case before the target. Numbers in brackets indicate the entered sequence of positions.

In both the error patterns at positions 8 and 6, the position to the right is entered after the target position. It necessary to stretch the thumb for touching to position 6 or 8, so the thumb is close to parallel to the board. For this reason, the accidentally touching to the next position increases during subjects return their thumbs. Position 4 is touched by bending the thumb. Position 5 is inserted before the position 4 when the thumb contacts the board before it is completely bent.

6 Character Input Using the Touch Board

The created touch board is connected to the smartwatch with Bluetooth. On/off of the eight sensors of the touch board are combined into one byte data with Arduino every millisecond. Here, ‘on’ means that the sensor is touched. Its value is 1. That data is send to the smartwatch with Bluetooth. The program running on the smartwatch judges the type of stroke and the position of the tap.

If the judgment is made after checking the fingertip has left the board, an input delay that the user can definitely feel is generated. This markedly deteriorates the usability of the input device. Therefore, the stroke judgement algorithm has been modified. The computer experiment in Sect. 5 showed that the first two positions of each stroke are reliable. Accordingly, we judge the stroke from only the first two positions. Since the stroke is decided during the fingertip moving, the delay is solved. In addition, the threshold of consecutive zeros to be deleted has been reduced to 5 in order to detect early that the thumb left the board. This is necessary to shorten tap judgment.

Table 6 shows the definition of the strokes and taps in Dan selection after the modification. Here, the first digit pair in each cell is a standard decision rule used primarily. Pairs of italic digits are prepared to repair the error where the middle position is not entered. Pairs of bold digits are to correct the tapping when the fingertip moves after touching the board.

We measured input speed of one subject who is skilled in the operation of both touch board and touch screen of SliT method. In this experiment, the subject inputs 5 hiragana words in each task. The task is repeated 5 times with 3-min break between. The length of the words is 4, 5, or 6 characters. For each word length, 50 words were selected in the most frequently used order [20]. After sorting 150 words in total at

Table 6. Sequences of positions determining the strokes and the taps after the modification.

Sequence	Input	Sequence	Input	Sequence	Input
6,5 *6,4*	あ	7,8 *7,1*	か	7,6 *7,5*	さ
8,1 *8,2*	た	8,7 *8,6*	な	1,2 *1,3*	は
1,8 *1,7*	ま	2,3 *2,4*	や	2,1 *2,8*	ら
3,4 *3,5*	わ	3,2 *3,1*	Symbol	4,5 *4,6*	Mode
6,0 **6,7**	BS	5,0 **5,4 5,6**	Space	4,0 **4,3**	Enter

random, 10 sets of 5 words from the top of the word list were cut out. Thus the same word never appears twice in the tasks. On the first day, the input speed was measured with the touch screen, on the second day with a touch board.

Table 7 shows the result. The standard deviation of each experiment is small. That means that the person is mastering how to operate. The input speed on the touch board is about 80% of that on the screen. This is because there were relatively many input errors. In particular, many errors occurred in the tap operation for replacing to voiced or semi-voiced sounds. In this operation, the right low position of the board is tapped. Although when input with the touch board, both of the fingertip and the screen must be watched, that position is far from the screen. In addition, since the board is at an angle to the screen, the touch position may be hidden by the finger used for operation. It is considered that the error occurred due to these reasons.

Table 7. Input speed of a person skilled in SliT.

Task	1	2	3	4	5	Avg.	SD
touch board [CPM]	54.4	50.8	54.0	54.6	52.4	53.2	1.61
touch screen [CPM]	67.7	68.0	65.1	70.3	67.3	67.7	1.85

Avg. = average input speed of the 5 tasks, SD = standard deviation

In order to reduce that tap mistake, we plan to make the sensors protrude a little from the surface of the touch board. This makes it possible to find the sensor gap by the feel of the fingertip, so the user can notice the tap mistake. In order to make the user aware of the character replacement, we will paint different colors on the background of voiced, semi-voiced, and unvoiced characters. Adding touch buttons to the upper band of the screen is another solution. In this case these buttons are tapped by the forefinger.

7 Conclusion

We developed a touch board dedicated to SliT which is the character input method for smartwatch. On the board, eight sensors are equally spaced around the center circle. When two adjacent sensors are touched at the same time, the position number assigned between them are output.

In SliT, specify one character by one stroke and one tap. By judging the stroke from the first two positions through which the finger passes, 98.7% of the 1800 strokes were correctly recognized. Even if an additional position was entered, 99.4% of 1200 taps were correctly recognized by judging the first position as the tap position. Therefore, it is considered that the touch board achieves sufficiently high accuracy as an input device of SliT.

