35th Annual International Conference of the IEEE EMBS
Osaka, Japan, 3 - 7 July, 2013

Asynchronous Brain–Computer Interfacing Based on Intended Movement Direction

Keita Shimpo and Toshihisa Tanaka

***Abstract*— A brain–computer interface (BCI) is a technique for controlling devices with the measured human brain activities. Especially, an asynchronous BCI is one of the most important topics since practical input interfaces are incomplete without self–paced inputs. In order to construct an asynchronous BCI, it is essential to recognize the standby state, where a user enters no commands. In this paper, we propose a novel method for detecting the standby state and develop an asynchronous BCI based on event–related potentials with the intended movement direction. We conducted online experiments with developed asynchronous BCI. As a result, all three subjects showed considerable recognition accuracies.**

I. Introduction

Brain–computer interfacing (BCI) is a challenging application of signal processing, machine learning, and neuroscience. BCIs capture brain activities associated to mental tasks and external stimuli [1], and realize non-muscular communication and control channel for conveying messages and commands to the external world. Basically, noninvasive measurement devices such as electroencephalogram (EEG), magnetoencephalogram (MEG), and functional magnetic response imaging (fMRI) are widely used to observe the brain activities. Among them, because of its simplicity and low cost, EEG is a practical measurement device for use in engineering applications. However, EEG contains various components such as noise and background potentials. Therefore, we need to extract useful features from EEG for BCI.

Well–known useful features appearing in EEG are event-related potentials (ERP), which is triggered as a result of thinking or cognition, visual evoked potentials (VEP), which is EEG responses for external visual stimuli, and so forth. These features have been extensively exploited in various types of BCI [2], [3].

To develop a practical interface, one of the important and challenging topics in BCI is a so-called self-paced BCI or *asynchronous* BCI [4], which enable a user to enter commands at any time. In other words, the most considerable feature of an asynchronous BCI is that it has a *standby* (state), that is the state while the user enters no command . If the state is recognized as not standby, the entered command is recognized. On the other hand, a conventional BCI, which is synchronous, always detect a command that the user enters.

A straightforward way to develop an asynchronous BCI is the so-called polling, a.k.a busy wait. The main step of this scheme is to check at regular short intervals whether or not the user is entering a command. Several studies of asynchronous BCI based on this idea have been reported [5], [6]. These previous methods deal with the standby state as single independent class. However, this approach does not take into account the following facts. 1) Samples of the standby state are scattered in the feature space. 2) In a practical asynchronous BCI, a user may mostly be in the standby state. This paper develops a classification method that considers the above observation. For an K-class BCI, we propose a supervised learning of a set of K regression functions named *relevance* to detect the standby state and classify the command. The relevance quantifies the closeness of the input sample to each class. If all the relevance to the sample is small, the user is considered in standby. Otherwise, the sample is considered as belonging to the class where the corresponding relevance is the largest.

We applied the proposed detection/classification method in an asynchronous BCI based on the intended movement direction [7]. The relevance was learnt with data obtained by an offline experiment based on the delayed saccade-and-reach task, evoking an ERP in the posterior parietal cortex (PPC), when a human intends a left or right direction movement with eye movements [8]. By extracting the ERP, we can estimate the intended movement direction [9]. We conducted online experiments of the asynchronous BCI with the proposed method. Experimental results show that all three subjects showed considerable recognition accuracies.

II. Methods

A. Proposed Method — Standby Detection

A relevance map defined in this paper is a map that moves samples of not belonging to any class around the origin. By extending the one-versus-the-rest classifier [10] to solve a multi-class classification problem as a regression problem using a two-class classifier, we quantified with a real value the similarity of an input sample to a particular class. We call this similarity the relevance defined as follows.

Assume that we have a training dataset which include N feature vectors, $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N \in \mathbb{R}^M$, and the K command $C_1, C_2, \ldots, C_K$. We consider the $(K+1)$–class regression-based classification where a test sample is classified to either K classes or the standby class.

*This work was supported in part by JSPS KAKENHI Grant Number 23650059.

K. Shimpo and T. Tanaka are with the Department of Electrical and Electronic Engineering, Faculty of Engineering, Tokyo University of Agriculture and Technology, Tokyo, Japan. (email: kei@sip.tuat.ac.jp, tanakat@cc.tuat.ac.jp)

T. Tanaka is also affiliated with the Laboratory for Advanced Brain Signal Processing, RIKEN Brain Science Institute, Saitama, Japan.

