JAISCR, 2015, Vol. 5, No. 3, pp.221 – 226
10.1515/jaiscr-2015-0030

IMPROVEMENT OF PERFORMANCE OF JAPANESE P300 SPELLER BY USING SECOND DISPLAY

Yuya Yamamoto, Tomohiro Yoshikawa, Takeshi Furuhashi

Department of Computational Science and Engineering,
Graduate School of Engineering Nagoya University,
Furo-cho, Chikusa-ku, Nagoya 4648603, Japan

Abstract

Brain computer interface (BCI) is a system allows a user to control external devices or to communicate with other people using only his or her thoughts. The P300 speller is one such BCI in which users input letters. For inputting letters via the P300 speller, higher accuracy and shorter input times are needed, especially given densely populated display screens. We propose a new interface with a second display in the P300 speller that the user can switch to and from by selecting the “next” or “back” commands, therby reducing the density of displayed letters and improving the performance of the P300 speller. We show the comparison results in terms of accuracy and input times between the conventional interface and proposed interface.

1 Introduction

Recently, numerous studies have focused on the brain computer interface (BCI), which allows users to control equipment or communicate with other people using their thoughts; more specifically, this interface uses information in signals transmitted from the human brain [1][2]. BCI is expected to aid people with physical disabilities, such as amyotrophic lateral sclerosis (ALS), who cannot move their muscles without help [3]. The P300 speller, first introduced by Farwell et al. [4], is a word input system that utilizes event-related potential as the target feature for classification. There are various types of P300 speller interfaces, and we target the letter matrix interface, shown in Fig.1 and Fig.2. The interface gives the user visual stimuli by flashing each row and column randomly one by one at certain times. When the flashing occurs, the user concentrates on the desired letter by counting the number of times it was flashed; thereby, P300 is obtained in the flash of the row or column that contains the desired letter. Next, the system discriminates the letter that is most likely the target. In this paper, we refer to a sequence as a series of flashes in which every row and column flashes once.

Figure 1. English interface

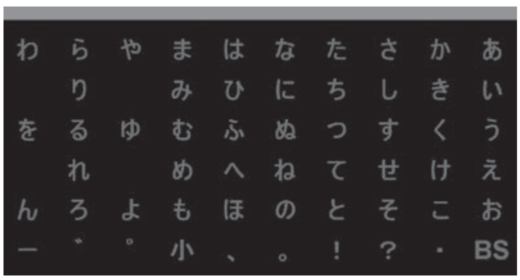


Figure 2. Japanes interface

The P300 interface, which was developed by Farwell and others, is an English interface containing alphabets and numbers, as shown in Fig.1. It has also been used for inputting Japanese; however, in clinical experiments with ALS subjects, more time was taken to input letters as the subjects were not able to input Japanese directly and were not accustomed to inputting Hiragana using Roman letters; therefore, the accuracy decreased. In this study, we used the Japanese interface, which has 56 letters containing Hiragana and symbols on a 6×10 matrix (Fig.2). The Hiragana interface needs more choices since the number of letters in Hiragana are more than that in the English interface (6×6). In the Japanese P300 speller, the flash time per sequence is longer and the space between letters is narrower because of the increased matrix size. Thus, this change in interface may cause increased input times and a decrease in accuracy.

We propose a new Japanese interface to shorten input times and improve accuracy by eliminating the problematic factors described above. To input a letter by the proposed interface, the user changes the displays to view either the first half or the second half of Hiragana by inputting the CD or CD command. We conducted experiments using both conventional and proposed interfaces for the P300 speller, and the results show the improvement of performance in inputting letters using the proposed interface.

