An EEG-Based BCI System for 2-D Cursor Control by Combining Mu/Beta Rhythm and P300 Potential

Yuanqing Li*, Jinyi Long, Tianyou Yu, Zhuliang Yu, Chuanchu Wang, Haihong Zhang, and Cuntai Guan, *Senior Member, IEEE*

***Abstract*—Two-dimensional cursor control is an important and challenging issue in EEG-based brain–computer interfaces (BCIs). To address this issue, here we propose a new approach by combining two brain signals including Mu/Beta rhythm during motor imagery and P300 potential. In particular, a motor imagery detection mechanism and a P300 potential detection mechanism are devised and integrated such that the user is able to use the two signals to control, respectively, simultaneously, and independently, the horizontal and the vertical movements of the cursor in a specially designed graphic user interface. A real-time BCI system based on this approach is implemented and evaluated through an online experiment involving six subjects performing 2-D control tasks. The results attest to the efficacy of obtaining two independent control signals by the proposed approach. Furthermore, the results show that the system has merit compared with prior systems: it allows cursor movement between arbitrary positions.**

***Index Terms*—Brain–computer interface (BCI), EEG, motor imagery, mu rhythm, P300 potential, two-dimensional cursor control.**

I. Introduction

As a communication and control pathway to directly translate brain activities into computer control signals, brain–computer interface (BCI) has attracted increasing attention in recent years from multiple scientific and engineering disciplines as well as from the public [1]–[5]. Offering augmented or repaired sensory-motor functions, it appeals primarily to people with severe motor disabilities [6]. Furthermore, it provides a useful test bed for the development of mathematical methods in brain-signal analysis [7].

An important issue in BCI research is cursor control, where the objective is to map brain signals to movements of a cursor on a computer screen. Potential applications include BCI-based neuroprostheses. In the EEG-based BCI field, most related studies were focused on 1-D cursor control, which was generally implemented through detecting and classifying the changes of mu (8–12 Hz) or beta (13–28 Hz) rhythm during different motor imagery tasks, e.g., imagination of left- and right-hand movement [8]–[12]. The physiological background is that the imagination of movement gives rise to short-lasting and circumscribed attenuation (or accentuation) in mu and beta rhythm activities in EEG, known as event-related desynchronization (or synchronization) (ERD/ERS) [13].

Compared with 1-D cursor control, multidimensional cursor control enables considerably enhanced interfacing between the user and the machine, implying a much wider range of applications. To date, most of the multidimensional cursor control BCIs have been invasive [14] or with expensive neuroimaging devices like magnetoencephalography [15]. On the other hand, the development of noninvasive EEG-based 2-D control BCI is impeded by the difficulty in obtaining two independent control signals from the noisy EEG data of poor spatial specificity. Therefore, the first report of an EEG-based 2-D cursor control BCI is remarkable [16]: the authors showed that through guided user training of regulating two particular EEG rhythms (mu and beta), two independent control signals could be derived from combinations of the rhythmic powers. However, the downside of this approach is the required intensive user training.

In recent years, other forms of 2-D BCI were also reported that adopted a discrete control paradigm using the steady-state visual evoked potential (SSVEP) [17]–[19]. Specifically, a few visual stimuli sources were placed around the display, and EEG was classified to determine if the SSVEP signal was registered by a visual stimulus. A positive detection would mean that the user was looking at that particular stimulus's position; thus, the system would move the cursor toward it. Faller *et al.* proposed a configurable application framework seamlessly integrate SSVEP stimuli within a desktop-based virtual environment [20]. An avatar in virtual environment was successfully driven to perform several complex tasks such as navigation in slalom scenario and apartment scenario. SSVEPs offer excellent information transfer rates within BCI systems while requiring only minimal training [20]. One common characteristic for SSVEP-based 2-D control is that for each update of the position, the cursor takes only one of a few fixed directions, e.g., turn 45°

Manuscript received January 26, 2010; revised May 4, 2010 and June 12, 2010; accepted June 12, 2010. Date of publication July 8, 2010; date of current version September 15, 2010. The work of Y. Li was supported by the National Natural Science Foundation of China under Grant 60825306 and by the Natural Science Foundation, Guangdong, China under Grant 9251064101000012. The work of Z. Yu was supported by the National Natural Science Foundation of China under Grant 60802068 and by the Fundamental Research Funds for the Central Universities, South China University of Technology under Grant 2009ZZ0055. *Asterisk indicates corresponding author*.

*Y. Li is with the School of Automation Science and Engineering, South China University of Technology, Guangzhou, 510640, China (e-mail: auyqli@scut.edu.cn).

J. Long, T. Yu, and Z. Yu are with the School of Automation Science and Engineering, South China University of Technology, Guangzhou, 510640, China.

C. Wang, H. Zhang, and C. Guan are with the Institute for Infocomm Research, Agency for Science, Technology and Research (A*STAR), Singapore 138632.

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org. This includes one multimedia RMVB format movie clip, which shows a natural scene. The size of the video is 14 MB.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TBME.2010.2055564

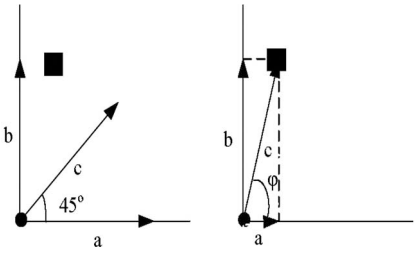


Fig. 1. Illustration example for comparison between discrete direction control, e.g., by SSVEP-based BCIs and arbitrary direction control, where the angle φ is time-varying. See the Section I for explanation.

left, turn 45° right, or walk straight ahead in [20], and the speed of the cursor is a constant. This, however, leads to unsmooth, zigzag like moves of the cursor. Similarly, there also existed several studies in which the P300 potential was used for 2-D control [18], [24].

Real-word applications often require 2-D moves of a cursor, etc. between arbitrary positions. A typical example is the computer mouse in which the initial position of the mouse and the positions of the targets are generally random in the screen. To carry out the movement, the route taken by the cursor should be smooth rather than zigzag. This requires that the cursor can take an arbitrary direction and a flexible step size in each update of position. We would like to illustrate an example in Fig. 1. The left subplot shows a case of discrete control by, e.g., a SSVEP-based system. The cursor (circle) is allowed to move in one of the three given directions "a," "b," and "c" at this time. Since the target (square) diverges from all the given directions, the user must set a movement direction and later alter it during the course of the movement. Furthermore, if the target is not located in a zigzag-like route starting from the origin of coordinates in this subplot, it will be difficult for the user to move the cursor to hit the target. In contrast, the cursor in the right subplot can move in arbitrary direction, resulting in more efficient 2-D control. This could be implemented through the combination of two independent control signals, in which the length of a is adjusted by the amplitude of the horizontal control signal.

Therefore, we propose a new paradigm for 2-D cursor control by simultaneously detecting two brain signals: P300 and motor imagery. We devise two uncued detection mechanisms for the two brain signals, respectively, and design a special graphical user interface (GUI) for 2-D cursor control. With this interface, the user control the vertical movement by using a P300 paradigm. At the same time, the user also use a motor imagery paradigm to control the horizontal movement. The P300 paradigm allows to select one of the three states: moving upwards, moving downwards, or no vertical movement. The velocity of vertical movement is fixed or zero. The motor imagery paradigm translates motor imagery into a continuous value that determine the direction and the velocity of the horizontal movement. Through selecting the vertical movement state and manipulating the continuous horizontal speed with direction, users will move the cursor along any direction in a self-paced manner.