The classifier is designed as follows. Let $f_1(\mathbf{x}),\ldots,f_K(\mathbf{x})$ be the K regression functions to be trained from dataset $\{(\mathbf{x}_n, y_k(\mathbf{x}_n))\}_{n=1}^{N}$, where y_k is designed as

$$y_k(\mathbf{x}) = \begin{cases} w_k(\mathbf{x}) & \text{if } \mathbf{x} \text{ belongs to } C_k \\ -1 & \text{otherwise} \end{cases}, \tag{1}$$

where $w_k(\mathbf{x})$ is a weighting function which represents the similarity between each feature vector and C_k.

For the regression function of class k, we call $L(f_k(\mathbf{x}))$ the *relevance* of $\mathbf{x}$ to class k, where L is a logistic function given as

$$L(t) = \frac{1}{1+\exp(-t)}. \tag{2}$$

We also define the *relevance map* from $\mathbb{R}^M$ to $\mathbb{R}^K$ as

$$\mathbf{r}(\mathbf{x}) = [L(f_1(\mathbf{x})),\ldots,L(f_K(\mathbf{x}))]^T. \tag{3}$$

For each test sample, we calculate the relevance to k to detect the state of the user, as described as follows.

1) If $L(f_k(\mathbf{x})) < T$ for all k, then $\mathbf{x}$ is classified into the standby class, where T is threshold.
2) Otherwise, $\mathbf{x}$ is classified into $C_{k^\star}$, where

$$k^\star = \underset{k=1,\ldots,K}{\arg\max} L(f_k(\mathbf{x})). \tag{4}$$

B. Experimental Settings

We carried out experiments consisting of two stages training phase and test phase. The training phase is conducted to learn parameters of relevance map. The test phase evaluates the performance of the asynchronous BCI with the learned relevance map.

Three males (Subject 1-3) in their twenties took part into our experiment. All subjects had normal vision and were given informed consent, and this study was approved by the research ethics committee of Tokyo University of Agriculture and Technology.

An LCD screen with a size of 23 inch's was used for displaying a visual cue and feedback. This screen has 120 Hz refresh rate and 1920×1080 resolution. During a whole experiment, subjects sat on a comfortable chair in front of the screen about 40 cm away and focused on the cue window. Additionally, the subjects' heads was held steady using a chin rest.

C. Data Acquisition

We used Ag/AgCl active electrodes which are product of Guger Technologies (g.tec) named g.LADYbird, g.LADYbirdGND (for GND), g.GAMMAearclip (for reference, earclip type) for recording EEG data. These were driven by the power supply unit named g.GAMMAbox (g.tec). The 14 electrodes following the 10–5 system [11] were located at positions OI1h, OI2h, O1, O2, POO9h, POO10h, I1, I2, PO7, PO8, PPO9h, PPO10h, PO9, PO10 and were referenced to the A1 and grounded to the AFz. The signals were amplified 20,000 times by MEG-6116 (Nihon Kohden), that has high-cut and low-cut analog filter for each channel. We set the high-cut filter and the low-cut filter to 100 Hz and 0.5 Hz, respectively. The EEG signal is sampled by A/D converter (AIO-163202F-PE, Contec) a rate of 256 Hz. The signals are recorded with Data Acquisition Toolbox of the MATLAB (MathWorks).

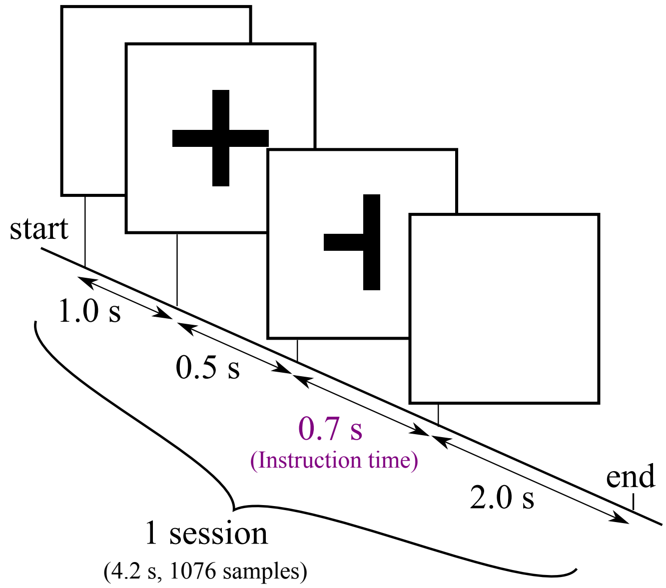

Fig. 1. The flow of visual cues in the training phase

D. Training Phase

The purpose of this phase is to learn the parameters of the relevance map by the delayed saccade-and-reach task [7]. In our experiment, we will construct a 2–command asynchronous BCI then K of relevance map is 2.