2 Proposed interface

2.1 Interface with second display

As shown in Fig.2, the conventional Japanese P300 speller consists of one screen in which all characters (i.e., Hiragana) and symbols are allocated in the form of a 6×10 matrix. In this interface, however, problems arise because of the increased matrix size. One problem is the increase in the number of flashes per sequence; another is that letters in the neighboring row or column of the target letter tend to be incorrectly identified, because the space between letters is narrower. Therefore, we propose the new interface shown in Fig.3, which has more space between letters and divides the Hiragana matrix into two smaller matrices, i.e., two 6×5 matrices, one per display. The right display of Fig.3, which has a line of あ〜な in the matrix, is shown to the user before inputting every letter, and then he or she can switch the display by inputting the ＜次＞ or ＜戻＞ command. The right display of Fig.3 is called the first display and the left one is called the second display, and each input of Hiragana or symbol is considered as one command in each interface. The conventional interface (Fig.2) is called a 1matrix interface, and the proposed interface (Fig.3) is called a 2 matrices interface.

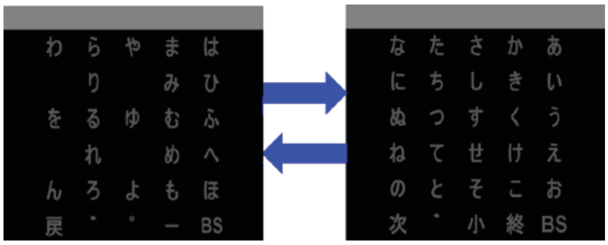


Figure 3. Proposed Interface

Table 1 shows the comparison of the 1matrix and 2matrices interfaces. Each display in the 2matrices interfaces has a smaller matrix relative to the 1matrix interface; therefore, the spaces between letters and symbols can be wider. Thus, we expect the rate of incorrect recognition of letters near the target to decrease. Unfortunately, the 2matrices interface may require more time for inputting letters on the second display, since the user has to input two commands (＜次＞ (＜ next ＞) and the letter on the second display). The accuracy may also decrease (e.g., if the accuracy is 90%, it will become 81% (0.9×0.9) for letters located in the second display).

2.2 Calculation of expected input time

According to Google's Japanese corpus data (1-gram), the occurrence rate of letters in the first display (line あ〜な) and that in the second display (line は〜わ) are 7:3, meaning that in Japanese sentences, 70% of letters appear in the first display and 30% in the second display. The input time caluculation per letter based on this occurrence rate is shown below. Inccorect time is calculated by considering the inccorect inputs, which are collected by $< BS >$ (BackSpace).

In the case of the 1matrix interface, the average input time per letter is given by Eq.(1). Similarly, for the 2 matrices, the average input time per letter is given by Eq.(2). When a non-target letter is inputted, the user selects $< BS >$ as the next target

Table 1. Comparison Of Interfaces

	1matrix	2matrices
Space between letters	Narrow (6 rows, 10 columns)	Wide (6 rows, 5 columns)
The number of stimuli (Per sequence)	16 (6rows, 10columns)	11 (6 rows, 5 columns)
Display switching	N/A	Necessary

to delete the preceding letter and input the full sentence correctly. The input time for the $< BS >$, <次> and <戻>commands is also considered in Eq.(2).

$$T_1 = \frac{a_1 \cdot s \cdot n}{2p-1} \tag{1}$$

$$\begin{aligned} T_2 &= \frac{a_2 \cdot s \cdot 0.7n}{2p-1} + 2\frac{a_2 \cdot s \cdot 0.3n}{2p-1} \\ &= 1.3\frac{a_2 \cdot s \cdot n}{2p-1} \end{aligned} \tag{2}$$

Here, T_1 and T_2 show the input time for a full sentence in each interface. In Eq.(1) and Eq.(2), p is the discriminant accuracy, n is the number of letters to be inputted, s is the interval between stimuli, a_1 is the average number of flashes for one command in the 1matrix interface and a_2 is that in the 2 matrices interface. In Eq.(2), the first term on the right side represents the input time of the letters in the first display and the second term represents the input time of the letters in second display. The coefficients 0.7 and 0.3 in the first and second terms represent the occurrence rate of letters in the first and second displays, respectively. Coefficient 2 in the second term represents two required inputs (<次> and the target letter on the second display). The input time of the 2matrices interface, derived from Eq.(1) and Eq.(2), is shown below.

$$T_2 = 1.3 \cdot \frac{a_2}{a_1} \cdot T_1 \tag{3}$$

Assuming that the same sequences are required to input a letter in each interface, we define the equation below.

$$a_1 = \frac{16}{11} \cdot a_2 \tag{4}$$

Thus, the expected value for the input time in the 2matrices interface is given below.

$$T_2 = 0.89T_1 \tag{5}$$

According to the above equation, the input time in the 2matrices interface is expected to be 0.89 times of that in 1matrix interface. Hence, the use of the 2matrices interface is expected to enable the user to input letters more quickly.