A real-time bimodal BCI system based on this method is implemented in which P300 is evoked by visual stimuli. Since the neural mechanism of motor imagery differs largely from that of P300, the two signals can be independently controlled by the user. The independence between two control signals plays an important role in 2-D cursor control as mentioned earlier. As will be seen in this paper, the user can move the cursor from random initial position to hit the target placed at another random position based on this temporary independence.

To evaluate the proposed method, we conduct an online experiment involving six subjects. After a short calibration session for subject-specific P300 modeling and another for subject-specific motor imagery modeling, the subjects undergo a few sessions of feedback training using the GUI, and subsequently perform cursor control tasks in test sessions. In particular, the feedback training emphasizes on learning of motor imagery control. Positive results are obtained and analyzed that attest to the efficacy of the proposed method.

It is interesting to note that the proposed system can be viewed as a hybrid BCI. Hybrid BCIs have become an active research topic in recent years [21]–[23]. Allison *et al.* [21], [22] demonstrated that by combining multiple brain signals like motor imagery and SSVEP, the BCI can improve accuracy especially for users with poor performance. While earlier studies were focused on cued (machine-paced) EEG-classification studies, this particular paper pays attention to 2-D control.

The remainder of this paper is organized as follows. The methodology including the GUI, the P300, and motor imagery detection algorithms is presented in Section II. The experimental results are presented in Section III. Further data analysis and discussions are in Section IV. Section V finally concludes the paper.

II. Methodology

A. System Paradigm

The system consists of two components: an EEG acquisition device and a computer system. The EEG device from Compumedics captures 32-channel electric potentials from the scalp of the user wearing an EEG cap LT 37 referenced to the right ear, digitized at a sampling rate of 250 Hz. The two components are connected via a USB port and a parallel port, the former for transmitting EEG data, while the latter for synchronization. In our system, two channels horizontal electrooculograph (HEOG) and vertical electrooculograph (VEOG) representing eye movements are excluded for signal processing. The remaining 30 channels are used without further channel selection, which are shown in Fig. 2.

The computer system receives the EEG data, detects P300 and motor imagery signals (see Section II-C), and translates the detection outputs into a cursor's movement in a special GUI (see Section II-B). The user sits comfortably in an armed chair in front of the computer display.

B. GUI and Control Mechanism

The GUI is illustrated in Fig. 3 in which the ball and the square, respectively, represent a cursor and a target. In each trial, the initial cursor and target randomly appear inside a rectangular

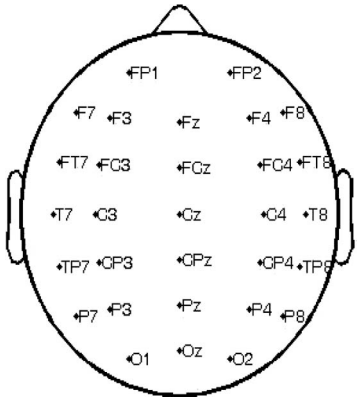


Fig. 2. Names and distribution of electrodes.

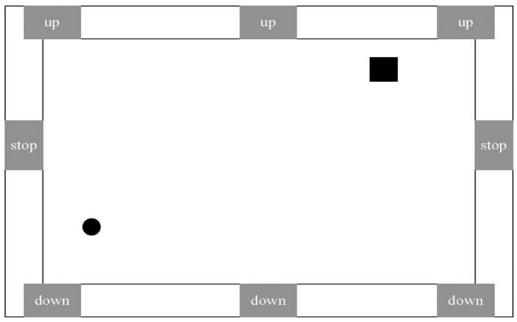


Fig. 3. GUI for 2-D cursor control in which a cursor (circle), a target (square), and eight flashing buttons ("up," "down," and "stop") are included.

workspace. This means that the positions of the initial cursor and the target as well as the distance between the two are random.

In our system, the GUI has a dimension of 1166 pixels $\times$721 pixels. The ratios of the size of the cursor, the size of the target, and the size of workspace are fixed to be 0.00084:0.003:1. The initial position of the cursor and the position of the target are randomly generated from a uniform distribution.

Eight flashing buttons are located at the horizontal and vertical edges of the screen, which would generate P300 potentials when the user focus attention on one of them. They serve functions for vertical movement control: we assign the three buttons at the top to the "up" function, the three buttons at the bottom to the "down" function, and the two buttons on both sides to the "stop" function.

Therefore, the user controls the vertical movement of the cursor by paying attention to one of the buttons, while ignoring the other buttons. Since a single P300 potential is difficult to detect due to low SNR, several repetitions of P300 are required before a detection decision is made (see Section II-C Part A for elaboration). In other words, the user needs to keep attention on a particular button while the system flashes the buttons for a few rounds, each round is a complete circle in which all the buttons are flashed once in a random order.

If the user wants to move the cursor up, then he needs to focus on one of the three "up" buttons. When the system detects his P300 corresponding to this button, the cursor is moved upward. To the contrary, if the user wants to move the cursor down, then he needs to focus on one of the three "down" buttons such that his P300 can be detected corresponding to this "down" button and the cursor is moved downward. Finally, if he does not want to move the cursor in vertical direction, then he can focus on one of the two "stop" buttons.

The main objective for the arrangement of three "up" buttons and three "down" buttons is for user's convenient use. For example, if the cursor is now in the right-hand side of the GUI, then the user can choose the "up" or "down" button located in the same side to control the vertical movement of the cursor. Another consideration of this arrangement is to improve the "oddball" effect using eight buttons instead of three buttons (1 "up" button, 1 "down" button, and 1 "stop" button).

The horizontal movement of the cursor is controlled by motor imagery, similar to that in [11]. When the system detects a right-/left-hand imagery, the cursor moves toward right (left) side. Therefore, if the user wants to move the cursor toward right side, he or she needs to imagine movement of the right hand, and *vice versa*.

In this GUI, we use P300 and motor imagery for controlling the vertical movement and the horizontal movement of the cursor, respectively. In this way, the direction of the horizontal movement of the cursor (right or left) is consistent with the user's motor imagery (right hand or left hand). This design ensures that the user conveniently use the biofeedback during the control. Thus, we do not take another choice in which P300 and motor imagery are used for controlling the horizontal movement and the vertical movement of the cursor, respectively.

A test trial for 2-D cursor control is described as follows. A trial begins when a target and a cursor appear. At 100 ms later, the eight buttons begin to flash alternately in a random order. Each button is intensified for 100 ms, while the time interval between two consecutive button flashes is 120 ms. From the beginning of a trial, the subject attempts to move the cursor to hit the target. The trial ends when the cursor hits the target or when the 60-s timeout limit occurs. The interval between two consecutive trials is 2 s.

The system simultaneously performs detection of P300 and motor imagery, and uses the detection outputs to control the cursor's movement in each of the two dimensions in the 2-D workspace. The cursor's position is updated every 200 ms. It is noteworthy that the user is able to control the cursor's movement at any time at will, without having to keep pace with machine-generated cues like in synchronous systems (here, "cue" means a symbol, e.g., "+" appearing in the GUI that represents the beginning of a new detection of P300 or motor imagery). These mechanisms similar to asynchronous ones are elaborated in Section II-C. Briefly speaking, each detection mechanism of P300 and motor imagery makes a decision from a sliding window of EEG ending at the current time point.

C. Models and Algorithms for 2-D Cursor Control

1) Control of Vertical Movement Based on P300 Potential: The vertical movement of the cursor is determined by the output of P300 potential detection. Let $c(k) \in \{1, -1, 0\}$ represents the output: 1 for "down," -1 for "up," and 0 for "stop" (applied to vertical movement only). The vertical movement model is

given by

$$y(k+1) = y(k) + c(k)v_0 \tag{1}$$

where $y(k)$ represents the vertical position at the kth update, at a fixed interval of 200 ms and v_0 is a positive speed constant. We set $v_0 = 10$ pixels initially, which is subject to adjustment for maximization of each subject's performance. Therefore, if the output $c(k)$ is 0, the cursor stops vertical movement; if the output is 1 or -1, the cursor moves up or down at a speed of 10 pixels every 200 ms.

