A maximum mutual information approach for constructing a 1D continuous control signal at a self-paced brain–computer interface

J. Neural Eng. **7** (2010) 056009 (11pp) doi:10.1088/1741-2560/7/5/056009

A maximum mutual information approach for constructing a 1D continuous control signal at a self-paced brain–computer interface

Haihong Zhang and Cuntai Guan

Institute for Infocomm Research, 1 Fusionopolis Way, Singapore 138632

E-mail: hhzhang@i2r.a-star.edu.sg and ctguan@i2r.a-star.edu.sg

Received 29 December 2009
Accepted for publication 6 August 2010
Published 14 September 2010
Online at stacks.iop.org/JNE/7/056009

Abstract

This paper addresses an important issue in a self-paced brain–computer interface (BCI): constructing subject-specific continuous control signal. To this end, we propose an alternative to the conventional regression/classification-based mechanism for building the transformation from EEG features into a univariate control signal. Based on information theory, the mechanism formulates the optimum transformation as maximizing the mutual information between the control signal and the mental state. We introduce a non-parametric mutual information estimate for general output distribution, and then develop a gradient-based algorithm to optimize the transformation using training data. We conduct an offline simulation study using motor imagery data from the BCI Competition IV Data Set I. The results show that the learning algorithm converged quickly, and the proposed method yielded significantly higher BCI performance than the conventional mechanism.

(Some figures in this article are in colour only in the electronic version)

1. Introduction

Recent advances in neural engineering have spurred a surge of interest in developing EEG-based brain–computer interface (BCI) technology, which caters to the demands from rehabilitation, assistive technology, and beyond [1, 2]. The operation of BCI can be broadly divided into synchronous (also known as cue-based or machine-paced) and self-paced (also known as asynchronous) categories (e.g. [2, 3]). The former (e.g. [4–7]) requires the user to execute mental activity in time windows determined by the machine, while the latter allows the user to perform BCI control at any time at will. There is a growing awareness of the importance of self-paced BCI systems [8–13], because of their more natural and potentially higher speed interfacing (e.g. [14]).

This paper focuses on a particular type of self-paced BCI, which transforms EEG into a continuous control signal (e.g. [15–17]). Providing the user with continuous control is essential for efficient and natural cursor control applications. It also serves two usages: first, it allows the user to sense and predict the system's action in a continuous and real-time manner so as to plan the mental states to activate desired BCI actions [15]; second, it enables feedback training in which the users, e.g. patients with ALS [17], can learn to regulate brain waves so as to gain better BCI operation.

The current methodology of transforming EEG features into a continuous control signal generally uses a regression approach. For instance, the works in [3, 16, 18] built a classifier to distinguish between two or multiple classes of EEG, while taking the continuous output of the classifier as the actual control signal. Thus, the classifier performed like a special regression machine. In the BCI Competition IV Data Set I [19], regression error in terms of mean-square error (MSE) was used as the metric for comparing un-cued motor imagery detection algorithms. At each time point, the error was measured between the output signal and the true mental state label: 0 for non-control (NC), and -1 and 1 for two motor imagery classes separately.

The regression approach is based on the MSE criterion and intuitive target values, while its link to actual BCI performance metrics is vague (see the related discussions in section 5). Therefore, it is interesting to see if an alternative method to regression can improve the BCI performance.

This paper proposes such a method, by using an information theoretic measure termed Shannon's mutual information [20] to design the transformation from EEG features to a univariate control signal. We consider the mutual information between the brain state (i.e. user's intention) and the control signal. Then the mutual information can be viewed as reduction of uncertainty about the mental state by observation of the control signal. Larger mutual information would mean less uncertainty or, in other words, better controllability of the signal by the mental state. Furthermore, the mutual information has been recommended in [21] for evaluating continuous BCI performance. It is then interesting to use the mutual information for directly optimizing the transformation (i.e. optimizing the control signal).