3 Experiment

The accuracy and input times for the 1matrix and 2matrices interfaces were compared via offline experiments.

3.1 Data description

We used a recorded dataset that contained EEG data measured from four subjects (Sub1-Sub4), each using the P300 interface. EEG data were recorded from five electrodes -Fz, Cz, Pz, O_1, O_2- and A_2 as a reference, at a sampling rate of 100 Hz using Polymate Ap216 (DIGITEX LAB. CO., LTD, Tokyo, Japan). Each stimulus was intensified for 200ms with an inter-stimulus interval of 200ms. We used the 1matrix (Fig.2) and 2matrices interfaces (Fig.3) and 40 letters worth of EEG data were recorded for each interface. When the target letter was intensified, the recorded data were labeled as “P300-data” and the others were labeled as “nonP300-data.” Considering subjects’ fatigue, recording in either the 1matrix or 2matrices interfaces was alternately assigned in an order that brought a balance among subjects. In each interface, the input of one letter consisted of ten sequences.

3.2 Experimental settings

The accuracy and input times per letter were compared. Twelve letters were utilized as learning data in each interface from the dataset. The pretraining data of the 1matrix interface consisted of

240 stimuli labeled P300-data and 1680 stimuli labeled nonP300-data, and that of the 2matrices interface consisted of 240 stimuli labeled P300-data and 1080 stimuli labeled nonP300-data. Therefore, the pre-training time of the 1matrix interface was longer than that of the 2matrices interface. The EEG data were down-sampled to 20Hz, and then 17 data points corresponding to 0-0.8s after each stimulus were extracted. The extracted data were classified using stepwise linear discriminant analysis[5] for the discrimination of P300/non-P300 data. In the test session in which users input letters, P300-data were selected in the target flash and nonP300-data were selected in any other flashes from the dataset, except for the data used in pre-training. Ten letters were inputted in a trial, which was conducted 100 times. The pre-training and test data were randomly changed in every trial.

In our experiments, to discriminate the target letter, RB-ARQ [6] was employed, a method that randomly presents stimuli until the maximum posterior probability is above a given threshold (0.95 and 0.9 in our experiments). In RB-ARQ, the number of stimuli is dynamically set, and the discrimination time becomes shorter relative to conventional discrimination, which fixes the number of stimuli. In RB-ARQ, the maximum flash times were set at ten sequences per letter.

4 Results and Discussions

Table 2 shows results of the accuracy and input times of Sub1-Sub4 in each interface when the threshold of RB-ARQ was 0.95. The accuracy was the rate of inputted correct letters in 1000 letters, and the input time was the average inputted time per letter. Input times were calculated by setting $n = 1$ in Eq. (1) and Eq. (2). Table 2 shows that there was no substantial difference in the accuracy between the 1matrix and 2matrices interfces. By applying RB-ARQ, when there is a difference in discriminant rates, the effect is reflected not on the accuracy but on the shortness of the input time. In table 3 , T_1 and T_2 are the input time in the 1matrix and 2matrices interfaces, respectively, and $0.89T_1$ is the expected input time in the 2matrices interfaces, as calculated by Eq. (5). Table 3 shows that T_2 of Sub1, Sub2, and Sub3 are a little longer than the expected value $0.89T_1$. We conclude that this is so because the 1matrix interface has more information per sequence, when the posterior probability of whether each letter will be a target, is calculated in every stimulus of row or column in RB-ARQ. We believe that the assumption that the same sequences are needed to input a letter in each interface described in 2.2 was incorrect. Therefore there was a difference between the experiment and expected input time. Conversely, T_2 (2matrices interface experimental value) was lower than T_1 (1matrix interface experimental value) in almost all subjects.