The output $c(k)$ is generated by the P300 detection algorithm described in the following.

Algorithm 1:

Step 1. Feature extraction based on the following Steps 1.1–1.3.

Step 1.1. All 30 channel EEG signals are filtered in the range of 0.1–20 Hz.

Step 1.2. Extract a segment of the EEG data of each channel for every flash. Considering that P300 generally occurs 300 ms post stimulus, we use the time window from 0 to 600 ms after a button flashes. Furthermore, downsample the segment of EEG signals by taking the first time point from each piece of 6 consecutive ones. The obtained data vector is denoted as $Pa_{i,j,q}$, where $i, j,$ and q represent the ith channel, the jth button associated with this flash, and the qth round, respectively. $Pa_{i,j,q}$ will contain a P300 waveform for part of channels, if the user is paying attention to this button. Note that the counting of the round number is restarted, if an output of the direction of the cursor's vertical movement is produced.

Step 1.3. Construct a feature vector corresponding to the jth button and the qth round by concatenating the vectors $Pa_{i,j,q}$ from all channels, i.e., $Fe_{j,q} = [Pa_{1,j,q}, \ldots, Pa_{30,j,q}]$.

Step 2. Train a support vector machine (SVM) classifier. During the training data-acquisition phase, the subject is asked to focus attention on six of the eight buttons one by one (three "up" buttons and three "down" buttons). In order to shorten the acquisition time of training data, the subject is not instructed to pay attention to the two "stop" buttons. The period of attention for each button lasts 64 consecutive rounds, where each round contains eight flashes from the eight buttons, respectively. Using the feature vectors of the training dataset $\{Fe_{j,q}, j = 1, \ldots, 8; q = 1, \ldots, 64\}$ and their corresponding labels, we train an SVM classifier. If the subject is focusing on the jth button in the qth round, then the label of $Fe_{j,q}$ is 1, while the labels of $Fe_{u,q}$ $(u \neq j)$ is -1.

Step 3. P300 detection with SVM scores based on the following Steps 3.1 and 3.2.

Step 3.1. For the lth round (counted from the previous output of a direction of the cursor's vertical movement), we extract feature vectors $Fe_{j,l}$ $(j = 1, \ldots, 8)$. Applying the trained SVM to these feature vectors, we obtain eight scores denoted as $s_{j,l}$ (the values of the objective function of the SVM).

Step 3.2. P300 detection requires discriminating between non-P300 rounds and P300 rounds, the former means that the user is in the so-called idle state or does not focus on a button. This may occur when the user does not intend to alter the vertical movement. The latter means that the user is paying attention to a button. For instance, when the user tries to alter the vertical movement, he needs to focus on a corresponding button. The detection of P300 is accomplished in this system by a threshold mechanism as below.

Calculate the sum of SVM scores for each button obtained from l accumulated rounds, then find the maximum and the second maximum of the eight summed scores. That is,

$$ss_j = s_{j,1} + \cdots + s_{j,l},\ j = 1, \ldots, 8 \tag{2}$$

$$ss_{j_0} = \max\{ss_1, \ldots, ss_8\}$$

$$ss_{j_1} = \max\{\{ss_1, \ldots, ss_8\} \backslash \{ss_{j_0}\}\} \tag{3}$$

where the j_1th button is obviously different from the j_0th button.

We now define a threshold condition by comparing the aforementioned maximum and the second maximum

$$1 - \frac{ss_{j_1}}{ss_{j_0}} > \theta_0 \tag{4}$$

where the threshold θ_0 is a predefined positive constant, and is empirically set at 0.3 that works favorably in our online experiment to be introduced later.

If the aforementioned threshold condition is satisfied, then the system makes a decision that P300 potential occurs at the j_0th button. Otherwise, the algorithm continues P300 detection of next round.

In our system, we set $5 \leq l \leq 15$. The first inequality implies that the detection of P300 for each round is based on at least five rounds of flashes (this round and the 4 previous rounds). This lower bound of l ensures sufficient time for classifying a P300 to some degree. The upper bound

of l implies that, if the aforementioned threshold condition is not satisfied in 15 rounds, the algorithm will make a decision that P300 occurs at the j_0th button with the maximum score ss_{j_0}.

Step 4. If P300 is detected for the lth round, which occurs at the j_0 button, then the system outputs a direction of the cursor's vertical movement corresponding to this button. Specifically, if the j_0 button is one of the three "up" buttons, set $c(k) = -1$ (the cursor will go up); if the j_0 button is one of the three "down" buttons, set $c(k) = 1$ (the cursor will go down); if the j_0 button is one of the two "stop" buttons, set $c(k) = 0$ (the cursor will have no vertical movement). See the GUI in Fig. 3.

If P300 is not detected for the lth round, i.e., the threshold condition (4) is not satisfied, then the algorithm does not reach a decision and the system does not change the direction of the vertical movement of the cursor, i.e., $c(k) = c(k-1)$.

Remark 1:

1) In Step 3.2 of Algorithm 1, if we only use the first maximum of SVM scores to determine the button eliciting a P300, then the processing of P300 detection contained in Steps 1, 2, and 3 is similar as in a standard P300-based BCI, e.g., a P300-based BCI speller [25].
2) In (4), we set a threshold for P300 detection by comparing the top two maximum scores. This is to emphasize the difference between the P300 and the background. A large difference indicates a higher probability of P300 occurrence. Extended visual attention to a flashing stimulus will increase the difference measure, making a positive detection more likely to happen.
3) If the threshold in (4) is set to zero, then the P300 detection only depends on the first maximum (Note: $1 - (ss_{j_1}/ss_{j_0})$ is always bigger than zero). In this case, Algorithm 1 reaches a decision in each round of flashes and outputs a direction of the vertical movement of the cursor. Furthermore, the user needs to continuously focus on his/her desired button. Otherwise, Algorithm 1 may output a wrong direction. This is, however, quite inconvenient for the user. On the other hand, the threshold in (4) cannot be too large. Otherwise, P300 can hardly be detected, making vertical speed control difficult for the user.
4) Two cases may lead to the violation of threshold condition (4) when the round number of flashes reaches the upper bound 15. The first is that no P300 is elicited, and the second is that two P300 potentials are elicited by two neighboring buttons (e.g., two neighboring "up" buttons), respectively. In these two cases, the difference between the first maximum and the second maximum of SVM scores defined in (3) may not be big enough for satisfying threshold condition (4). We make the assumption that the second case is not probable to happen in our system. This is because compared to a typical P300-based BCI, e.g., in [26], the buttons in our GUI are much further apart and the upper bound 15 of round number of flashes in our GUI is sufficiently large. Since the user focuses on only one button, it is unusual for its neighboring button(s) to elicit a P300. Thus, more rounds of flashes lead to bigger difference in scores between the desired button and its neighboring button(s).

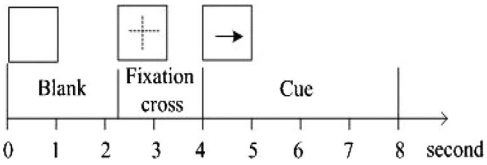


Fig. 4. Paradigm for training data acquisition of motor imagery (one trial). In the initial state (0–2.25 s), the screen remains blank. From 2.25 to 4 s, a cross appears in the screen to attract subject's visual fixation. From 4 to 8 s, a left-/right-arrow cue is shown and subject is instructed to imagine their left-/right-hand movement according to the cue.