1) Task: A subject pushes Enter key to start a recording session of this phase. Figure 1 shows the flow of visual cues of each session. At the beginning of the session, the display shows a white window. After 1.0 s, it displays symbol $+$ for 0.5 s on the window, and for the next 0.7 s, the symbol either $\dashv$ or $\vdash$ appears. These symbols are the cue for the leftward movement and rightward movement, respectively. In this paper, we call this period of 0.7 s the *instruction time*. The subject, when the cue appeared, pays attention to the target and performed the eye movement consciously. After that, the subject moved his right hand accordingly as soon as the visual cue has disappeared. The leftward movement is to press the left Ctrl key on the keyboard and the rightward movement is to push the one on the opposite side. When the cue has disappeared, after 2.0 s, the session is complete with beep sound. It should be noted that a recorded EEG signal in as session has 1076 samples (see Fig. 1), since the session lasts for 4.2 s with the sampling frequency of 256 Hz. The training phase has been completed after subjects performed 100 sessions, consisting of 50 sessions each for left and right, in random order.

2) Feature Extraction and Parameter Learning: First, we cut out an epoch with a length of 180 samples (0.7 s) from the recorded EEG. The start point of each epoch is shifted by 10 samples. The number of epochs obtained in a single

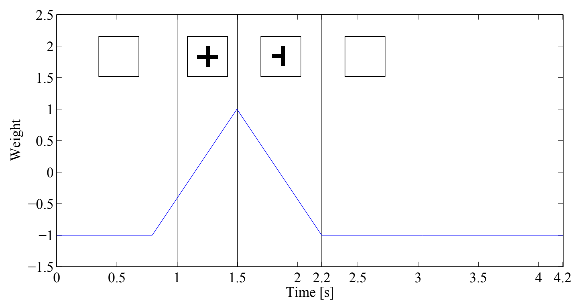

Fig. 2. The weight function, $w_k(\mathbf{x})$, given as in (5). On the upper part in the figure, the corresponding visual cues are depicted.

session is 90 (we did not use the last 6 samples), since a session has 1076 samples. For each epoch at all channels, we apply Butterworth band–pass filter (1–30 Hz). Then, the filtered epoch signal is averaged over 7 channels on each left and right occipital hemisphere. Finally, we concatenate these two vectors to get a 360 dimensional feature vector $\mathbf{x}$. We executed this feature extraction for all the 9000 epochs (90 epochs $\times$ 100 sessions).

For regression, we should define the weighting function $w_k(\mathbf{x})$ as

$$w_k(\mathbf{x}) = \begin{cases} \frac{(s(\mathbf{x})-294)}{90} & \text{if } 204 < s(\mathbf{x}) \leq 384 \\ -\frac{(s(\mathbf{x})-474)}{90} & \text{if } 384 < s(\mathbf{x}) \leq 564 \\ -1 & \text{otherwise} \end{cases} \quad (5)$$

where $s(\mathbf{x})$ is the start point of a epoch in a session. Figure 2 shows the plot of $w_k(\mathbf{x})$.

The underlying idea behind the above weighting function is that take the ratio of ERP on an epoch signal into account. For an epoch that clearly contains no ERP, the weight is set to -1.

We used the support vector regression (SVR) [12] as a relevance map, and adopted the Gaussian kernel. The kernel parameters are precision γ and normalizing parameter C. We determined the parameters by grid search as follows. Assume that $\gamma = 2^x$ and $C = 2^y$. We change by 0.2 the value of x and y subject to $-6 \leq x \leq 2$, $-2 \leq y \leq 2$ and calculate accuracy using 10-fold cross-validation on training dataset. We selected the parameters that gave the highest accuracy. The accuracy was obtained by to solve three-class classification problem (standby, left, and right) using multi-class support vector machine (SVM). In our experiment, we used LIBSVM [13] for implementation of SVR and SVM.

E. Test Phase

In this phase, we implemented an asynchronous BCI based on the intended movement direction using a relevance map obtained in the training phase and evaluated its performance.