The character input speed with the touch board was about 80% of that with the touch screen. This decrease in input speed is mainly caused by that voiced or semi-voiced characters were entered as unvoiced characters due to tap failure. However, this experiment has done only one person. We will do the same experiment for many persons and investigate the cause of the difference of input speed. After that, we will improve the touch board in order to improve input speed.

Acknowledgement. This work was supported by JSPS KAKENHI Grant Number JP16K00286.

References

1. Siek, K.A., Rogers, Y., Connelly, K.H.: Fat finger worries: how older and younger users physically interact with PDAs. In: Costabile, M.F., Paternò, F. (eds.) INTERACT 2005. LNCS, vol. 3585, pp. 267–280. Springer, Heidelberg (2005). https://doi.org/10.1007/11555261_24
2. Akita, K., Tanaka, T., Sagawa, Y.: SliT: character input system using slide-in and tap for smartwatches. In: Kurosu, M. (ed.) HCI 2018. LNCS, vol. 10903, pp. 3–16. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-91250-9_1
3. Fujitsu Develops Glove-Style Wearable Device. http://www.fujitsu.com/global/about/resources/news/press-releases/2014/0218-01.html. Accessed 20 Dec 2018
4. Ring-Type Wearable Device Enables Users to Write in the Air with Finger. https://journal.jp.fujitsu.com/en/2015/02/10/01/. Accessed 20 Dec 2018
5. Ring Zero. https://logbar.jp/en/index.html. Accessed 25 Dec 2018
6. Zhang, Y., Zhou, J., Laput, G., Harrison, C.: SkinTrack: using the body as an electrical waveguide for continuous finger tracking on the skin. http://www.gierad.com/assets/skintrack/skintrack.pdf. Accessed 25 Dec 2018
7. NEC develops ARmKeypad air, a contact-free virtual keyboard for a user's arm. https://www.nec.com/en/press/201607/global_20160713_01.html. Accessed 25 Dec 2018
8. Wong, P., Zhu, K., Fu, H.: FingerT9: leveraging thumb-to-finger interaction for same-side-hand text entry on smartwatches. In: Proceedings of CHI 2018, Paper no. 178 (2017)
9. Whitmier, E., et al.: DigiTouch: reconfigurable thumb-to-finger input and text entry on head-mounted displays. In: Proceedings ACM IMWUT 2017, vol. 1, no. 3, Article 133 (2017)
10. Finger Touching. http://www.yankodesign.com/2007/10/02/wearable-mobile-device-for-enhanced-chatting/. Accessed 25 Dec 2018
11. Tap. http://www.tapwithus.com. Accessed 25 Dec 2018
12. Haier Asu Smartwatch. https://www.digitaltrends.com/smartwatch-reviews/haier-asu-review/. Accessed 25 Dec 2018
13. Sun, K., Wang, Y., Yu, C., Yan, Y., Wen, H., Shi, Y.: Float: one-handed and touch-free target selection on smartwatches. In: Proceedings of CHI 2017, pp. 692–704 (2017)
14. Gong, J., Yang, X., Irani, P.: WristWhirl: one-handed continuous smartwatch input using wrist gesture. In: Proceedings of UIST 2016, pp. 861–872 (2016)

15. Gong, J., et al.: WrisText: one-handed text entry on smartwatch using wrist gestures. In: Proceedings of CHI 2018, Paper no. 181 (2018)
16. Li, F., Guy, R., Yatani, K., Truong, K.: The 1line keyboard: a QWERTY layout in a single line. In: Proceedings of UIST 2011, pp. 461–470 (2011)
17. 5-TILES. https://en.wikipedia.org/wiki/5-Tiles, https://9to5google.com/2015/07/07/wear-5-tiles-is-a-text-messaging-app-with-a-keyboard-for-your-wear-watch/, https://techcrunch.com/2014/05/24/5-tiles-keyboard/. Accessed 25 Dec 2018
18. TouchOne Keyboard. http://www.touchone.net. Accessed 25 Dec 2018
19. Oney, S., Harrison, C., Ogan, A., Wiese, J.: ZoomBoard: a diminutive QWERTY soft keyboard using iterative zooming for ultra-small devices. In: ACM CHI 2013, pp. 2799–2802 (2013)
20. The word list of the Balanced Corpus of Contemporary Written Japanese of the National Institute for Japanese Language and Linguistics. http://pj.ninjal.ac.jp/corpus_center/bccwj/en/freq-list.html. Accessed 23 Jan 2019