There are two technical challenges toward mutual information-based control signal design: (1) accurate estimation of the mutual information for general output distributions, and (2) learning of the optimum transformation such that the mutual information estimate is maximized. Note that the first point has been raised in [21], which pointed out that existing mutual information evaluation techniques were based on Gaussian assumption rather than on general output distributions.

The present work aims at solving these two problems. First, we introduce a non-parametric mutual information estimate, and formulate the optimum transformation for designing a univariate control signal. It thus enables us to consider more general data distributions than a simple Gaussian. Next, we develop a gradient-based optimization algorithm to learn from training data the optimum linear transformation. In addition, we assess the proposed method using human motor imagery data from the BCI Competition IV Data Set I. A cross-validation is performed, in which we study the convergence property of the proposed optimization algorithm and evaluate the performance using the receiver–operator-characteristics (ROC) analysis.

The rest of the paper is organized as follows. Section 2 describes the mutual information estimate for designing the optimum control signal. Section 3 develops a gradient-based learning solution. Section 4 describes the evaluation of the proposed approach, followed by discussions in section 5. Section 6 concludes this paper.

2. Maximum mutual information estimate for BCI control

We consider a 1D control BCI system using *motor imagery* [22], which refers to the imagination or mental rehearsal of a motor action without any real motor output. The primary phenomenon of motor imagery in EEG is event-related desynchronization (ERD) [22, 23]—the attenuation of the rhythmic activity over the sensorimotor cortex in the μ (8–14 Hz) and β (14–30 Hz) rhythms. The ERD can be induced by both imagined movements in healthy people or intended movements in paralyzed patients [17].

Effective control requires distinguishing and mapping the ERD and the NC signal (also called the idle state EEG) into a univariate signal that can differentiate various mental states. To this end, the spatio-spectral filtering approach [6, 24–26] has been quite successful for producing discriminative features. Using the spatio-spectral features, we design an un-cued BCI system as illustrated in figure 1.

The input to the system is a time sequence of multi-channel EEG $\hat{\mathbf{x}}(t)$, while the output is a univariate control signal $z(t)$. The procedure of transforming $\hat{\mathbf{x}}(t)$ into $z(t)$ comprises the following.

- *Feature extraction* yields a feature vector $\mathbf{a}$ to represent the EEG data at each time point (or practically at a fixed small interval of e.g. 0.5 s). The features contain discriminative spatio-spectral information among different mental states (i.e. motor imagery classes and NC). Below is a sequence of processing steps.
  (i) Processes raw EEG $\hat{\mathbf{x}}(t)$ with a band-pass filter h. The output signal is denoted by $\mathbf{x}(t)$. This step is meant for extracting subject-specific rhythmic activities in EEG. The responsive ERD rhythm needs to be identified, since it varies from one subject to another [24]. To this end, we select a band-pass filter h in conjunction with the spatial filters [26] (see section 4.1).
  (ii) Computes discriminative spatial components $\mathbf{y}$ by projecting $\mathbf{x}$ onto an array of vectors (representing individual spatial patterns)
  $$\mathbf{y}(t) = \mathbf{W}^T\mathbf{x}(t). \tag{1}$$
  The matrix $\mathbf{W}$ contains the spatial patterns in columns. This step addresses the subject-specific spatial patterns of the ERD. The matrix $\mathbf{W}$ can be constructed using the common spatial pattern algorithm [27].
  (iii) Computes the features in the form of log-power of the spatial components.
  $$\mathbf{a}(t) = \log\left[\frac{1}{L}\int_0^L [\mathbf{y}(t-\tau)]^2\,\mathrm{d}\tau\right], \tag{2}$$
  where L is the length of a short-time window ($L = 2$ s in this work) for computing the instantaneous power. The logarithm operation has been widely used since the introduction of CSP in [27], which described its purpose as 'to approximate normal distribution of the data'. We would like to note that another positive effect of the logarithm operation is the reduced dynamic range, which facilitates the subsequent processing. In addition, extreme feature values (suspected artifacts) in some EEG blocks can be largely reduced before the corrupted information (such as intra-class variance) is fed into the learning machine.

