The Study of Generic Model Set for Reducing Calibration Time in P300-Based Brain–Computer Interface

Jing Jin , *Senior Member, IEEE*, Shurui Li, Ian Daly, Yangyang Miao, Chang Liu, Xingyu Wang, and Andrzej Cichocki , *Fellow, IEEE*

Abstract — **P300-based brain-computer interfaces (BCIs) provide an additional communication channel for individuals with communication disabilities. In general, P300-based BCIs need to be trained, offline, for a considerable period of time, which causes users to become fatigued. This reduces the efficiency and performance of the system. In order to shorten calibration time and improve system performance, we introduce the concept of a generic model set. We used ERP data from 116 participants to train the generic model set. The resulting set consists of ten models, which are trained by weighted linear discriminant analysis (WLDA). Twelve new participants were then invited to test the validity of the generic model set. The results demonstrated that all new participants matched the best generic model. The resulting mean classification accuracy equaled 80% after online training, an accuracy that was broadly equivalent to the typical training model method. Moreover, the calibration time was shortened by 70.7% of the calibration time of the typical model method. In other words, the best matching model method only took 81s to calibrate, while the typical model method took 276s. There were also significant differences in both accuracy and raw bit rate between the best and the worst matching model methods. We conclude that the strategy of combining the generic models with online training is easily accepted and achieves higher levels of user satisfaction (as measured by subjective reports). Thus, we provide a valuable new strategy for improving the performance of P300-based BCI.**

J. Jin, S. Li, Y. Miao, C. Liu, and X. Wang are with the Key Laboratory of Advanced Control and Optimization for Chemical Processes, Ministry of Education, East China University of Science and Technology, Shanghai 200237, China (e-mail: jinjingat@gmail.com; shuruili1008@163.com; miaoyy1991@163.com; lcecust@mail.ecust.edu.cn; xywang@ecust.edu.cn).

I. Daly is with the Brain-Computer Interfacing and Neural Engineering Laboratory, School of Computer Science and Electronic Engineering, University of Essex, Colchester CO4 3SQ, U.K. (e-mail: i.daly@essex.ac.uk).

A. Cichocki is with the Skolkovo Institute of Science and Technology (SKOLTECH), 143026 Moscow, Russia, with the RIKEN Brain Science Institute, Wako 351-0198, Japan, and also with Nicolaus Copernicus University (UMK), 87-100 Torun, Poland (e-mail: a.cichocki@riken.jp).

Index Terms — **P300 speller, brain computer interface, WLDA, generic model set, matching method, online training strategy.**

I. INTRODUCTION

AS A field of technology, brain computer interfacing (BCI) provides a way to communicate between the brain and an external devices without the involvement of muscles and peripheral nerves [1]–[3]. BCI technology translates electroencephalography (EEG) signals into control commands, which particularly benefits individuals who find it difficult or impossible to communicate, for example individuals with amyotrophic lateral sclerosis (ALS) or who are in the locked-in state (LIS) [4]–[6]. In this case, three type of signals are most widely used from the EEG: the event-related potential (ERP, typically, the P300 ERP is most frequently used), the steady-state visual evoked potential (SSVEP), and the event-related desynchronization (ERD), which is associated with motor imagery (MI). In the present study the P300-based BCI system is explored.

In recent years, the P300-based BCI system, which was firstly demonstrated by Farewell and Donchin in 1988. Reference [7] has been widely investigated. The P300 component is the largest positive detection in the EEG after the stimulus onset, with a latency around 300ms [8]. In addition, P300 BCI systems can also evoke P100, N200, and N400 ERP components [9], [10]. In general, the performance of P300 BCIs can be assessed by classification accuracy and information transfer rate (ITR) [11]. In addition, researchers have designed numerous novel visual stimulus patterns and optimized classification algorithms to improve the performance of these systems [9], [11]–[13]. These studies have made significant progress in improving the performance of the P300 BCI. However, many of these approaches have not taken user-friendliness into account, which could cause user exhaustion. In an effort to reduce visual fatigue of users, Xu *et al.* [14] developed a new BCI speller based on miniature asymmetric visual evoked potentials (aVEPs), which demonstrated the feasibility of using very small lateral visual stimuli to implement an efficient BCI system. In addition, many previous research methods required long time training and offline calibration data. As an example, Townsend *et al.* [15] reported that participants experienced two sessions including a calibration

Fig. 1. The framework of the proposed EEG modeling and classification technique.

Fig. 2. The interface of the P300-based character speller.

phase and an online test phase on different days within the duration of a week, which required twelve minutes of calibration data to train their classifier, while Jin *et al.* [9] took an online and three offline runs on the same day, with each offline run four minutes. Therefore, reducing calibration time is a significant element in minimizing user exhaustion and maximizing the utility of BCI systems.

Vo *et al.* [16] proposed that the subject-independent BCI system did not require new users to perform the training stage. Lu *et al.* [17] pointed out that the key to reducing calibration time is that making effective use of inter-participant information and building generic models. However, recent studies [18], [19] have suggested that large individual differences exist in the P300 across participants, especially in amplitude and latency. As a consequence, it is challenging to build a generic model to reduce calibration time.

Lu *et al.* [17] proposed an online model adaptation technique without any calibration for a new user to achieve excellent performance. This approach made use of inter-participant information to generate a generic model. Nevertheless, it required many training iterations to accurately update the model, which was time-consuming. Jin *et al.* [20] combined a generic model built from data recorded from ten participants with an online training strategy, which had considerable success in improving performance in generic model building. Waytowich *et al.* [21] combined trained classifiers with other individual data by a spectral-meta learning method. This approach was shown to reduce the feature dimension by using a spatial filter, requiring fewer samples to train the model. Xu *et al.* [22] investigated how the incorporation of other subject's data could improve the subject-specific classifier accuracy for P300-spellers by using a classifier calibration strategy, Weighted Ensemble Learning Generic Information (WELGI). Furthermore, Qi *et al.* [23] provided a novel method which applied the Riemannian distance measurement to select similar samples from others to train the model. This not only reduced calibration times, it also resulted in a better classification performance.

Although the above studies have addressed the problem of training generic models in BCI to a certain degree, they were either based on a relatively small sample size or did not take individual differences into account. To data, designing an efficient calibration strategy is still a significant challenge in P300-based BCI systems.

In the current work, we propose the concept of a generic model set, and combined a model matching method with

an online training strategy to decrease calibration time. The framework of the proposed strategy is shown in Fig. 1. First, we extracted the P300 feature from EEG recorded from 116 participants during offline processing of the dataset. Second, to account for the variations of the P300 across participants, the processed features were clustered by a k-means clustering algorithm before being used to build a generic model set by WLDA. Due to the large individual differences between participants, this allowed us to record enough information to identify the matching model. Therefore, some labeled data from new participants are acquired to calibrate the model. The labeled data were then used to calibrate a generic model set via a process of online-LDA training. In the end, new participants completed an online spelling task using two models. Specifically, the best and worst matching models were both evaluated to illustrate the advantage of our approach..