Table 3. Comparison of Experimental and Expected Input Time (Threshold: 0.95)

	T_1 (1matrix)[s]	$0.89T_1$[s]	T_2 (2matrices)[s]
Sub1	17.5	15.6	16.0
Sub2	15.9	14.2	15.7
Sub3	50.6	45.0	50.7
Sub4	19.5	17.3	14.8

Table 4 shows results of the accuracy and input times of Sub1-Sub4 in each interface when the threshold of RB-ARQ was 0.9. Table 4 shows that there were no substantial differences in accuracy between the 1matrix and 2matrices interfaces; similar to the case when the threshold was 0.95. T_2 was also lower than T_1 for almost all subjects. Only Sub3 required additional input time to reach a similar level of accuracy in each threshold, since the difference between P300-data and nonP300-data in Sub3 might be smaller than other subjects.

Since there is essentially a tradeoff between accuracy and input times, we utilized "Utility"[7] defined in Eq.(6), as the performance index for evaluating accuracy and input times at the same time. Utility corresponds to the information transfer rate when spelling is performed perfectly using "Backspace" to delete the last incorrect letter. In Utility calculation, $< BS >$ was defined as one of the letter candidates, which ＜次＞ and ＜戻＞ in the 2matrices interface were not considered as letter candidates because Utility is the mutual information utilizing the information as to which letter will be inputted by the P300 speller. Utility is defined as

$$\text{Utility} = \frac{(2p-1)\log_2(N-1)}{d} \qquad (6)$$

Table 2. Accuracy and Input Time (Threshold 0.95)

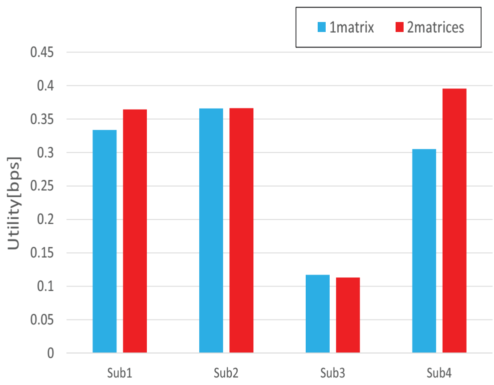
	1matrix		2matrices	
	Accuracy	Input Time[s]	Accuracy	Input Time[s]
Sub1	0.94	17.5	0.95	16.0
Sub2	0.94	15.9	0.93	15.7
Sub3	0.90	50.6	0.88	50.7
Sub4	0.93	19.5	0.93	14.8

Table 4. Accuracy and Input Time (Threshold 0.90)

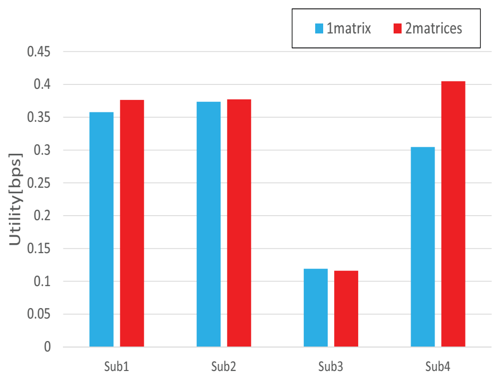
	1matrix		2matrices		$0.89T_1$[s]
	Accuracy	Input Time[s]	Accuracy	Input Time[s]	
Sub1	0.92	16.3	0.90	15.5	14.5
Sub2	0.88	15.6	0.88	15.5	13.9
Sub3	0.85	49.0	0.83	51.0	43.6
Sub4	0.87	19.5	0.90	14.3	17.3

where N is the number of classes (N =56 for the 1matrix interface, and N =54 for the 2matrices interface), p is the accuracy, and d is the input time per letter. Note that when p <0.5, U=0.