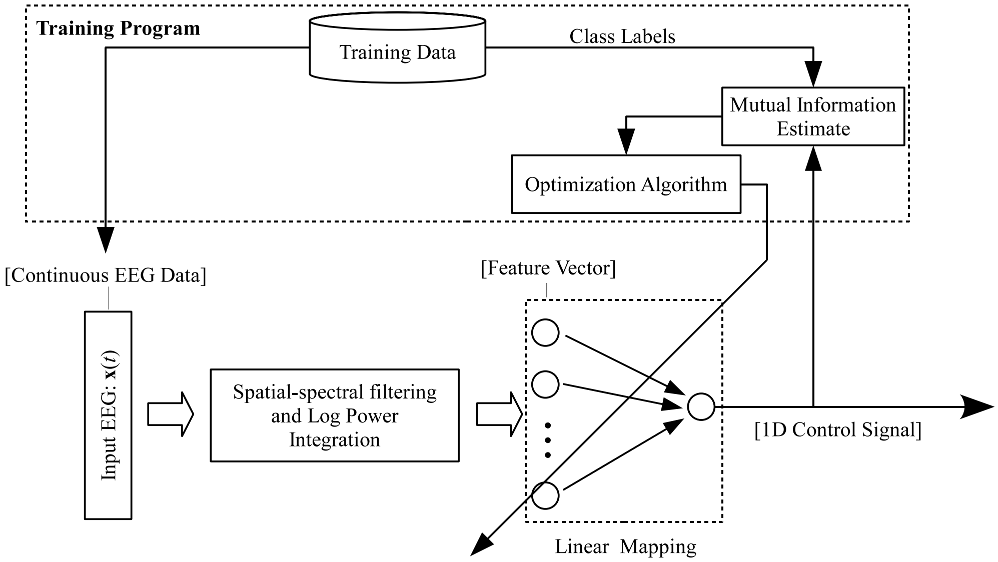

Figure 1. Diagram of learning and processing of motor imagery EEG for a self-paced continuous control BCI. Unlike in prior designs, the linear mapping has no preset target values for different mental states. Instead, it constructs a linear projection that maximizes the mutual information between the control signal and the mental state.

- Linearly transforms the log-power features into a univariate control signal z.

$$z(t) = \mathbf{r}^T \mathbf{a}(t), \tag{3}$$

where $\mathbf{r}$ is the projection vector, obtained as described below.

In the following we derive a formulation of the objective function for designing optimum linear transformation in equation (3), using the information theoretical measure of mutual information.

Mathematically, the mutual information between the control signal variable $\mathcal{Z}$ and the mental state variable $\mathcal{C}$ is given by [20]

$$I(\mathcal{Z},\mathcal{C}) = H(\mathcal{Z}) - H(\mathcal{Z}|\mathcal{C}) = H(\mathcal{Z}) - \sum_{c\in\mathcal{C}} H(\mathcal{Z}|c)P(c). \tag{4}$$

Here $H(\mathcal{Z})$ denotes the entropy of the continuous 1D control signal variable $\mathcal{Z}$ ($z \in \mathcal{Z}$ being a particular control signal value), $\mathcal{C}$ is the categorical brain state variable ($c \in \mathcal{C}$ being a particular mental state) and $H(\mathcal{Z}|c)$ is the conditional entropy of the control signal for the particular brain state c.

Shannon's entropy $H(\mathcal{Z})$ of the control signal and the conditional entropy $H(\mathcal{Z}|c)$ are respectively defined by

$$H(\mathcal{Z}) = -\int_{-\infty}^{\infty} p_z(z)\log(p_z(z))\,\mathrm{d}z, \tag{5}$$

and

$$H(\mathcal{Z}|c) = -\int_{-\infty}^{\infty} p_z(z|c)\log(p_z(z|c))\,\mathrm{d}z. \tag{6}$$

Theorem 1. *Under the following transformation of the control signal z,*

$$z' = g_1 z + g_0, \tag{7}$$

the mutual information is invariant if $g_1 \neq 0$.