In Algorithm 1, linear SVM is used as a classifier for detecting P300 as in [27]–[29]. The readers can also choose other classifiers such as Fisher's linear discriminant (FLD) [30], [31] and stepwise linear discriminant analysis (SWLDA) [32], [33]. In [31], the authors compared five methods commonly used for P300 classification, which were Person's correlation, FLD, SWLDA, linear SVM, and Gaussian kernel SVM. It follows from their statistical test results based on an experimental dataset that FLD, SWLDA, and linear SVM may provide better classification performance than the other two.

2) Control of the Horizontal Movement Through Motor Imagery: In our system, the horizontal movement of the cursor is controlled by the user's motor imagery using the following model:

$$x(k+1) = x(k) + \frac{a}{3}(f(k-2) + f(k-1) + f(k)) + b \quad (5)$$

where k represents the kth update of the cursor position, $x(k)$ is the horizontal coordinate of the cursor, $f(k)$ is the continuous output (score) of the SVM classifier, and a and b are two constants. $f(k)$, a and b will be described in details in the following. We introduce delays into the control model (5) to make the cursor move smoothly.

The score $f(k)$ is generated by an SVM classifier at every 200 ms. In specific, the system extracts the EEG block of 1200 ms ending at the current time point, and performs the following pre-processing steps before SVM: 1) spatial filtering with common average reference [8]; 2) bandpass filtering in specific mu rhythm band (8–13 Hz); and 3) spatial filtering based on a common spatial pattern (CSP) transformation matrix W.

The parameters in the first two preprocessing steps are fixed, while the CSP matrix W in the last step is learned in a subject-specific manner to enhance the separability between left/right motor imagery EEG data. In particular, the procedure of learning of W is as follows. First, a training dataset is collected for the user, where 60 trials of guided motor imagery tasks are performed (see Fig. 4 for an illustration of the paradigm of a trial). Second, the CSP matrix is constructed by the well-known joint diagonalization method [8].

The CSP-based spatial filtering essentially projects EEG data samples onto spatial components that carry discriminative spatial patterns for each motor imagery class. In this study, we select the top three components and the bottom three components from W, which best separate the two motor imagery classes. Furthermore, their logarithm variances are calculated and a 6-D feature vector is constructed.

The feature vector is fed into an SVM classifier that is trained using the training data to separate the two classes of feature vectors. Instead of using the hard output, we use the continuous output (score) of the SVM: $f(k)$.

In motor imagery control, the cursor shall not move, if the user is in idle state. This is accomplished here by the introduction of the two parameters: a and b [see (5)]. Specifically, we introduce the following procedure for the calibration of the two parameters for each subject.

An EEG dataset is first collected when the user is in idle state of motor imagery. This dataset contains N time segments of 200 ms ($N = 600$ in this paper), i.e., lasts 2 min. According to the aforementioned method, we calculate the SVM scores $f(1), \ldots, f(N)$.

Set

$$m = \frac{1}{N}\sum_{k=1}^{N} f(k)$$
$$mi = \min\{f(k),\ k = 1, \ldots, N\}$$
$$ma = \max\{f(k),\ k = 1, \ldots, N\}. \tag{6}$$

Then we calculate a and b as

$$a = \frac{h}{\max\{ma - m, m - mi\}} \qquad b = -am. \tag{7}$$

In the aforementioned calibration, the parameter h in (7) is used for adjusting the velocity of the cursor's horizontal movement. It may have different settings for different subjects. In our experiments, h is fixed to 8 for all the subjects.

Considering the model (5) and the aforementioned parameter setting of a and b, we can find that the average horizontal movement during the idle state $\frac{1}{N}\sum_{k=1}^{N-1}(x(k+1) - x(k))$ is close to zero. Thus, our calibration method has the advantage that the cursor almost does not change its horizontal position, if the subject is in the idle state of motor imagery. This has been demonstrated in our online experiments.

Remark 2: In the aforementioned algorithm, we choose CSP spatial filtering and linear SVM for feature extraction and classification, respectively. These two methods are commonly used in motor imagery based BCIs and proved to be effective [34]–[37]. CSP method can be explained under the framework of Rayleigh coefficient maximization like FLD [38]. Different classifiers could be employed in this system; however since that is not the focus of this study, the linear SVM was employed for classification.

Combining the aforementioned algorithms for vertical movement control and horizontal movement control, we obtain our algorithm for 2-D cursor control of which the diagram is shown in Fig. 5. It follows from Fig. 5 that for the kth movement of the cursor, the horizontal coordinate $x(k)$, and the vertical coordinate $y(k)$ are determined by motor imagery and P300 potential, respectively. Furthermore, the horizontal movement control based on motor imagery and the vertical movement control based on P300 are performed simultaneously in our algorithm as well as in our online BCI system. As will be shown in our data analysis, the two control signals are nearly independent.

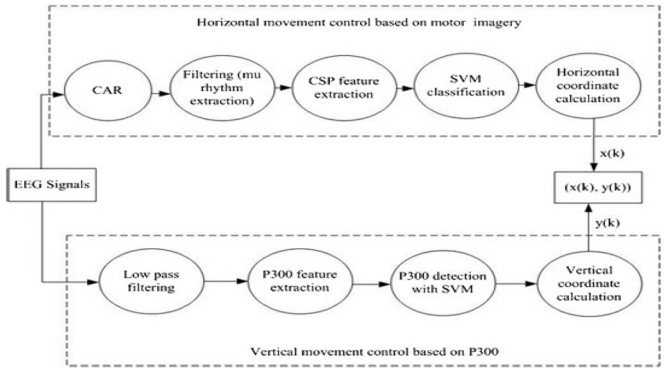


Fig. 5. Diagram of the algorithm for 2-D cursor control.

III. Online Experimental Results

Six subjects, five males and one female aged from 22 to 30, attended the online experiment. Two of them had limited prior experience in the 2-D-cursor control system during the system's development. The other four subjects were naïve users.

Before a subject controlled the cursor, three datasets were collected for subject-specific system calibration. The three sets were used in training P300 control model, training the motor imagery control model, and calibration. Prior parameters of the system were set as described in Section II. After system calibration, the user started to perform 2-D cursor-control tasks.

The following control strategy was explained to the subjects. To alter the vertical movement of the cursor, the user needs to continuously focus on a corresponding button until the cursor moves in his/her desired vertical direction. At the same time, the user may perform motor imagery to control the horizontal movement. This procedure can be seen at the beginning of all trials of supplementary experiment 2 in Appendix 1. When the vertical movement is set correctly, he may shift main attention to motor imagery; however, it is not advisable to shift eyesight away from the desired button to avoid false positives of P300 detection for vertical movement control. No other specific instruction about the route of movement was given to the subjects.