II. THE P300 SPELLER DATASET

A. Dataset Description

The P300 dataset used in the current study is obtained from the 2018 World Robot Conference (WRC) BCI Contest. The dataset consists of 116 participants (72 males, 44 females) who were all undergraduate or postgraduate university students ranged in age from 17 to 23 years old at the time of data collection. This data was recorded with a 64-channel wireless EEG acquisition system at a sample rate of 1000Hz. The paradigm used to evoke P300 ERPs while recording the dataset consisted of a 6×6 matrix, sometimes referred to as a conventional oddball paradigm, which contained 26 characters and 10 numbers. The interface is shown in Fig. 2. A row or column was selected at random and intensified in brightness for 80ms every 160ms. Each participant was asked to undertake two runs of the experiment, including an offline run and an online run. The offline run included 36 epochs/characters (A-Z, 1-9, 0) with 4 trials per epoch. In our case, we only consider the EEG signals recorded during the offline training part of the experiment. The experimental flow diagram used to record the dataset is shown in Fig. 3.

B. Feature Extraction

Feature extraction is a key process to reduce dimensionality in the dataset to avoid overfitting and reduce computation time. In the P300-based BCI system, a band pass filter between
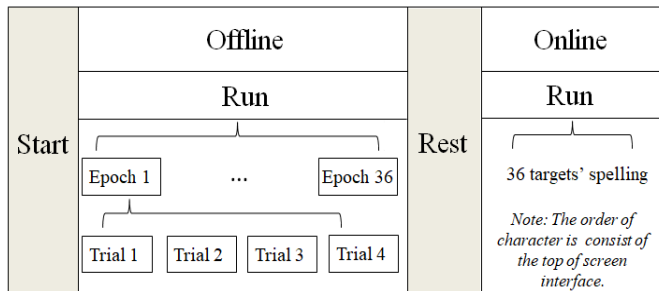
Fig. 3. The experimental flow diagram of dataset.

0.2Hz to 40Hz was applied to reduce high frequency noise. The filtering algorithm we applied was a 5th order Butterworth filter [24]. In order to eliminate the impact of electrical noise, a 5th order IIR comb notch filter at 50Hz was then applied. To decrease the dimensionality of the data and the complexity of model, the filtered EEG data was then down-sampled from 1000Hz to 50Hz (i.e. by a factor 20). Since the first 600ms of EEG after target stimulus presentation was extracted from each channel [25], the size of the feature vector is 10×30 where 10 is the number of channels we used in the whole experimental process, and 30 is the number of samples per channel.

III. Materials and Methods

We built a generic model set derived from 116 participants' data and explored the differences across participants. Then, to evaluate the efficacy of this generic model set in training a BCI, 12 new participants were invited to test the performance of the model. In the following section, we describe the experimental process and present the generic model set building method and matching technique.

A. Experimental Process

Twelve healthy participants (S1-S12), eight of whom were male and four female, aged 22–25 years, with a mean age of 24, and with normal or corrected to normal vision participated in this experiment. Before the experiment began, all participants were asked to sign a consent form, which the local ethics committee approved. After completion of the study, they were given 100 RMB as a reward. All participants' native language was Mandarin Chinese, and all participants were familiar with the Western characters used in the display. Seven of the twelve participants had participated in a BCI experiment previously.

In the experiment, we instructed participants to sit approximately 105 cm in front of an LCD computer monitor, 23.6-in. Dell VG2401 series. And the computer processor (CPU) is an Intel Xeno(R) E5-2603 v3, processor running at 1.6GHz with 16GB of RAM. Participants were instructed to relax themselves and to avoid unnecessary movement when we were recoding their EEG.

The EEG acquisition system included an EEG cap, wireless EEG amplifier, and wireless router. The 64-channel wireless amplifier connected to the electrode cap was used to record EEG at a sample rate of 1000Hz. The amplifier was connected to the amplifier control router via the WIFI. In this paper, EEG
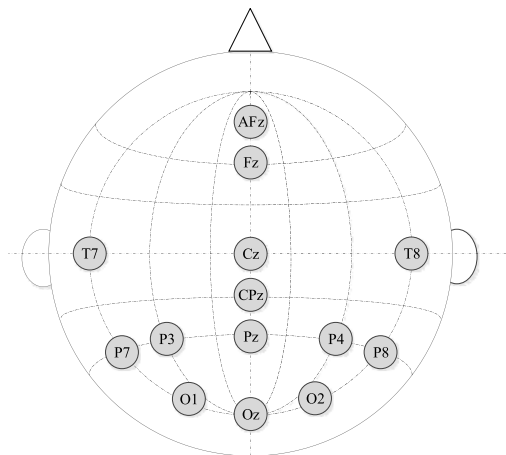

Fig. 4. The configuration of the selected electrode positions.

signals were recorded via ten electrodes placed at positions Fz, Cz, Pz, P3, P4, P7, P8, O1, Oz, and O2 based on the international 10-20 system. These electrodes were chosen as they lie over areas of the brain which are associated with vision [26]. The ground electrode was placed at AFz, while the reference electrodes were placed at locations CPz, T7 and T8 [27], [28]. That is, data from each channel was referenced online to channel CPz and re-referenced offline to the mean of channel T7 and T8. The impedance was kept below 10kΩ in the experiments [29]. Fig. 4 illustrates the configuration of the selected electrode positions.

In this study, three cases were considered.

Case 1 (Typical Model): Participants were asked to complete two runs, an offline run and an online run. The offline run covers 36 target spelling tasks with 4 trials per target. The model was trained using a linear discriminant analysis (LDA) approach and used to spell 20 characters online. Offline calibration time was 276 s.

Case 2 (The Best Matching Model): In this case, in order to record enough information to identify the matching model, participants first need to spell 7 characters (i.e. 1, J, X, H, D, G, and 7; see Fig. 2.) to calibrate or train the generic model online. Each character has 6 trials and it takes 81s to train the model. Participants were then instructed to complete an online spelling run of 20 characters using the best matching generic model.

Case 3 (The Worst Matching Model): In order to compare the differences between generic models and illustrate the validity of the matching rule, we also applied the worst calibrated generic model in an online spelling system.

In order to ensure consistency, we used 20 characters for spelling in the online experiment, and the online spelling order of the 20 characters remained the same in all three cases, i.e. regardless of in which case a given participant was asked to attempt the letters they were asked to spell remained the same. After each run, the participants were given a rest for 4-5 minutes.

B. Algorithms

1) Feature Clustering: Since there are obvious temporal domain characteristics in the P300 potential, we took the filtered EEG data from 116 participants as features. Due to

the high dimensionality of the EEG [30], we decided to use the PCA dimensionality reduction algorithm to decrease the feature set, to save computation, and to reduce preprocessing time. After reducing feature dimension, a clustering algorithm was used for feature processing of the EEG to account for individual differences. Notably, a k-means clustering algorithm was used due to its good convergence rate [31].

a) PCA: The principal component analysis (PCA) [32] is an unsupervised dimensionality reduction method, which is widely applied in many field. In this work, we use PCA to reduce the dimensionality of the EEG feature set.