Proof. The entropy of the transformed signal becomes

$$\begin{aligned} H(g_1\mathcal{Z} + g_0) &= -\int_{-\infty}^{\infty} p(g_1 z + g_0)\log(p(g_1 z + g_0))d(g_1 z + g_0) && (8)\\ &= -\int_{-\infty}^{\infty} g_1 \frac{1}{|g_1|} p_z(z)\log\left(\frac{p_z(z)}{|g_1|}\right)\mathrm{d}z && (9)\\ &= -\int_{-\infty}^{\infty} p_z(z)\log(p_z(z))\mathrm{d}z + \log(|g_1|) && (10)\\ &= H(\mathcal{Z}) + \log(|g_1|), && (11)\end{aligned}$$

where p_z is the probability density function of the variable $\mathcal{Z}$.

However, the change in the entropy is canceled out in the mutual information:

$$\begin{aligned} I(g_1\mathcal{Z} + g_0, \mathcal{C}) &= H(g_1\mathcal{Z} + g_0) - \sum_{c\in\mathcal{C}} P_c(c)H(g_1\mathcal{Z} + g_0|c) && (12)\\ &= H(\mathcal{Z}) + \log(|g_1|) - \sum_{c\in\mathcal{C}} P_c(c)[H(\mathcal{Z}) + \log(|g_1|)] && (13)\\ &= H(\mathcal{Z}) - \sum_{c\in\mathcal{C}} P_c(c)H(\mathcal{Z}) && (14)\\ &= I(\mathcal{Z},\mathcal{C}), && (15)\end{aligned}$$

where P_c is the probability function of the variable $\mathcal{C}$.

Hence, the mutual information remains the same as before the transformation. □

The invariant property above is desirable in designing algorithms for a continuous control BCI, in which linear transformation of the output signal shall have no effect on the performance of the BCI.

The mutual information is a function of output density distributions, and generally cannot be expressed in an explicit form. To address this problem, we introduce a mutual information estimation method below.

The mutual information (equation (5) or equation (6)) can be expressed as an expectation [28], which in turn can be approximated using empirical samples. Suppose there are n_z examples of the control signal: z_i, $i = 1, \ldots, n_z$. The approximation to $H(\mathcal{Z})$ takes the following form:

$$H(\mathcal{Z}) = -E[\log(p_z(z))] \cong -\frac{1}{n_z}\sum_{i=1}^{n_z}\log(p_z(z_i)). \tag{16}$$

The probability density function p_z in the above expression can be estimated using a Gaussian kernel density estimator

$$\hat{p}_z(z) = \frac{1}{n_a}\sum_{i=1}^{n_a}\varphi(z - z_i), \tag{17}$$

where

$$\varphi(z - z_i) = \alpha \exp\left(-\frac{\psi^{-1}}{2}(z - z_i)^2\right). \tag{18}$$

Here α is a factor that ensures that the integration of equation (17) equals 1 so as to meet the requirement for a probability density function. The bandwidth of the Gaussian kernel, ψ, is computed by

$$\psi = \zeta\frac{1}{n_z - 1}\sum_{i=1}^{n_z}(z_i - \bar{z})^2, \tag{19}$$

where $\bar{z}$ is the empirical mean of z, and the coefficient $\zeta = \left(\frac{4}{3n_z}\right)^{0.1}$ according to the normal optimal smoothing strategy [29].

By substituting equation (17) into equations (16), the entropy of feature-vector variable can be estimated using

$$\hat{H}(\mathcal{Z}) = -\frac{1}{n_a}\sum_{i=1}^{n_a}\log\left\{\frac{1}{n_a}\sum_{j=1}^{n_a}\varphi(z_i - z_j)\right\}, \tag{20}$$

and the conditional intra-class entropy can be estimated similarly by

$$\hat{H}(\mathcal{Z}|c) = -\frac{1}{n_c}\sum_{z_i\in c}\log\left\{\frac{1}{n_c}\sum_{z_j\in c}\varphi(z_i - z_j)\right\}. \tag{21}$$

The mutual information estimate is then

$$\hat{I}(\mathcal{Z}, \mathcal{C}) = \hat{H}(\mathcal{Z}) - \sum_c P(c)\hat{H}(\mathcal{Z}|c). \tag{22}$$

This mutual information estimate shares the invariant property with the original mutual information (see theorem 1).