The training time for effectively using the system for the four naïve users ranged from two to eight independent sessions, of which each training session lasted about 2 h including preparation. Specifically, two of them were trained for two sessions, the other two were trained for five and eight sessions, respectively. All the training sessions for a subject were arranged in several consecutive weeks. Our criteria for terminating the training for a new user was that he/she can achieve the hit rate of about 80%. Note that each trial would automatically end, if the user could not move the cursor to hit the target in 60 s. According to our experience, the most difficult training task for a naïve user is

TABLE I
RESULTS IN EXPERIMENT 1

	Number of trials	Hit rate (%)	Average time (s)
Subject A	80	97.5	24.8
Subject B	80	91.25	25.0
Subject C	80	86.25	30.7
Subject D	80	84.5	28
Subject E	80	92.5	31.8
Subject F	80	92.5	25.6

TABLE II
CORRELATION COEFFICIENTS BETWEEN A CONTROL VARIABLE (MU OR P300) AND A POSITION VARIABLE ($\bar{x}$ OR $\bar{y}$), AND BETWEEN THE TWO CONTROL VARIABLES

Subject	$C_{mu,\bar{x}}$	$C_{mu,\bar{y}}$	$C_{P300,\bar{x}}$	$C_{P300,\bar{y}}$	$C_{P300,mu}$
A	0.685	0.016	0.038	0.653	0.042
B	0.521	0.008	0.044	0.566	0.004
C	0.432	0.048	0.013	0.407	0.028
D	0.443	0.019	0.041	0.506	0.021
E	0.501	0.001	0.024	0.321	0.001
F	0.411	0.035	0.008	0.417	0.012

in controlling the horizontal movement of the cursor by motor imageries. It is relatively easy to learn to combine the control based on P300 and the control based on motor imageries.

Table I shows the experimental results including the numbers of trials, accuracy rates for hitting the target, and the average time of a successful trial for the six subjects. Note that for each subject, all the trials were performed in a single session.

From Table I, we can find out that the accuracy rates of all the six subjects are satisfactory. However, the control time of each trial was not short (about 28 s in average). There are two main reasons.

1) The relative small size of the cursor and the target. For instance, the ratio of the target size and the workspace size is just 0.3%. It is more difficult for the user to move the cursor to hit the target for small sizes of cursor and target than for large sizes of cursor and target. Therefore, the subjects need a long time to control the cursor to hit the target.
2) Triggering and effectively detecting P300 are time consuming to some degree. Our future researches are to reduce the time for detecting P300 and improve the speed of our system.

IV. DATA ANALYSIS AND DISCUSSIONS

As described in the earlier section, we use P300 potential and mu rhythm as two control signals in our 2-D cursor-control system. For efficient 2-D control, the two control signals need to be as independent as possible. In the following, we first analyze the data collected from our online experiment and assess the independence between the two control signals by correlation method similarly as in [16].

In a trial of 2-D cursor control, there exist horizontal and vertical control variables with values $f(k)$, $c(k)$ (see algorithms in Section II), which are used for the kth update of the position of the cursor. Here, $f(k)$ is an SVM score from motor imagery, while $c(k)$ with value of 1, -1, or 0 is determined by the button ("down," "up," or "stop") at which P300 occurs. Furthermore, the button at which P300 occurs is determined by $ss_{j_0}(k)$, the maximum of 8 SVM scores corresponding to the eight buttons, respectively (see Algorithm 1 in Section II). Since $ss_{j_0}(k)$ is generally positive, we let $c(k)ss_{j_0}(k)$ to be a new vertical control variable with signs and amplitude.

For each trial, we then define two position variables representing the relative position of the cursor and the target

$$\bar{x}(k) = x_t - x_c,\ \bar{y}(k) = y_t - y_c \tag{8}$$

where x_t and x_c are the horizontal coordinates of the target and the cursor (at its initial position), respectively, and y_t and y_c are the vertical coordinates of the target and the cursor (at its initial position), respectively.

We calculate the correlation coefficients for each pair of control variable ($f(k)$ or $c(k)ss_{j_0}(k)$) and position variable ($\bar{x}(k)$ or $\bar{y}(k)$), and the correlation coefficient for the two control variables. These correlation coefficients shown in Table II are used to assess the independence between two control signals.

From Table II, we can see that for each subject, each control variable correlates strongly with its own dimension of target cursor position and does not correlate with the other variables' dimension and also does not correlate with the other variable. Hence, such results support the independence of the horizontal and vertical control variables.

Here, we would like to compare this system with the one in [16]. In [16], initial sessions were designed for all users. In these initial sessions, the transition from 1-D to 2-D control was accomplished by gradually increasing the magnitude of movement in the second dimension and/or by alternating between 1-D runs in the vertical and horizontal dimensions, and then switching to 2-D runs. In the present study, such special initial sessions are not necessary for training new users. Our experimental results indicate that a subject can use the system smoothly, as long as he or she is able to generate distinguishable P300 EEG and separable motor imagery EEG. In other words, if a user can perform P300-based BCI control and motor imagery-based BCI control separately, then in principle, he or she can use our 2-D cursor-control system through a simple training for combining mu/beta rhythm and P300 potential. Therefore, our system can be relatively convenient for naïve users.

Next, we present our data analysis results to show that the subjects truly used P300 and motor imagery in the 2-D cursor control.

We plot the topographies of average weights of SVM classifier used in P300 detection and the event-related potential (ERP) curves. These average weights are calculated as follows: from Algorithm 1, a feature vector associated with a button flash is constructed by concatenating 30 subvectors from all 30 channels, respectively. Through training, an SVM classifier, a weight vector is obtained for P300 detection. The components of this weight vector can also be partitioned into 30 subsets corresponding to the aforementioned 30 subvectors, respectively. The average weight corresponding to a channel is calculated by averaging the absolute values of all SVM weights of the subset related to this channel. For each of Subjects A and B, the

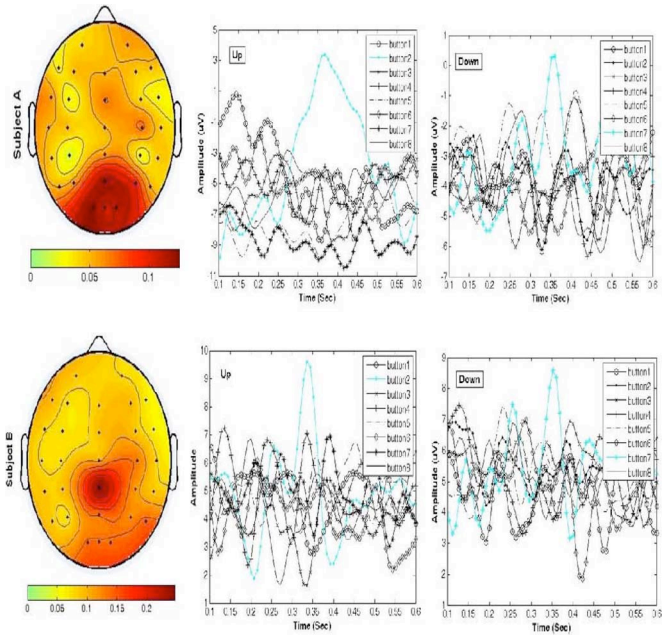


Fig. 6. Topographies of average weights of the SVM used for P300 detection and the ERP curves calculated from raw EEG signals for two subjects. Left: for each subject, average weights of the SVM are presented as scalp distribution. Right: for each subject, two green P300 ERP curves correspond to an "up" button (button 2) and a "down" button (button 7), respectively. No significant P300 waveform appears on the other buttons.