The original data is comprised of the sample vectors of all participants and is represented by $X = \{x_1, \ldots, x_N\}$ where x_i is the row vector of the i_{th} participant and N is the number of participants, that is, 116. There were 518400 (36*4*12*10*30) eigenvalues for each participant, where 36 is the number of characters, 4 is the number of trials, 12 is the number of flashes, 10 is the number of channels, and 30 is the number of samples. The covariance matrix is given as,

$$s = \sum\nolimits_{i=1}^{N} (x_i - \bar{x})(x_i - \bar{x})^T \tag{1}$$

$$\bar{x} = \frac{1}{N} \sum\nolimits_{i=1}^{N} x_i \tag{2}$$

Let $A = x_i - \bar{x}$, then $s = AA^T$, and implement singular value decomposition to get the eigenvalue $\lambda_i (i = 1, \ldots, N)$ and the correspond eigenvector $v_i (i = 1, \ldots, N)$ of the matrix s. Suppose that $\lambda_i (i = 1, \ldots, r)$ are the eigenvalues, which are not zero. The normalized orthogonal eigenvector u_i is:

$$u_i = \frac{1}{\sqrt{\lambda_i}} * A^T * v_i, \quad i = 1, \ldots, r \tag{3}$$

We re-order $i(i = 1, \ldots, r)$ in descending order of the magnitude of the eigenvalues, and adjust the order of u_i accordingly. In this work, the orthonormal eigenvector is $U = \{u_1, \ldots, u_d\}$. We used the PCA function with 'econ' parameter in Matlab in which, if the number of samples N is smaller than the number of eigenvalues, the $N - 1$ principle component is acquired. This can be significantly faster when the number of eigenvalues is larger than N [32], [33]. Consequently, we selected the u_i corresponding to the 115 largest eigenvalues for each participant, that is $d = 115$, and the contribution rate is greater than 95%. It is enough to represent the whole data with fewer features. The transformed data is defined by $Z = X^T U \in \mathbb{R}^{N \times d}$, Where each column of Z (i.e. $z_j \in \mathbb{R}^N, j \in \{1, \ldots, d\}$) is a principal component. It can be found that each principal component is a linear combination of all the original features [34].

b) K-means Clustering Algorithm: The k-means algorithm, is one of the most popular and the simplest clustering algorithms [31]. It works by minimizing the cluster variance. Let $Z = \{z_1, \ldots, z_{115}\}$ be the set of 116 d-dimensional points, and take K as the input to cluster 116 samples into K clusters, $C = \{c_k, k = 1, \ldots, K\}$. Let $\mu_k = \sum_{x_i \in C_k} x_i / n_k$ be the centroid of cluster C_k, where n_k denotes the number of points in C_k. Cluster similarity is measured according to the dissimilarity between a data point and the centroid of the cluster. The squared error between μ_k and the points in cluster

C_k is defined as

$$J(c_k) = \sum\nolimits_{x_i \in C_k} ||x_i - m_k||^2 \tag{4}$$

K-means is determined by minimizing the sum of squared errors over all K clusters,

$$J(C) = \sum\nolimits_{k=1}^{K} \sum\nolimits_{x_i \in C_k} ||x_i - m_k||^2 \tag{5}$$

Different initializations can lead to different final clustering because k-means only converges to a local minima [35]. One way to overcome the local minima is to run the k-means algorithm, for a given K, with multiple different initial partitions and choose the partition with the smallest squared error. In this paper, we applied k-means with $K = 10$ [36], [37] to cluster sample data with the minimization of the sum of squared errors among multiple clusters.

2) Model Online Update and Classification: In this section, we developed a weighted linear discriminant analysis method (WLDA) to obtain offline classification models of every cluster which belongs to the generic model set. In addition, we also applied Online Linear Discriminant Classifiers (OLDA) to enable updates to accommodate changing environments during online training [13].

a) Weighted Linear Discriminant Analysis (WLDA): The WLDA is an extension of LDA, that is, a weighted LDA. The objective of WLDA (the same as Fisher's LDA) is to use hyper-planes to separate the data representing the different classes. Compared with other complex classifiers, LDA has very low computational requirements and processing time, which makes it suitable for online spelling [38], [39]. Moreover, this classifier is simple to use and generally obtains better classification performance and acceptable accuracy [22], [40]. Some improved algorithms have been developed based on LDA, for instance the stepwise LDA (SWLDA) [41], Bayesian LDA (BLDA) [42], and spatial temporal discriminant analysis (STDA) [43]. LDA can achieve better performance in some situations, for example, Syan and Harnarinesingh [44] indicated that the LDA classifier achieved the best performance in classifying unseen P300 spatiotemporal features in both the single-trial (74.19%) and multi-trial case (100%). Raudys and Jain [45] proposed that, if there is a small training set, simple techniques with few parameters should be used, such as LDA, which is why we use LDA in our work.

In the current work, WLDA we proposed is used to classify the P300 feature of multiple participants. We assigned a weight for each feature of each participant according to the contributions of each participant to the result. In this system, we only take the two-class circumstance to train offline data. Let $x_i \in R^D$ be the input vectors of the classifier, and $y_i \in \{-1, 1\}$ be the label vector, which represents class 1 G_1 and class 2 G_2 respectively. Finally, the LDA weight vector we require can be defined by,

$$w = S_w^{-1}(m_1 - m_2) \tag{6}$$

$$m_k = \frac{1}{N_k} \sum\nolimits_{i \in G_k} x_i, \quad k = 1, 2 \tag{7}$$

where N_k denotes the number of classes k, and S_w^{-1} is the inversion of the within-scatter matrix.

$$S_w = \sum_{k=1}^{2} \sum_{i \in G_k} (x_i - m_k)(x_i - m_k)^T \tag{8}$$

During the model construction for each cluster the input vectors is taken from multiple participants who are grouped into the same cluster. Given this, we decided to assign different weights according to the contribution rate of the participants to the results of the offline classification. In this section of the work, we considered only the between-class scatter matrix.

First, we used feature of ten clusters to compute corresponding discriminant vector $w_k(k = 1, \ldots, 10)$. Clustered participants are defined by $c_k = \{x_i, i = 1, \ldots, n_k\}$, where n_k is the number k_{th} cluster.

Second, we used the trained vector w_k to classify x_i, $(i = 1, \ldots, n_k)$ and the offline accuracy acc_i was used to get a feature weight e_i for every participant, as shown in equation (9).

$$e_i = \frac{1}{n_k} \sum_{j=1}^{n_k} acc_i \tag{9}$$

Finally, we used a weighted feature vector $c_k = \{x_i e_i, i = 1, \ldots, n_k\}$ to re-train the discriminant vector $w_k'(k = 1, \ldots, 10)$. The WLDA weighted vector can be described by equation (10).

$$w' = S_W^{-1}(m_1' - m_2') \tag{10}$$

where

$$m_k' = \sum_{i \in G_k} x_i e_i, \quad k = 1, 2 \tag{11}$$

b) Online Linear Discriminant Analysis: Online Linear Discriminant Analysis (OLDA) was proposed by Kuncheva and Plumpton [13]. The online training strategy is based on the inverse of the common covariance matrix, updated with the Sherman–Morrison–Woodbury formula. Moreover, it is simple and can reduce the amount of calculation, which makes it suitable for online BCI systems.