Theorem 2. *The mutual information estimate in equation (22) is invariant against nontrivial transformation of the control signal z.*

Proof. Suppose the control signal is transformed by $z' = g_1 z + g_0$ (equation (7)), and the factor α (equation (18)) is fixed. The bandwidth ψ in equation (19) changes accordingly

$$\psi' = g_1^{-2}\psi. \tag{23}$$

The Gaussian kernel function in equations (20) and (21) will become

$$\begin{aligned}\hat{\varphi}(z'_i - z'_j) &= \alpha \exp\left(-\frac{\psi'^{-1}}{2}(z'_i - z'_j)^2\right)\\ &= \alpha \exp\left(-\frac{g_1^{-2}\psi^{-1}}{2}g_1^2(z - z_i)^2\right) \qquad (24)\\ &= \alpha \exp\left(-\frac{\psi^{-1}}{2}(z - z_i)^2\right) = \varphi(z - z_i). \qquad (25)\end{aligned}$$

Therefore, the transformation does not result in change in the Gaussian kernel function, thus has no effect on the entropy estimate in equations (17) and (21) as well as the mutual information estimate (equation (22)). □

It can be easily seen that, if the bandwidth ψ is fixed and not a function of z, the property would generally not hold. Therefore, we would like to emphasize the importance of the variable bandwidth ψ for z during optimization.

Now we formulate the learning of the optimum linear projection vector $\mathbf{r}_{\mathrm{opt}}$ so as to maximize the mutual information estimate:

$$\mathbf{r}_{\mathrm{opt}} = \underset{\mathbf{r}}{\operatorname{argmax}}\,\hat{I}(\mathcal{Z}, \mathcal{C}). \tag{26}$$

Hereafter we refer to the linear projection with $\mathbf{r}_{\mathrm{opt}}$ as *optimum linear mapping* or OLM.

3. Learning algorithm for optimum linear mapping

From equation (22), the gradient of mutual information estimate with respect to the linear projection vector $\mathbf{r}$ can be expressed as

$$\nabla_{\mathbf{r}}\hat{I}(\mathcal{Z}, \mathcal{C}) = \nabla_{\mathbf{r}}\hat{H}(\mathcal{Z}) - \sum_{c\in\mathcal{C}}P(c)\nabla_{\mathbf{r}}\hat{H}(\mathcal{Z}|c). \tag{27}$$

From equation (20), we have

$$\nabla_{\mathbf{r}}\hat{H}(\mathcal{Z}) = -\frac{1}{n_a}\sum_{i=1}^{n_a}\beta_i\frac{1}{n_a}\sum_{j=1}^{n_a}\frac{\partial\varphi(z_i - z_j)}{\partial\mathbf{r}}, \tag{28}$$

where

$$\beta_i = \left(\frac{1}{n_a}\sum_{j=1}^{n_a}\varphi(z_i - z_j)\right)^{-1}. \tag{29}$$

From equation (18), we have

$$\frac{\partial\varphi(z_i - z_j)}{\partial\mathbf{r}} = -\frac{1}{2}\varphi(z_i - z_j)\left[2\psi^{-1}(z_i - z_j)\frac{\partial(z_i - z_j)}{\partial\mathbf{r}} + (z_i - z_j)^2\frac{\partial\psi^{-1}}{\partial\mathbf{r}}\right]. \tag{30}$$

Since $z = \mathbf{r}^T\mathbf{a}$,

$$\frac{\partial(z_i - z_j)}{\partial\mathbf{r}} = (\mathbf{a}_i - \mathbf{a}_j). \tag{31}$$