Fig.4 shows the Utility of Sub1-Sub4 in each interface. The figure shows that the total performance of the 2matrices interface was better than that of the 1matrix interface in almost all subjects. There was a significant difference between the Utility of the 2matrices interface and the 1matrix interface in each threshold by a paired t-test at the significant level of 0.025 (considering the multi comparison (Bonferroni), threshold 0.95: $p = 1.67 \times 10^{-8} < 0.025(= 0.05/2)$, threshold 0.9: $p = 1.88 \times 10^{-5} < 0.025(= 0.05/2)$). These results show that using the 2matrices interface reduced pre-training time and input time while maintaining accuracy compared with the 1matrix interface.



(a) Threshold: 0.95



(b) Threshold: 0.90

Figure 4. Utility in each subject

5 Conclusion

We proposed a new Japanese interface for the P300 speller with two displays to remove problems of the conventional Japanese interface. In the proposed interface, a user changes the display between a first half and second half of Hiragana by inputting the or commands. Our experimental results showed that Utility, an index of total performance considering both accuracy and input times for inputting letters, improved using the 2matrices interface. In a future study, we intend to investigate how changing the size of the letters and the space between them in the 2matrices interface will impact accuracy, as well as the mental burden that users face in changing the displays of input letters.

Acknowledgement

This work was supported by the Strategic Information and Communications R&D Promotion Program (SCOPE) no.141206007 of The Ministry of Internal Affairs and Communications in Japan.

References

[1] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain-computer interfaces for communication and control," *Clinical Neurophysiology*, vol. 113, no. 6, pp. 767–791, 2002.

[2] K. Tatsuhiro and Y. Kiyoyuki, "Current studies on brain-computer-interface(bci)," in *The Bulletin of School of High-Technology for Human Welfare, Tokai Univ*. Tokai Univercity, 2011, pp. 7–12.

[3] D. Kaub-Wittemer, N. Steinbuchel, M. Wasner, G. Laier-Groeneveld, and G. Borasio, "Quality of life and psychosocial issues in ventilated patients with amyotrophic lateral sclerosis and their caregivers," *Journal of pain and symptom management*, vol. 26, no. 4, pp. 890–896, 2003.

[4] L. Farwell and E. Donchin, "Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials," *Electroencephalography and clinical Neurophysiology*, vol. 70, no. 6, pp. 510–523, 1988.

[5] D. J. Krusienski, E. W. Sellers, S. Bayoudh, F. Cabestaing, D. J. McFarland, T. M. Vaughan, and J. R. Wolpaw, "A comparison of classification techniques for the p300 speller," *Journal of Neural Engineering*, vol. 3, no. 4, pp. 299–305, 2006.

[6] H. Takahashi, T. Yoshikawa, and T. Furuhashi, "A study on application of reliability based automatic repeat request to brain computer interfaces," in *Advances in Neuro-Information Processing*. Springer, 2009, pp. 1013–1020.

[7] B. Dal Seno, M. Matteucci, and L. Mainardi, "The utility metric: a novel method to assess the overall performance of discrete brain computer interfaces," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 1, pp. 20–28, 2010.

Yuya Yamamoto received the B.E. degree from the Nagoya University, Japan, in 2014. He is currently a Graduate student at the Nagoya University. His research interest includes Brain computer interface. He is a member of SOFT and IEICE.

Tomohiro Yoshikawa received the Ph.D degree from Dept. of Information Electronics, Nagoya University, Japan, in 1997. From 1997 to 1998, he was a visiting researcher, University of California at Berkeley. From 1998 to 2005, he was an assistant professor, Dept. of Electrical and Electronic Engineering, Mie University. He is currently an associate professor of Dept. of Computational Science and Engineering, Nagoya University. His research interest includes soft computing and its applications. He is a member of SOFT, IEICE, IPSJ, JSAI, IEEE and The Japanese Society for Evolutionary Computation.

Takeshi Furuhashi was an assistant professor of School of Engineering, Nagoya University, Japan, from 1988 to 1990. From 1990 to 2001, he was an associate professor of School of Engineering, Nagoya University. From 2001 to 2004, he was a professor of School of Engineering, Mie University. He is currently a professor of Dept. of Computational Science and Engineering, Nagoya University. His research interests include soft computing and Kansei engineering. He is a member of SOFT, SICE, IEICE, JSAI, IEEJ and IEEE.