C. Practical Bit Rate

In this work, we applied two methods of measuring bit rate, practical bit rate (PBR) and raw bit rate (RBR), to describe the speed and the number of online spellings made by the ERP-BCI system [10]. The practical bit rate was used to estimate the speed of the system in a real-world setting. Though both methods were shown in TABLE I, in general, we only analyzed the RBR in this work. The practical bit rate is defined as $PBR = RBR * (1 - 2P)$, where RBR is the raw bit rate and P is the error rate of online spelling in the system. Both PBR and RBR include the time to make a selection. The raw bit rate, which also includes selection time, represents the online transfer rate of the P300-based online spelling system.

D. Adaptive System Setting

In order to improve the performance of the system, an adaptive strategy was added into the online spelling system [42]. Specifically, the number of trials required to select each

---

Algorithm 1 Algorithm Process

Offline training part:
Input Labeled EEG signal from 116 training participants is observed. **Output** Ten generic models built from ten clusters of participants
Step 1: Extract feature vectors $x_i(i = 1,\ldots,116)$ from 116 training participants' EEG signals where x_i is 2-D feature and the corresponding label vector is denoted by $y_i(i = 1,\ldots,116)$.
Step 2: Reduce the dimensionality of the filtered data $X = \{x_1,\ldots,x_{116}\}$ by PCA. Then, cluster the data of the 116 training participants into 10 clusters by the k-means algorithm.
Step 3: Construct ten WLDA classifiers from the feature vector from the ten clusters of participants.
Online testing part:
Input Ten classifiers and EEG signal of twelve novel test participants are observed.
Output The feedback of online spelling for 20 characters from the best and worst matching model.
Step 1: Ask the test participants to attempt to spell 7 characters online and, at the same time, use the OLDA to calibrate the model, then match the model which corresponds to maximal and minimal accuracy.
Step 2: Use the best and the worst matching model after online training to complete online adaptive spelling for 20 characters respectively.

---

character is related with the classifier output, and the classifier output can be got after each trial when using the online system. In particular, when the classifier recognized the same character after two subsequent trials, there are no new flashes and the recognized character is be presented on the screen as feedback. If the number of trials for a given character reaches 10 with no consensus reached, the classifier will automatically choose the target recognized on the final trial. Note that the score of each trial is the sum of the previous one. For example, suppose that '1' is the target character, which the classifier recognized in the first trial. If the character '1' was recognized again in the second trial, the final output will be '1'. We can describe with the formal as $cha(n) = cha(n-1)(1 < n \leq 10)$.

E. Subjective Reports

At the end of each test, each participant was asked two questions. Each question was given a rating in the range 1-5, with a score of 1 indicating the lowest agreement and a score of 5 indicating the highest level of agreement. The two questions were:

Q1: Did you feel tired in the three cases? Please give a score respectively.

Q2: Did you feel annoyed in the three cases? Please give a score respectively.

The questions were asked in Mandarin Chinese.

IV. EXPERIMENTAL RESULTS

In this system, the generic model set was updated by the labeled data of 7 characters from each participant, and

Participants		S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	AVG±STD
Typical model (Case 1)	ACC(%)	85	100	85	90	85	75	65	80	80	90	90	55	81.7±12.1
	RBR	30.8	51.3	39.5	40.9	42.3	27.1	20.3	30.6	27.8	37.4	35.4	21.6	33.7±9.1
	PBR	21.5	51.3	27.6	32.7	29.6	13.5	6.1	18.3	16.7	29.9	28.3	2.2	23.2±13.2
	C-time(s)	276	276	276	276	276	276	276	276	276	276	276	276	276±0.0
	S-time(s)	212	185	180	187	172	200	208	198	212	198	206	168	193.8±15.2
	AVT	3.9	3.2	3.0	3.2	2.8	3.6	3.8	3.5	3.9	3.5	3.7	2.7	3.4±0.4
The best matching model (Case 2)	ACC(%)	80	100	90	75	90	75	65	90	85	60	60	90	80±13.1
	RBR	24.9	44.9	41.5	23.4	33.1	21.8	17.3	32.3	28.9	16.4	16.0	40.3	28.4±10.1
	PBR	14.9	44.9	33.2	11.7	26.5	10.9	5.2	25.9	20.2	3.3	3.2	32.2	19.3±13.4
	C-time(s)	81	81	81	81	81	81	81	81	81	81	81	81	81±0.0
	S-time(s)	229	202	185	221	216	233	233	220	221	221	225	189	216.2±16.0
	AVT	4.3	3.6	3.2	4.1	4.0	4.4	4.4	4.1	4.1	4.1	4.2	3.3	4.0±0.4
The worst matching model (Case 3)	ACC(%)	35	80	70	45	40	30	10	45	40	15	55	55	43.3±20.3
	RBR	6.3	30.1	23.9	10.5	9.0	5.1	0.6	10.9	7.8	1.5	13.4	16.2	11.3±8.7
	PBR	0	18.1	9.5	0	0	0	0	0	0	0	1.3	1.6	2.5±5.6
	C-time(s)	81	81	81	81	81	81	81	81	81	81	81	81	81±0.0
	S-time(s)	233	196	198	216	210	223	227	210	233	223	227	198	216.2±13.6
	AVT	4.5	3.6	3.6	4.1	3.9	4.3	4.4	3.9	4.5	4.3	4.4	3.6	4.1±0.4

ACC refers to online accuracy; C-time refers to the total calibration time; S-time refers to the online spelling time; AVT refers to the average number of trials each target; AVG refers to average value; STD refers to the standard deviation.
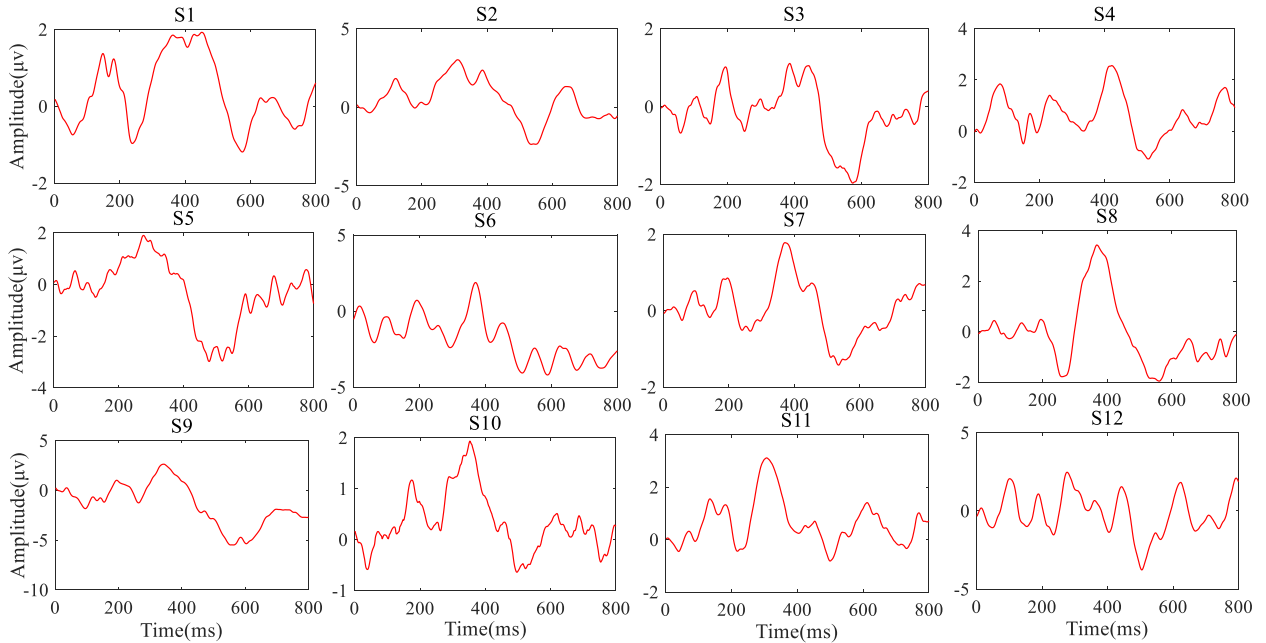
Fig. 5. P300 waveform of twelve participants at channel Cz.