To compute $\frac{\partial\psi^{-1}}{\partial\mathbf{r}}$, it follows from equation (19) that

$$\frac{\partial\psi^{-1}}{\partial\mathbf{r}} = \eta\sum_{i=1}^{n_z}\frac{\partial(r^T\mathbf{a}_i - r^T\bar{\mathbf{a}})^2}{\partial\mathbf{r}}, \tag{32}$$

where

$$\eta = -\zeta^{-1}(n_z - 1)\left[\sum_{i=1}^{n_z}(r^T\mathbf{a}_i - r^T\bar{\mathbf{a}})^2\right]^{-2}. \qquad (33)$$

We further develop equation (32):

$$\begin{aligned}\frac{\partial\psi^{-1}}{\partial\mathbf{r}} &= \eta\sum_{i=1}^{n_z}\frac{\partial(\mathbf{r}^T(\mathbf{a}_i - \bar{\mathbf{a}})(\mathbf{a}_i - \bar{\mathbf{a}})^T\mathbf{r})}{\partial\mathbf{r}}\\ &= \eta\frac{\partial\left(\mathbf{r}^T\left[\sum_{i=1}^{n_z}(\mathbf{a}_i - \bar{\mathbf{a}})(\mathbf{a}_i - \bar{\mathbf{a}})^T\right]\mathbf{r}\right)}{\partial\mathbf{r}}\\ &= 2\eta(n_z - 1)\mathbf{r}^T\Psi_a, \qquad (34)\end{aligned}$$

where Ψ_a is the empirical covariance matrix of the feature vector variable **a**.

With the above equations, we are able to explicitly compute the gradient $\nabla_{\mathbf{r}}\hat{H}(\mathcal{Z})$. It can be seen that the gradient of the class conditional entropy $\nabla_{\mathbf{r}}\hat{H}(\mathcal{Z}|c)$ can also be computed with cth class EEG data only. Finally, the gradient $\nabla_{\mathbf{r}}\hat{I}(\mathcal{Z}, \mathcal{C})$ can be obtained, and an iterative optimization procedure can be applied using the following update function:

$$\mathbf{r}_{n_{\text{iter}}+1} = \mathbf{r}_{n_{\text{iter}}} + \lambda\nabla_{\mathbf{r}}\hat{I}_{n_{\text{iter}}}(\mathcal{Z}, \mathcal{C}), \qquad (35)$$

where λ is the step size. We employ a backtracking search procedure to determine the step size [30].

The mutual information estimate may not be a convex function. So the above gradient-based optimization procedure may fall into a local optimum instead of the global one. In other words, the initial condition is important for the optimization procedure. We tentatively use the multi-class linear discriminant analysis to generate the initial **r** [31].

4. Experiments and results

4.1. EEG data and evaluation setting

The proposed method was evaluated using the BCI Competition IV Data Set I [15], which was recorded from four human subjects performing motor imagery tasks. Fifty-nine EEG channels were used that were most densely distributed over sensorimotor areas. Each subject participated in two sessions: the calibration session and the evaluation session.

- *Calibration.* Each subject selected two classes of motor imagery from left hand, right hand or foot. During data collection, a visual cue was displayed in a computer screen to the subjects who then started to perform a motor imagery task for 4 s according to the cue. Each subject performed a total of 200 motor imagery tasks (balanced between the two classes). Consecutive motor imagery tasks were interleaved with a 4 s break.
- *Evaluation.* The subjects followed the soft voice commands from an instructor to perform motor imagery tasks of varying length between 1.5 and 8 s. Consecutive tasks were also interleaved with a varying length interval from 1.5 to 8 s. This session was meant for validation of un-cued motor imagery classification algorithms (see [8]).

This experiment involves an offline cross-validation study. The cross-validation assessed how the results generated by the method would generalize to an independent data set. It involved partitioning each subject's data into ten continuous blocks of equal duration, alternately performing the learning on one block (called the training set), and validating the learned model on the others (aggregated as the test set).