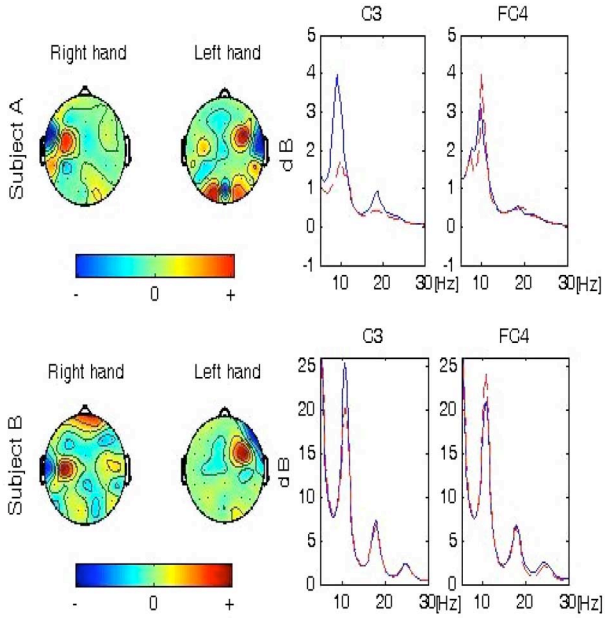


Fig. 7. Topographies of two selected CSP filters and the spectra of two channels of raw EEG signals for two subjects. Left: for each subject, two of the selected CSP filters (the first and the last rows of W) displayed as scalp maps. Right: for each subject, the spectra of two channels of raw EEG signals with blue curves referring to the motor imagery of right hand and red curves referring to the motor imagery of left hand.

average weights of SVM classifier used in P300 detection are displayed as a scalp map on the left of Fig. 6. It follows from these two scalp maps that our P300 detection is mainly based on those channels located in the occipital and parietal areas rather than the frontal area, an area considered to be related to EOG. Furthermore, for Subjects A and B, we choose an "up" button and a "down" button separately, which the users pay attention to during real-time control. For the "up"/"down" button, an ERP curve is obtained by averaging 15 epochs of raw EEG signal from channels "Pz" and "CPz" for subjects A and B, respectively (see Fig. 2). Each epoch is the time frame from 100 to 600 ms after onset of stimulus. Similar processing is performed for other buttons. The results are shown in the four subplots in the right of Fig. 6 from which we can see P300 potentials elicited by the two chosen buttons and no P300 potentials elicited by other buttons.

We also plot the topographies of CSP filters and the power spectra of two channels of raw EEG signals calculated based on the training dataset. For subjects A and B, two of the selected CSP filters (the first and the last rows of W) are displayed as scalp maps on the left of Fig. 7, which are easily related to the motor imageries of right and left hands, respectively. Plots on the right of Fig. 7 show the spectra calculated from two channels (C3 and FC4) of raw EEG signals. The discriminability of the brain signals corresponding to the motor imageries of right and left hands is demonstrated.

Two supplementary experiments are presented in Appendix 1. In the first experiment, we show the trajectories of the cursor starting at the center of the GUI and moving to one of the four fixed targets at the four corners, respectively. These trajectories are quite smooth. In the second experiment, we show that P300 control and motor imagery control can work simultaneously.

The user may shift primary attention between the two tasks: visual attention for P300, and motor imagery. Our empirical study suggests that it is not necessary to use the switched control mechanism: when P300 control is performed, motor imagery control is completely ignored, and *vice versa*. Please see supplementary experiment 2 in Appendix 1 for details.

V. Conclusion

In this paper, we have presented a new BCI and its implementation for 2-D cursor-control by combining the P300 potential and motor imagery. Two almost independent signals have been obtained for controlling two degrees of movements of a cursor simultaneously. In particular, a motor imagery detection mechanism and a P300 potential detection mechanism were devised and integrated with a specially designed GUI. A real-time implementation of the approach was assessed through an online experiment involving six subjects performing 2-D cursor control tasks. Using our system, the six subjects successfully carried out 2-D cursor control with satisfactory accuracies ($>80\%$). The results attest to the efficacy of obtaining two almost independent control signals by the proposed approach. Furthermore, it was demonstrated in our online experiment that: 1) a user can use our 2-D BCI system, if he/she can perform motor imagery-based BCI control and P300-based BCI control separately and 2) it allows cursor movement between arbitrary positions.

Hybrid BCIs should draw on existing principles of dual task integration and strive to minimize destructive interference

between the different tasks used to convey information [22]. In [22], it was shown that ERD activity was less apparent when an SSVEP task was added. In our system, the two brain activities, P300 and ERD, may also interfere with each other. To alleviate the problem, we set a threshold in (4) in the P300 detection algorithm (see Algorithm 1). With this threshold, the user may not be required to pay attention all the time primarily to the P300 stimulus associated with the desired vertical movement. This effectively eases the user's burden, and reduces the interference problem.

Future work could explore combination of SSVEP and motor imagery for 2-D cursor control. This combination was suggested in [21] and [22].

APPENDIX

TWO SUPPLEMENTARY EXPERIMENTS

In this appendix, we present two supplementary experiments. In the first experiment, we show the trajectories of the cursor starting at the center of the GUI and moving to one of the four fixed targets at four corners, respectively. While in the second experiment, we show that P300 control and motor imagery control work simultaneously.

Supplementary experiment 1: In the GUI in Fig. 3, the initial position of the cursor and the position of the target are random for each trial. Thus, the trajectories of the cursor are different for different trials. In order to show the average trajectory of the cursor based on desired movements, we perform a supplementary experiment. In this supplementary experiment, the target randomly appears at one of the four corners, while the starting position of the cursor is fixed at the center of the GUI for each trial.

Only Subject A attended this experiment, and 80 trials of data were collected. The hit rate and the average control time are 97.5% and 24.7 s (for successful trials), respectively. The trajectories of single trials and the average trajectories for four target positions are shown in Fig. 8, which are quite smooth.

Supplementary experiment 2: In this supplementary experiment, we fix two targets, one at the middle of upper part, and the other at the down-right corner, and fix the initial position of the cursor at the middle left of the GUI as shown in Fig. 9. In each trial, the subject needs to move the cursor to hit these two targets sequentially without stop. There are two classes of trials appearing in random order with the same probability. In the first class of trials, the subject move the cursor to first hit the target at the middle of upper part and then hit the target at down-right corner, while in the second class, the subject move the cursor to first hit the target at down right corner and then hit the target at the middle of upper part.

Subject A attended this experiment, and 80 trials of data were collected for two classes of trials (about 40 trials for each class). To implement this task, the subject first moved the cursor to the right by imagining right-hand movement, and focused on the desired button simultaneously to elicit a P300. After the cursor hits the first target, the subject focused on another desired button to change the direction of the vertical movement.

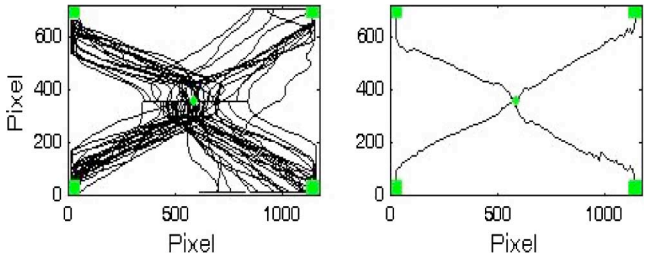


Fig. 8. Results in the supplementary experiment 1. Left: trajectories of the cursor obtained in 80 trials. Right: four average trajectories of the cursor for the four target positions, respectively.

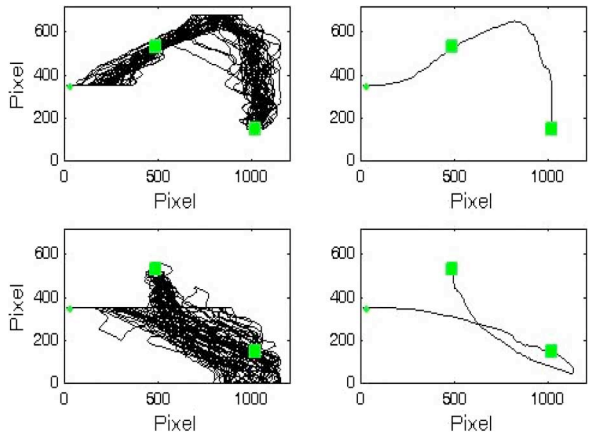


Fig. 9. Results in the supplementary experiment 2. First row: all trajectories of the cursor (left) and the average trajectory of the cursor (right) for the first class of trials. Second row: all trajectories of the cursor (left) and the average trajectory of the cursor (right) for the second class of trials.