then the updated generic model was used to classify EEG related to 7 characters respectively. In this case, the matching strategy used was that the classification result of each model was regarded as the matching result. In order to explore the individual differences in time domain across participants, especially in amplitude and latency, Fig. 5 illustrates the P300 waveform variations at channel Cz for twelve healthy participants calculated through averaging 288 EEG flashes of the targets (36 characters $*$ 4 trials $*$ 2 flashes). Fig. 6 shows matching accuracy of the generic model set for the test set of 12 new participants. The green bar corresponds to the best matching model and the red bar corresponds to the worst matching model. Due to individual variations, we can find that some participants matched the different best matching model, while some participants shared the same best matching model.
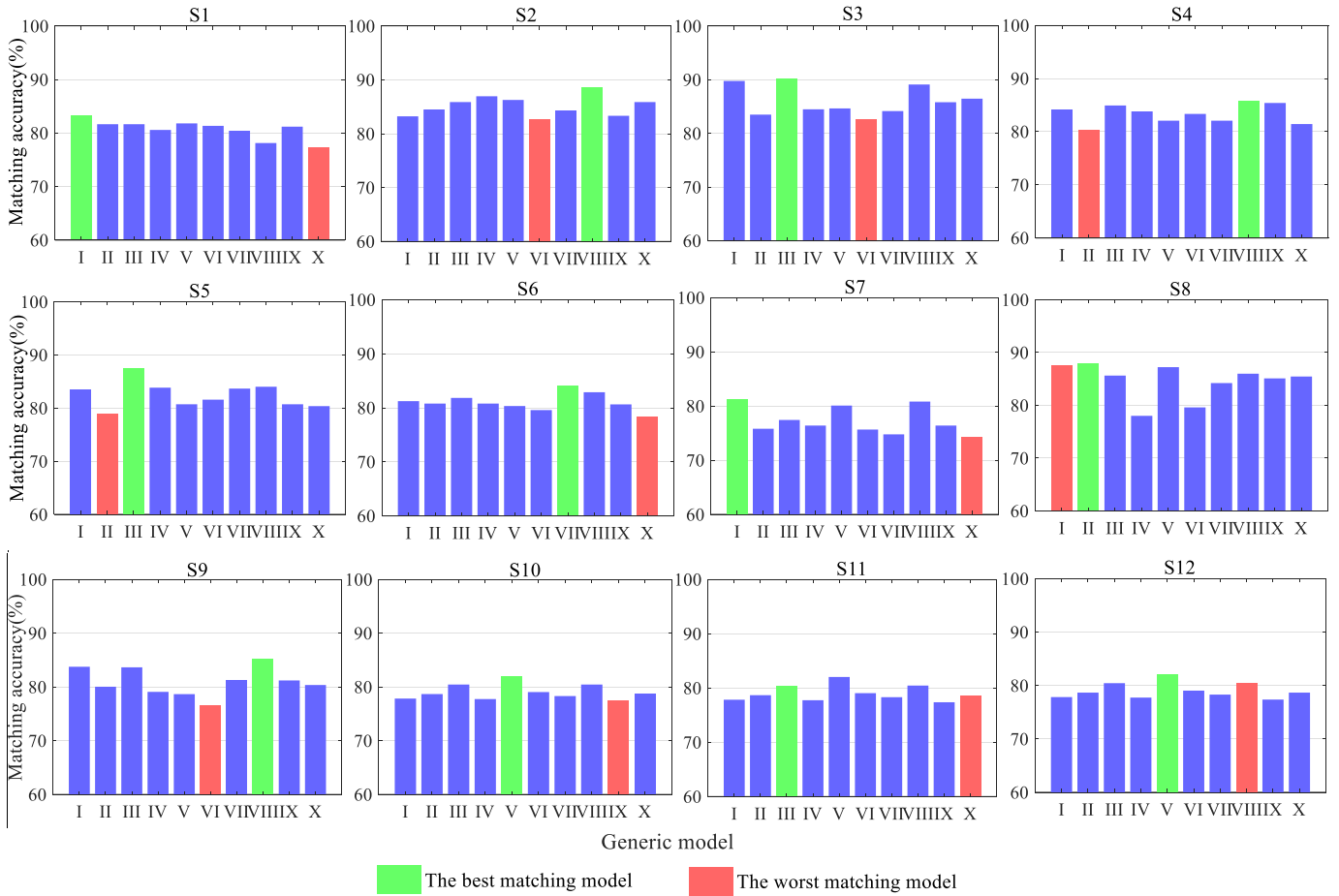
Fig. 6. The matching result of the model set for 12 participants. Roman numerals (I - X) on the horizontal axis indicate which matching model is used. The green bar corresponds to the best matching model and red bar corresponds to the worst matching model. (Note: the accuracy was 66.66% (2/3) when all labels were wrong, so we set the minimum value of the vertical axis to 60%).

TABLE I describes the online classification accuracy, raw bit rate, practical bit rate, calibration time, online spelling time and the average number of trials in three cases (typical model, the best, and the worst matching model). In terms of online accuracy, we can see that, when using the best matching model, the accuracy of twelve participants can reach a mean accuracy of 80%. As for the online spelling time of a run, we use 'S-time' of three cases in TABLE I to describe, where a 3.2s interval between characters was included. Although participant S2 showed superior performance in all three cases, this participant reported that the generic model was "a good fit" and that the spelling task was easy to complete.

Participants S10 and S11, can achieve outstanding results in case 1, while using our proposed best matching model they obtained a mean accuracy of 60%. At the same time, their matching results were lower than other participants. This may be a result of relatively low matching result. In other words, their ERP data does not match well with the data from the training dataset. In addition, participant S12 displayed poor performance using the typical model, while the online classification result reached an accuracy of 90% using the best matching generic model.