1) Asynchronous BCI Design: The BCI system always has a ring buffer stored EEG data of 180 samples and convert from the entire buffer to a feature vector using the same feature extraction methods as the training phase every $1/8$ s. By using the relevance map that obtained from the training phase, this system calculates the relevance of the feature vector every getting it. When a relevance value exceeds the fixed threshold, this system recognizes that the command corresponding to the component is entered .

To determine the threshold corresponding to the command input sensitivity. The subjects actually enter commands to this asynchronous BCI and get feedback. By the references of the feedback, they adjust the sensitivity in the manual.

2) Task: The subjects push Enter key for start to each session of test phase. At the beginning of each session, the display shows a white window. After 0.2 s, it display $+$, $\dashv$, or $\vdash$ symbol during the 0.5 s to the window. These mean the cue of the no action (standby) command, entering the left command, and entering the right command, respectively. The subjects, when the cue disappeared, enter the specified command within 5 seconds as soon as possible. However, to enter the left and right commands, subjects execute only a motor imagery and the eye movement that they gaze at the target, without actually moving their hand. The session is completed by 1) entering the left or right commands, 2) elapsing 5 s. Then, to enter the standby command, subjects do not enter any commands during 5 s. At the end of a session, this system displays the entered command on the screen as visual feedback. The test phase has been completed after the subject performed total 150 sessions (the standby, left, and right, 50 sessions each) in random order.

III. Results and Discussion

A. ERP for Intended Movement Directions Tasks

Figures 3 and 4 show the difference of temporal EEG waveforms obtained from different intended movement direction for Subject 1. The figures illustrate waveforms of all channels averaged over 50 trials of the training phase data at the left or right sessions. In Figs. 4 and 5, the reference lines were added at point of 1.5 s, 1.7 s, 1.82 s, and 2.2 s. The interval from 1.5 s to 2.2 s is the instruction time. At the points of 1.7 s and 1.82 s, a feature of ERP was clearly appeared.

To quantitatively evaluate the difference of ERPs between two different tasks, we conducted the two-class classification of the right and left command using Bayesian linear discriminant analysis (BLDA) applied to the epoch of the instruction time. For classification accuracies, we used the EEG data of the training phase (left and right, 50 sessions each) with 10–fold cross-validation. As a result, we obtained considerably high accuracy of 0.97, 0.98, and 0.92 for Subjects 1, 2, and 3, respectively. It is interesting that this accuracy is much larger than the largest value of 0.80 reported in [9], where spectral power features derived from short-time Fourier transforms are used.

B. Classification Accuracy of the Asynchronous BCI

Tables I, II, and III list the experimental results of confusion matrix of each subject. For the left and right command, the average time till the relevance exceeds the threshold from the beginning of the cue is also shown.

TABLE I
ACCURACY OF SUBJECT 1

Output	Cue			Average Time [s]
	Left	Right	Standby	± S.D.
Left	0.70	0.08	0.08	1.285 ± 1.063
Right	0.30	0.76	0.20	1.601 ± 1.233
Standby	0.00	0.16	0.72	–

TABLE II
ACCURACY OF SUBJECT 2

Output	Cue			Average Time [s]
	Left	Right	Standby	± S.D.
Left	0.92	0.04	0.02	1.227 ± 1.112
Right	0.04	0.88	0.08	1.558 ± 1.342
Standby	0.04	0.08	0.90	–

TABLE III
ACCURACY OF SUBJECT 3

Output	Cue			Average Time [s]
	Left	Right	Standby	± S.D.
Left	0.74	0.24	0.06	0.702 ± 0.745
Right	0.26	0.50	0.16	0.996 ± 1.188
Standby	0.00	0.26	0.78	–

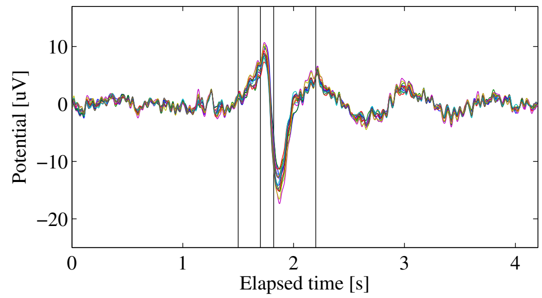

Fig. 3. Average of temporal EEG waveforms of Subject 1 at left sessions.