Simultaneously, using motor imagery, he moved the cursor to the right for the first class or to the left for the second class of trials.

The hit rate and the average control time are 98.75% and 49.1 s (for successful trials), respectively. The trajectories of single trials and the average trajectories for two classes of trials are shown in Fig. 9. From Fig. 9, we can see that for each trial, the direction of vertical movement of the cursor is changed while the cursor simultaneously moves to the right. Specifically, we see only horizontal movement of the cursor at the beginning of each trial, which implies that there is only motor imagery control. After a while, there appears vertical movement for the cursor. This means that P300 control starts to work. After the cursor hit the first target, the cursor changes its direction of vertical movement to hit the second target. In each trial of cursor movement, the direction change in the vertical movement along with the horizontal movement toward the target implies that the cursor can be simultaneously controlled by the two control signals of P300 and motor imagery.

ACKNOWLEDGMENT

The authors would like to thank anonymous reviewers and P. Lucas, the Associate Editor for the insightful and constructive suggestions.

REFERENCES

[1] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain-computer interfaces for communication and control," *Clin. Neurophysiol.*, vol. 113, pp. 767–791, 2002.

[2] J. Mellinger, G. Schalk, C. Braun, H. Preissl, W. Rosenstiel, N. Birbaumer, and A. Kblera, "An MEG-based brain-computer interface (BCI)," *NeuroImage*, vol. 36, pp. 581–593, 2007.

[3] G. Pfurtscheller, C. Neuper, C. Guger, W. Harkam, H. Ramoser, A. Schlogl, B. Obermaier, and M. Pregenzer, "Current trends in Graz brain-computer interface research," *IEEE Trans. Rehabil. Eng.*, vol. 8, no. 2, pp. 216–218, Jun. 2000.

[4] N. Weiskopf, F. Scharnowski, R. Veit, R. Goebel, N. Birbaumer, and K. Mathiak, "Self-regulation of local brain activity using real-time functional magnetic resonance imaging (fMRI)," *J. Physiol Paris*, vol. 98, pp. 357–373, 2004.

[5] H. Ramoser, J. Müller-Gerking, and G. Pfurtscheller, "Optimal spatial filtering of single trial EEG during imagined hand movement," *IEEE Trans. Rehabil. Eng.*, vol. 8, no. 4, pp. 441–446, Dec. 2000.

[6] N. Birbaumer, N. Ghanayim, T. Hinterberger, I. Iversen, B. Kotchoubey, A. kubler, J. Perelmouter, E. Taub, and H. Flor, "A spelling device for the paralysed," *Nature*, vol. 398, pp. 297–298, 1999.

[7] J. P. Donoghue, "Connecting cortex to machines: Recent advances in brain interfaces," *Nat. Neurosci. Supplement*, vol. 5, pp. 1085–1088, 2002.

[8] G. Blanchard and B. Blankertz, "BCI competition 2003-a data set IIa: Spatial patterns of self-controlled brain rhythm modulations," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 1062–1066, Jun. 2004.

[9] M. Cheng, W. Y. Jia, X. R. Gao, S. K. Gao, and F. S. Yang, "Mu-rhythm-based cursor control: An offline analysis," *Clin. Neurophysiol.*, vol. 115, pp. 745–751, 2004.

[10] G. E. Fabiani, D. J. McFarland, J. R. Wolpaw, and G. Pfurtscheller, "Conversion of EEG activity into cursor movement by a brain-computer interface (BCI)," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 12, no. 3, pp. 331–338, Sep. 2004.

[11] D. J. McFarland and J. R. Wolpaw, "Sensorimotor rhythm-based brain-computer interface (BCI): feature selection by regression improves performance," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 13, no. 3, pp. 372–379, Sep. 2005.

[12] D. J. McFarland, W. A. Sarnacki, T. M. Vaughan, and J. R. Wolpaw, "Brain-computer interface (BCI) operation: Signal and noise during early training sessions Clinical," *Neurophysiology*, vol. 116, pp. 56–62, 2005.

[13] G. Pfurtscheller, C. Neuper, D. Flotzinger, and M. Pregenzer, "EEG-based discrimination between imagination of right and left hand movement," *Electroencephalogr. Clin. Neurophysiol.*, vol. 103, pp. 642–651, 1997.

[14] L. R. Hochberg, M. D. Serruya, G. M. Friehs, J. A. Mukand, M. Saleh, A. H. Caplan, A. Branner, D. Chen, R. D. Penn, and J. P. Donoghue, "Neuronal ensemble control of prosthetic devices by a human with tetraplegia," *Nature*, vol. 442, no. 13, pp. 164–171, 2006.

[15] M. V. Gervena and O. Jensenb, "Attention modulations of posterior alpha as a control signal for two-dimensional brain-computer interfaces," *J. Neurosci. Methods*, vol. 179, pp. 78–84, 2009.

[16] J. R. Wolpaw and D. J. McFarland, "Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans," *Proc. Natl. Acad. Sci. USA*, vol. 101, no. 51, pp. 17849–17854, 2004.

[17] L. J. Trejo, R. Rosipal, and B. Matthew, "Brain-computer interfaces for 1-D and 2-D cursor control: Designs using volitional control of the EEG spectrum or steady-state visual evoked potentials," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 14, no. 2, pp. 225–229, Jun. 2006.

[18] F. Beverina, G. Palmas, S. Silvoni, F. Piccione, and S. Giove, "User adaptive BCIs: SSVEP and P300 based interfaces," *PsychNol. J.*, vol. 1, no. 4, pp. 331–354, 2003.

[19] P. Martinez, H. Bakardjin, and A. Cichocki, "Fully online multicommand brain-computer interface with visual neurofeedback using SSVEP paradigm," *Comput. Intell. Neurosci.*, vol. 2007, pp. 94561-1–94561-9, 2007.

[20] J. Faller, G. Muller-Putz, D. Schmalstieg, and G. Pfurtscheller, "An application framework for controlling an avatar in a desktop-based virtual environment via a software SSVEP brain-computer interface," *Presence*, vol. 19, no. 1, pp. 25–34, 2010.

[21] B. Z. Allison, C. Brunner, V. Kaiser, G. R. Muller-Putz, C. Neuper, and G. Pfurtscheller, "Toward a hybrid brain-computer interface based on imagined movement and visual attention," *J. Neural Eng.*, vol. 7, no. 2, pp. 1–9, 2010.

[22] C. Brunner, B. Z. Allisona, D. J. Krusienskib, V. Kaisera, G. R. Muller-Putza, G. Pfurtschellera, and C. Neupera, "Improved signal processing approaches in an online simulation of a hybrid brain-computer interface," *J. Neurosci. Methods*, vol. 188, no. 1, pp. 165–173, 2010.

[23] Y. Li, C. Wang, H. Zhang, and C. Guan, "An EEG-based BCI system for 2D cursor control," in *Proc. Int. Joint Conf. Neural Netw.*, Hong Kong, 2008, pp. 2214–2219.

[24] F. Piccione, F. Giorgi, P. Tonin, K. Priftis, S. Giove, S. Silvoni, G. Palmas, and F. Beverina, "P300-based brain computer interface: Reliability and performance in healthy and paralysed participants," *Clin. Neurophysiol.*, vol. 117, pp. 531–537, 2006.

[25] M. Thulasidas, C. Guan, and J. Wu, "Robust classification of EEG signal for brain-computer interface," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 14, no. 1, pp. 24–29, Mar. 2006.