In summary, most participants using the best matching method can obtain the same performance level as the typical model method. From the perspective of the calibration time in the three cases we proposed, it is obvious that using the best matching model method we proposed has results in a reduction in calibration time of 70.7% compared to the typical model method.

In order to make sure that the classification accuracy and raw bit rate can be analyzed, data was statistically tested via a One-Sample Ryan-Joiner test for normal distributions. A repeated-measures ANOVA (RM-ANOVA) was chosen to evaluate the effect of stimuli paradigm on the results. Mauchly's test of sphericity was executed before the analysis. If the assumption was broken, Greenhouse-Geisser correction was performed to adjust the degrees of freedom. In the end, we applied Bonferroni correction in a post hoc test. The alpha level reached 0.05 after Bonferroni correction [10].

We explore the P300 variability by using three models. Typical models of the twelve test participants were built from their offline run. In addition, the best and the worst matching models built from the dataset were matched and trained online respectively. A one-way repeated measures ANOVA was applied to show the classification accuracy
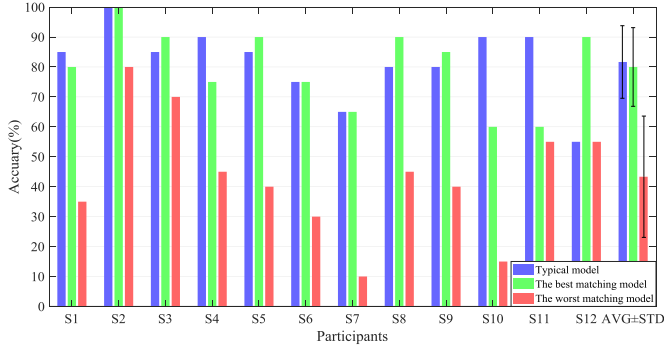
Fig. 7. The spelling accuracies achieved with the three cases using the typical model, the best matching model, and the worst matching model method.
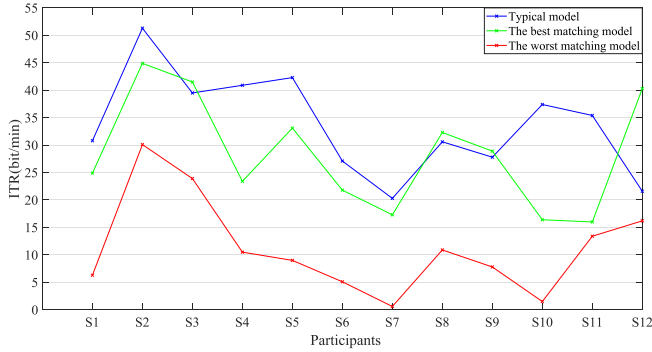

Fig. 8. The Spelling ITR in three conditions using typical model, the best matching model, and the worst matching model method.

$(F(1.827, 20.102) = 36.631, p < 0.05, Eta^2 = 0.769)$ and raw bit rate $(F(1.352, 14.873) = 44.823, p < 0.05, Eta^2 = 0.803)$ difference across three kinds of models under three strategies. In TABLE I, there are no statistically significant differences in classification accuracy ($p > 0.05$) and raw bit rate ($p > 0.05$) between the typical model method and the best matching model method, which demonstrates that using the best matching model could obtain the same online accuracy and raw bit rate as the typical model. TABLE II gives a description of the p values of the significance of the difference under pairwise comparison in the three cases. Bonferroni correction for multiple comparisons was applied.

Furthermore, the typical model takes approximately seven minutes to train, while the matching model took less than one and a half minutes, which spent less than 70.7% of the calibration time it previously took. In other words, it significantly reduced calibration time ($p < 0.05$). Fig. 7 shows the online accuracy using three kinds of models. The blue bar indicates results obtained with the typical model, the green bar indicates the best matching model, and red bar indicates the worst matching model. Fig. 8 shows the practical bit rates achieved using three kinds of models. The blue line indicates the typical model, the green the best matching model, and the red the worst matching model.

As Fig. 7 and Fig. 8 show, it is easily noted that the accuracy ($p < 0.05$) and raw bit rate ($p < 0.05$) of the best matching model method is significantly higher than the worst matching

p value	ACC	RBR
Case 1 vs Case 2	0.749	0.121
Case 2 vs Case 3	$3.73*10^{-6}$	$6.24*10^{-7}$
Case 1 vs Case 3	$3.19*10^{-5}$	$1.15*10^{-6}$

ACC is online accuracy; RBR is raw bit rate;

model method and the accuracy ($p = 0.749$) and raw bit rate ($p = 0.121$) of the typical model method is not significantly different to the results achieved with the best matching model method. In addition, the typical model method has significantly better performance than the worst matching model method in accuracy ($p < 0.05$) and raw bit rate ($p < 0.05$). The results indicate that the EEG signal of a new participant, who has the same P300 characteristics as some of the participants in the training dataset, can expect an improvement in classification performance.

TABLE III displays the feedback answers of the 12 participants who attempted the three cases. For the sake of distinguishing the differences, a Friedman nonparametric test was applied to explore the differences in feedback. According to users' feedback, we find that there are significant trends ($\chi^2 = 7.091, p < 0.05$) toward tiredness of all participants in the experiment. In addition, in terms of the annoyance, there was no significant trend ($\chi^2 = 4.389, p > 0.05$) in the experiment. According to the study of Möckel, long time in a cognitive task causes mental fatigue [46]. This means that our proposed method which shortened the calibration time may decrease mental fatigue of participants and improve practicability of the system [43].

V. DISCUSSION

P300-based BCI systems have been widely investigated in over recent years and are beginning to be used in clinical applications [4], [5]. However, despite significant progress in the operation of the P300 BCI, many BCIs still spend considerable time on offline training/calibration. As a consequence, developing an efficient calibration strategy is an important challenge to solve in P300-based BCI development.

In general, a calibration strategy should take the classification accuracy and calibration time into consideration simultaneously. Previous studies have shown that using inter-participant information to build a model can waste time. A better approach is to strategically update the model [10], [21], [47]. These studies attempted to train an initial classifier in advance and update it using an appropriate updating strategy. Given the aim of reducing the system calibration time, we adopted the concept of a generic model set and matched the samples of new participants with those of participants in the training set with the same ERP characteristics.

In the data processing stage, the EEG data of 116 training participants was used to ensure that we had enough samples to build a model set that improved the generalization of the system. During online testing, participants were required to complete a seven characters spelling task. This allowed

TABLE III
PARTICIPANTS' RESPONSES TO TWO QUESTIONS FOR EACH OF THE THREE CASES

Score		S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	Average
Q1: tiredness	Case1	2	4	3	2	3	4	4	3	3	3	2	3	3
	Case2	3	2	2	2	1	3	3	2	2	3	3	2	2.3
	Case3	4	2	2	2	2	4	3	3	2	3	4	3	2.8
Q2: annoyance	Case1	3	4	3	3	4	3	4	3	3	2	3	4	3.3
	Case2	3	1	2	3	3	3	4	2	3	3	3	2	2.7
	Case3	4	2	2	3	3	4	3	4	4	3	4	3	3.3

us to record enough information to identify the matching model. As a whole, the system works very well. However, according to feedback from our participants, the system still has shortcomings. Specifically, there is a short stimulus onset asynchrony (SOA) and unobvious stimulus flash. In this system, the SOA is set to be 160ms, which is too short for many participants to count accurately. In work reported by Allison and Pineda [48] faster SOAs were shown to result in a considerable number of errors. Participant S7 reported that he did not see each individual flash. In this case, the amplitude of the P300 recorded from participant S7 was reduced since the P300 occurs only in response to detected targets. This also resulted in lower accuracy.