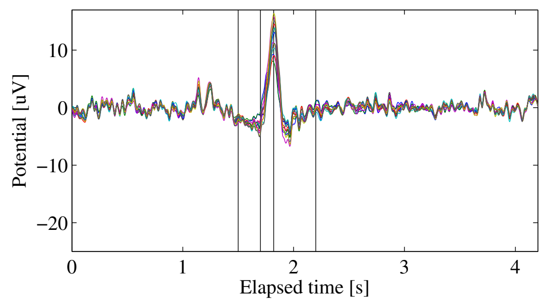

Fig. 4. Average of temporal EEG waveforms of Subject 1 at right sessions.

The average of accuracy Left, Right, and Standby for the Subject 1 to 3 were 0.73, 0.90, and 0.67, respectively. This is a considerable result at 3-class classification problem. Subject 3 showed the lowest accuracy of 0.5 for right command. However, the accuracy is still higher than the error rate of 0.24. This implies that in principle, subjects can enter the desired command if they are allowed to spend unlimited time to enter commands.

These values of accuracy are basically lower than that of the aforementioned two-class classification with the BLDA. Therefore, including the asynchrony together with the standby state in the BCI could deteriorate the classification accuracy. However, as mentioned before, the accuracy greatly depends on the classifier design and feature extraction; therefore, more appropriate design of the classifier can improve the performance.

IV. CONCLUSION

In this paper, we proposed a method named relevance map to construct an asynchronous BCI employing the method of polling. We designed and implemented the asynchronous BCI based on intended movement direction with the proposed classifier, and evaluated the classification performance by the online experiment. As a result, we obtained considerable accuracy (0.67–0.90) for all three subjects.

REFERENCES

[1] J. J. Vidal, "Toward direct brain–computer communication," *Annu. Rev. Biophys. Bioeng.*, vol. 2, pp. 157–180, 1973.

[2] G. Pfurtscheller, C. Neupwe, C. Guger, W. Harkam, H. Ramoser, A. Schlögl, B. Obermaier, and M. Pregenzer, "Current trends in Graz brain–computer interface (BCI) research," *IEEE Trans. Neural Syst. Rehab. Eng.*, vol. 8, pp. 216–219, June 2000.

[3] M. Middendorf, G. McMillan, G. Calhoun, and K. S. Jones, "Brain–computer interfaces based on the steady-state visual-evoked response," *IEEE Trans. Neural Syst. Rehab. Eng.*, vol. 8, no. 2, pp. 211–214, June 2000.

[4] R. Leeb, D. Friedman, G. R. Müller-Putz, R. Scherer, M. Slater, and G. Pfurtscheller, "Self-paced (asynchronous) BCI control of a wheelchair in virtual environments: A case study with a tetraplegic," *Computational Intelligence and Neuroscience*, vol. 2007, 2007.

[5] A. Bashashati, R. K. Ward, and G. E. Birch, "Towards development of a 3–state self–paced brain–computer interface," *Computational Intelligence and Neuroscience*, vol. 12, Aug. 2007.

[6] C. Zhang, Y. Kimura, H. Higashi, and T. Tanaka, "A simple platform of brain-controlled mobile robot and its implementation by SSVEP," in *Proc. IEEE International Joint Conference on Neural Networks*, June 2012, pp. 1543–1549.

[7] P. Hammon, S. Makeig, H. Poizner, E. Todorov, and V. de Sa, "Predicting reaching targets from human EEG," *IEEE Signal Processing Mag.*, vol. 25, no. 1, pp. 69–77, Jan. 2008.

[8] Y. Wang and S. Makeig, "Predicting intended movement direction using EEG from human posterior parietal cortex," *Lecture Notes in Computer Science*, vol. 5638, pp. 437–446, 2009.

[9] J. Li, Y. Wang, L. Zhang, and T.-P. Jung, "Combining ERPs and EEG spectral features for decoding intended movement direction," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2012, pp. 1769–1772.

[10] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY: Springer, 2006.

[11] R. Oostenveld and P. Praamstra, "The five percent electrode system for high-resolution eeg and erp measurements." *Clinical Neurophysiology*, vol. 112, pp. 713–719, Apr. 2001.

[12] H. Drucker, C. Burges, L. Kaufman, A. Smola, and V. Vapnik, "Support vector regression machines," *Advances in Neural Information Processing Systems*, vol. 9, pp. 155–161, 1997.

[13] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.