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

[27] M. Kaper, P. Meinicke, U. Grossekathoefer, T. Lingner, and H. Ritter, "BCI competition 2003-data set IIb: Support vector machines for the P300 speller paradigm," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 1073–1076, Jun. 2004.

[28] A. Rakotomamonjy and V. Guigue, "BCI competition III: Dataset II-ensemble of SVMs for BCI P300 speller," *IEEE Trans. Biomed. Eng.*, vol. 55, no. 3, pp. 1147–1154, Mar. 2008.

[29] P. Meinicke, M. Kaper, F. Hoppe, M. Heumann, and H. Ritter, "Improving transfer rates in brain computer interfacing: A case study," *Adv. Neural Inf. Process. Syst.*, vol. 15, pp. 1131–1138, 2003.

[30] U. Hoffmann, J. Vesin, T. Ebrahimi, and K. Diserens, "An efficient P300-based brain-computer interface for disabled subjects," *J. Neurosc. Methods*, vol. 167, pp. 115–125, 2008.

[31] D. Krusienski, E. Sellers, F. Cabestaing, S. Bayoudh, D. McFarland, T. Vaughan, and J. R. Wolpaw, "A comparison of classification techniques for the P300 speller," *J. Neural Eng.*, vol. 3, pp. 299–305, 2006.

[32] E. Donchin, K. M. Spencer, and R. Wijesinghe, "The mental prosthesis: Assessing the speed of a P300-based brain-computer interface," *IEEE Trans. Rehabil. Eng.*, vol. 8, no. 2, pp. 174–179, Jun. 2000.

[33] L. A. Farwell and E. Donchin, "Talking off the top of your head: Toward a mental prosthesis utilizing event-related brain potentials Electroenceph," *Clin. Neurophysiol.*, vol. 70, pp. 510–523, 1988.

[34] B. Blankertz, G. Curio, and K. R. Müller, "Classifying single trial EEG: Towards brain computer interfacing," *Adv. Neural Inf. Process. Syst.*, vol. 14, pp. 157–164, 2002.

[35] D. Garrett, D. A. Peterson, C. W. Anderson, and M. H. Thaut, "Comparison of linear, nonlinear, and feature selection methods for EEG signal classification," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 11, no. 2, pp. 141–144, Jun. 2003.

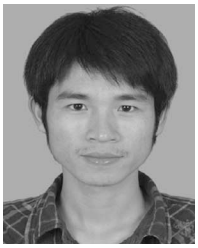
[36] K. Thomas, C. Guan, C. Lau, A. Vinod, and K. Ang, "A new discriminative common spatial pattern method for motor imagery brain–Computer interfaces," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 11, pp. 2730–2733, Nov. 2009.

[37] H. Ramoser, J. Müller-Gerking, and G. Pfurtscheller, "Optimal spatial filtering of single trial EEG during imagined handmovement," *IEEE Trans. Rehabil. Eng.*, vol. 8, no. 4, pp. 441–446, Dec. 2000.

[38] Y. Li and C. Guan, "Joint feature re-extraction and classification using an iterative semi-supervised support vector machine algorithm," *Mach. Learning*, vol. 71, pp. 33–45, 2008.

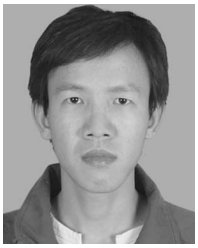
Yuanqing Li was born in Hunan, China, in 1966. He received the B.S. degree in applied mathematics from Wuhan University, Wuhan, China, in 1988, the M.S. degree in applied mathematics from South China Normal University, Guangzhou, China, in 1994, and the Ph.D. degree in control theory and applications from the South China University of Technology, Guangzhou, in 1997.

Since 1997, he has been with the South China University of Technology, Guangzhou, China, where he became a Full Professor in 2004. During 2002–2004, he was a Researcher at the Laboratory for Advanced Brain Signal Processing, RIKEN Brain Science Institute, Saitama, Japan. He was a Research Scientist at the Laboratory for Neural Signal Processing, Institute for Infocomm Research, Singapore during 2004–2008. He is the author or coauthor of more than 60 scientific papers in journals and conference proceedings. His research interests include blind signal processing, sparse representation, machine learning, brain–computer interface, EEG, and fMRI data analysis.



Jinyi Long received the B.S. degree in materials processing and controlling engineering from Northeastern University, Shenyang, China, in 2006. He is currently working toward the Ph.D. degree in pattern recognition and intelligent systems at the South China University of Technology, Guangzhou, China.

His research interests include machine learning, brain-signal analysis, and brain–computer interfaces.



Tianyou Yu received the B.Sc. degree in automatic control from the Wuhan University of Science and Technology, Wuhan, China, in 2008. He is currently working toward the Ph.D. Degree in pattern recognition and intelligent systems at the South China University of Technology, Guangzhou, China.

His research interests include noninvasive brain–computer interfaces and brain-signal analysis.

Zhuliang Yu received the B.S.E.E. and M.S.E.E. degrees in electronic engineering from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 1995 and 1998, respectively, and the Ph.D. degree from Nanyang Technological University, Singapore, in 2006.

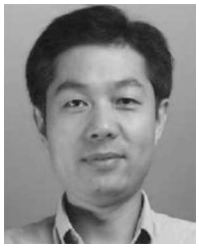
From 1998 to 2000, he was a Software Engineer at Shanghai BELL Company Ltd. He joined the Center for Signal Processing, Nanyang Technological University, in 2000 as a Research Engineer, and then became a Research Fellow. Since 2008, he has been an Associate Professor at the College of Automation Science and Engineering, South China University of Technology, Guangzhou, China. His research interests include array signal processing, acoustic signal processing, adaptive signal processing and their applications in communications, and biomedical engineering, etc.

Chuanchu Wang received the B.Eng. and M.Eng. degrees in communication and electrical engineering from the University of Science and Technology of China, Hefei, Anhui, China, in 1988 and 1991, respectively.

He is currently a Research Manager at the Institute for Infocomm Research, Agency for Science, Technology and Research (A*STAR), Singapore. His research interests include EEG-based brain–computer interface and related applications.

Haihong Zhang received the Ph.D. degree in computer science from the National University of Singapore, Singapore, in 2005.

He joined the Institute for Infocomm Research, Agency for Science, Technology and Research (A*STAR), Singapore as a Research Follow, where he is currently a Project Manager for multimodal neural decoding. His research interests include machine learning, pattern recognition, and brain-signal processing for high-performance brain–computer interfaces.



Cuntai Guan (SM'03) received the Ph.D. degree in electrical and electronic engineering from Southeast University, China, in 1993.

From 1993 to 1994, he was at the Southeast University, where he was involved in the research on speech vocoder, speech recognition, and text-to-speech. During 1995, he was a Visiting Scientist at the Centre de Recherche en Informatique de Nancy (CRIN)/Centre National de la Recherche Scientifique (CNRS)-Institut National de Recherche en Informatique et en Automatique (INRIA), France, where he was involved in the research on key word spotting. From 1996 to 1997, he was at the City University of Hong Kong, Hong Kong, where he was engaged in developing robust speech recognition under noisy environment. From 1997 to 1999, he was at the Kent Ridge Digital Laboratories, Singapore, where he was involved in the research on multilingual, large vocabulary, continuous speech recognition. He was a Research Manager and the R&D Director for five years in industries, focusing on the development of spoken dialogue technologies. In 2003, he established the Brain–Computer Interface Group at Institute for Infocomm Research, Agency for Science, Technology and Research (A*STAR), Singapore, where he is currently a Senior Scientist and a Program Manager. His research interests include brain–computer interface, neural signal processing, machine learning, pattern classification, and statistical signal processing, with applications to assistive device, rehabilitation, and health monitoring.