In terms of stimuli, this system used a white/gray flicker matrix as a visual stimulus, which induced discomfort [49]. In addition, Martens *et al.* [50] showed that traditional row/column flashes can cause overlap and refractory effects and negatively impact on the performance of the system. We can see this negative effect from the online spelling results. Dias *et al.* [51] proposed that there are differences in the amplitude and latency of the P300 signal for different age groups. Thus, in order to improve the usage of the system, the age of the participants should be considered in the experiment design. Note, the age group of our participants in the training dataset was consistent with the test participants.

VI. CONCLUSION

In the current study, we proposed the concept of a generic model set and introduced an online training strategy combined with a model matching method. The generic model set including ten models trained by a weighted linear discriminant analysis (WLDA) method. The results prove that most participants using the best matching model achieved the same classification performance as the typical model, which reduced the calibration time by approximately 70.7% and proved the effectiveness of the method. In addition, the results of the worst matching model indicated that not all generic models are suitable for every participant to achieve good performance. In other words, differences exist across generic models.

In future work, we plan to identify optimal parameters and paradigms to build a generic model set. In addition, in order to improve the generality of the model set, we plan to build generic models for specific participant age groups, particularly for older participants and for participants who live with amyotrophic lateral sclerosis (ALS) and are in the locked-in state (LIS).

On the other hand, some participants may have poor performance using BCI systems and may be unfamiliar with how to use it. Thus, we should adopt new ways to improve BCI systems. We will continue to train our generic model set, for example by using neural networks as proposed by Hiraiwa *et al.* [52], to increase the applicability of the system.

REFERENCES

[1] U. Hoffmann, J. M. Vesin, T. Ebrahimi, and K. Diserens, "An efficient P300-based brain-computer interface for disabled subjects," *J. Neurosci. Methods*, vol. 167, no. 1, pp. 115–125, Jan. 2008.

[2] A. Kübler, B. Kotchoubey, J. Kaiser, J. R. Wolpaw, and N. Birbaumer, "Brain-computer communication: Unlocking the locked in," *Psychol. Bull.*, vol. 127, no. 3, p. 358, 2001.

[3] J. Wolpaw, N. Birbaumer, D. McFarland, G. Pfurtscheller, and T. Vaughan, "Brain-computer interfaces for communication and control," *Clin. Neurophys.*, vol. 113, no. 6, pp. 767–791, Jun. 2002.

[4] Y. Miao *et al.*, "An ERP-based BCI with peripheral stimuli: Validation with ALS patients," *Cogn. Neurodyn.*, 2019, doi: 10.1007/s11571-019-09541-0.

[5] F. Nijboer *et al.*, "A P300-based brain–computer interface for people with amyotrophic lateral sclerosis," *Clin. Neurophysiol.*, vol. 119, no. 8, pp. 1909–1916, Aug. 2008.

[6] A. Kubler and N. Neumann, "Brain-computer interfaces—The key for the conscious brain locked into a paralyzed body," *Boundaries Consciousness, Neurobiol. Neuropathol.*, vol. 150, pp. 513–525, 2005.

[7] L. A. Farwell and E. Donchin, "Talking off the top of your head: Toward a mental prosthesis utilizing event-related brain potentials," *Electroencephalograph. Clin. Neurophysiol.*, vol. 70, no. 6, pp. 510–523, Dec. 1988.

[8] L. Bianchi, C. Liti, and V. Piccialli, "A new early stopping method for P300 spellers," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 8, pp. 1635–1643, Aug. 2019.

[9] J. Jin *et al.*, "The changing face of P300 BCIs: A comparison of stimulus changes in a P300 BCI involving faces, emotion, and movement," *PLoS ONE*, vol. 7, no. 11, Nov. 2012.

[10] J. Jin, I. Daly, Y. Zhang, X. Y. Wang, and A. Cichocki, "An optimized ERP brain-computer interface based on facial expression changes," *J. Neural Eng.*, vol. 11, no. 3, p. e49688, Jun. 2014.

[11] T. Kaufmann, S. M. Schulz, C. Grunzinger, and A. Kubler, "Flashing characters with famous faces improves ERP-based brain-computer interface performance," *J. Neural Eng.*, vol. 8, no. 5, Oct. 2011, Art. no. 056016.

[12] J. Jin, H. H. Zhang, I. Daly, X. Y. Wang, and A. Cichocki, "An improved P300 pattern in BCI to catch user's attention," *J. Neural Eng.*, vol. 14, no. 3, Jun. 2017, Art. no. 036001.

[13] L. I. Kuncheva and C. O. Plumpton, "Adaptive learning rate for online linear discriminant classifiers," in *Proc. Joint IAPR Int. Workshops Stat. Techn. Pattern Recognit. (SPR) Struct. Syntactic Pattern Recognit. (SSPR)*, vol. 5342. Berlin, Germany: Springer, 2008, pp. 510–519.

[14] M. Xu, X. Xiao, Y. Wang, H. Qi, T.-P. Jung, and D. Ming, "A brain–computer interface based on miniature-event-related potentials induced by very small lateral visual stimuli," *IEEE Trans. Biomed. Eng.*, vol. 65, no. 5, pp. 1166–1175, May 2018.

[15] G. Townsend, J. Shanahan, D. B. Ryan, and E. W. Sellers, "A general P300 brain–computer interface presentation paradigm based on performance guided constraints," *Neurosci. Lett.*, vol. 531, no. 2, pp. 63–68, Dec. 2012.

[16] K. Vo, T. Pham, D. N. Nguyen, H. H. Kha, and E. Dutkiewicz, "Subject-independent ERP-based brain–computer interfaces," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 4, pp. 719–728, Apr. 2018.

[17] S. Lu, C. Guan, and H. Zhang, "Unsupervised brain computer interface based on intersubject information and online adaptation," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 17, no. 2, pp. 135–145, Apr. 2009.

[18] R. Y. Emmerson, R. E. Dustman, D. E. Shearer, and C. W. Turner, "P3 latency and symbol digit performance correlations in aging," *Exp. Aging Res.*, vol. 15, no. 3, pp. 151–159, 1989.

[19] J. Polich, "On the relationship between EEG and P300: Individual differences, aging, and ultradian rhythms," *Int. J. Psychophysiol.*, vol. 26, nos. 1–3, pp. 299–317, 1997.

[20] J. Jin, E. W. Sellers, Y. Zhang, I. Daly, X. Wang, and A. Cichocki, "Whether generic model works for rapid ERP-based BCI calibration," *J. Neurosci. Methods.*, vol. 212, no. 1, pp. 94–99, Jan. 2013.

[21] N. R. Waytowich, V. J. Lawhern, A. W. Bohannon, K. R. Ball, and B. J. Lance, "Spectral transfer learning using information geometry for a user-independent brain-computer interface," *Front. Neurosci.*, vol. 10, p. 430, Sep. 2016.

[22] M. P. Xu *et al.*, "Incorporation of inter-subject information to improve the accuracy of subject-specific P300 classifiers," *Int. J. Neural Syst.*, vol. 26, no. 3, 2016, Art. no. 1650010.

[23] H. Qi, Y. Xue, L. Xu, Y. Cao, and X. Jiao, "A speedy calibration method using Riemannian geometry measurement and other-subject samples on a P300 speller," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 3, pp. 602–608, Mar. 2018.

[24] B. Wu *et al.*, "A virtual Chinese keyboard BCI system based on P300 potentials," *Acta Electronica Sinica*, vol. 37, no. 8, pp. 1733–1745, 2009.

[25] Z. H. Gu, Z. B. Chen, J. T. Zhang, X. C. Zhang, and Z. L. Yu, "An online interactive paradigm for P300 brain–computer interface speller," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 2, pp. 152–161, Feb. 2019.

[26] N. Gao, W. Zhai, J. Lu, S. Lu, and Y. Tian, "Intelligent wheelchair research based on SSVEP brain-computer interface," in *Proc. BIBE*, Jun. 2018, pp. 1–5.

[27] M. J. Alhaddad, "Common average reference (CAR) improves P300 speller," *Int. J. Eng. Technol.*, vol. 2, no. 3, p. 21, 2012.

[28] X. Huang, E. Yin, Y. Wang, R. Saab, and X. Gao, "A mobile eeg system for practical applications," in *Proc. IEEE Global Conf. Signal Inf. Process. (Globalsip)*, Nov. 2017, pp. 995–999.

[29] Y. Yu, Y. D. Liu, E. W. Yin, J. Jiang, Z. T. Zhou, and D. W. Hu, "An asynchronous hybrid spelling approach based on EEG-EOG signals for Chinese character input," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 6, pp. 1292–1302, Jun. 2019.

[30] J. Feng *et al.*, "Towards correlation-based time window selection method for motor imagery BCIs," *Neural Netw.*, vol. 102, pp. 87–95, Jun. 2018.

[31] B. Nguyen and B. De Baets, "Kernel-based distance metric learning for supervised *k*-means clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 10, pp. 3084–3095, Oct. 2019.

[32] I. Jolliffe, *Principal Component Analysis*. Springer, 2011, pp. 1094–1096.

[33] J. E. Jackson, *A User's Guide to Principal Components*. Hoboken, NJ, USA: Wiley, 2005.

[34] S. Yi, Z. He, X.-Y. Jing, Y. Li, Y.-M. Cheung, and F. Nie, "Adaptive weighted sparse principal component analysis for robust unsupervised feature selection," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published.

[35] A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern Recognit. Lett.*, vol. 31, no. 8, pp. 651–666, Jun. 2010.

[36] K. Krishna and M. N. Murty, "Genetic K-means algorithm," *IEEE Trans. Syst. Man, Cybern. B, Cybern.*, vol. 29, no. 3, pp. 433–439, Jun. 1999.

[37] K. R. Zalik, "An efficient k'-means clustering algorithm," *Pattern Recognit. Lett.*, vol. 29, no. 9, pp. 1385–1391, Jul. 2008.

[38] Z. Amini, V. Abootalebi, and M. T. Sadeghi, "Comparison of performance of different feature extraction methods in detection of P300," *Biocybern. Biomed. Eng.*, vol. 33, no. 1, pp. 3–20, 2013.

[39] V. Bostanov, "BCI competition 2003-data sets Ib and IIb: Feature extraction from event-related brain potentials with the continuous wavelet transform and the t-value scalogram," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 1057–1061, Jun. 2004.

[40] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi, "A review of classification algorithms for EEG-based brain–computer interfaces," *J. Neural Eng.*, vol. 4, no. 2, pp. R1–R13, Jun. 2007.

[41] E. Yin, Z. Zhou, J. Jiang, F. Chen, Y. Liu, and D. Hu, "A speedy hybrid BCI spelling approach combining P300 and SSVEP," *IEEE Trans. Biomed. Eng.*, vol. 61, no. 2, pp. 473–483, Feb. 2014.

[42] J. Jin *et al.*, "An adaptive P300-based control system," *J. Neural Eng.*, vol. 8, no. 3, Jun. 2011, Art. no. 036006.

[43] Y. Zhang, G. Zhou, Q. Zhao, J. Jin, X. Wang, and A. Cichocki, "Spatial-temporal discriminant analysis for ERP-based brain-computer interface," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 21, no. 2, pp. 233–243, Mar. 2013.

[44] C. S. Syan and R. E. S. Harnarinesingh, "Comparison of pre-processing and classification techniques for single-trial and multi-trial P300-based brain computer interfaces," *Amer. J. Appl. Sci.*, vol. 7, no. 9, pp. 1219–1225, 2010.

[45] S. J. Raudys and A. K. Jain, "Small sample size effects in statistical pattern recognition: Recommendations for practitioners," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 3, pp. 252–264, Mar. 1991.

[46] T. Mockel, C. Beste, and E. Wascher, "The effects of time on task in response selection—An ERP study of mental fatigue," *Sci. Rep.*, vol. 5, Jun. 2015, Art. no. 10113.

[47] B. Rivet, H. Cecotti, M. Perrin, E. Maby, and J. Mattout, "Adaptive training session for a P300 speller brain–computer interface," *J. Physiol.-Paris*, vol. 105, nos. 1–3, pp. 123–129, Jan./Jun. 2011.

[48] B. Z. Allison and J. A. Pineda, "Effects of SOA and flash pattern manipulations on ERPs, performance, and preference: Implications for a BCI system," *Int. J. Psychophysiol.*, vol. 59, no. 2, pp. 127–140, Feb. 2006.

[49] K. Takano, T. Komatsu, N. Hata, Y. Nakajima, and K. Kansaku, "Visual stimuli for the P300 brain–computer interface: A comparison of white/gray and green/blue flicker matrices," *Clin. Neurophysiol.*, vol. 120, no. 8, pp. 1562–1566, Aug. 2009.

[50] S. M. M. Martens, N. J. Hill, J. Farquhar, and B. Schölkopf, "Overlap and refractory effects in a brain–computer interface speller based on the visual P300 event-related potential," *J. Neural Eng.*, vol. 6, no. 2, Apr. 2009.

[51] N. S. Dias, P. M. Mendes, and J. H. Correia, "Subject age in P300 BCI," in *Proc. 2nd Int. IEEE EMBS Conf. Neural Eng.*, Mar. 2005, pp. 579–582.

[52] A. Hiraiwa, K. Shimohara, and Y. Tokunaga, "EEG topography recognition by neural networks," *IEEE Eng. Med. Biol. Mag.*, vol. 9, no. 3, pp. 39–42, Sep. 1990.