We applied the proposed method to generate a univariate output continuously in an un-cued manner in both training set and test set. After band-pass filtering (band-pass filter selection to be described later) of the whole training/test set, we employed a 2 s shifting window to extract EEG segments with a shift step of 0.5 s. The EEG segments were labeled according to the contained samples, and there were two possible situations: if all the samples in a EEG segment were during the same motor imagery period or the same NC period, the EEG segment was labeled as belonging to the specific motor imagery class or the NC class; otherwise, the EEG segment was considered as transition data and was discarded from this study.

EEG segments contaminated with electrooculography (EOG) were removed. Specifically, a particular threshold was selected for each data set after visual inspection of the waveforms; the EEG segments that contained larger-than-threshold samples were rejected. Statistically, approximately 5–8% of EEG segments were removed from each data set. We would like to emphasize that removal of artifacts is important for BCI research, since neurological phenomena should be the only source of control in any BCI system. Artifacts are undesirable signals that can interfere with neurological phenomena [32]. In a self-paced BCI system, they can affect the performance of the system either by changing the shape of the neurological phenomenon during the control period or by mimicking the neurological phenomenon during NC. The consequence would be either a decreased true positive rate (TPR) or an increased false positive rate (FPR).

We employed a recently developed algorithm [26] to compute spatio-spectral features. In the training set, it first decomposed EEG data into a continuous array of pass-bands, using 8 Chebyshev Type II filters centered at frequencies from 8 Hz to 32 Hz at a constant interval in logarithm. All the filters had a constant Q factor (bandwidth-to-center frequency) of 0.33. The training EEG segments were then extracted from the filtered data as mentioned earlier. The CSP method learned the spatial filtering and was applied to the segments, separately in each pass-band. The log-power features from all the bands were aggregated to form a raw feature vector, from which a small subset of the features was selected using a feature selection technique based on mutual information. The current implementation differed slightly from [33] in that it constrained the algorithm to select features from one band. Therefore, it effectively selected a single-frequency band for each subject independently. The selected frequency band and the CSP spatial patterns were used in the procedure described in section 2 to generate spatio-spectral features that represent EEG segments in both training set and test set.

Note that, unlike the traditional CSP method, the feature extraction approach need not pair the spatial patterns and

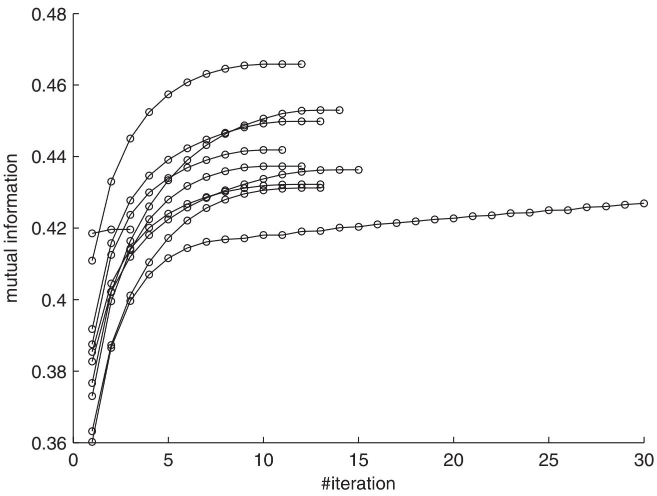

Figure 2. Convergence of the gradient-based optimization procedure. The graph shows the curve of the mutual information estimate (see section 2) over the iterations. Each curve corresponds to a particular testing result from ten-fold cross-validation using the calibration data from subject ‘b’.

can select an arbitrary number of features. Tentatively we considered selecting three features only, though a larger number of 4 or 5 yielded similar results according to our tests.

The study normalized the features to the range [−1 1], and applied the proposed learning method (OLM) to optimize the linear projection from the features into a univariate control signal. We also compared the method to two linear regression methods, including linear mean-square error regression (LMS) and a linear support vector regression (SVR_L) algorithm. The LMS method was from MATLAB, while SVR_L used ϵ-SVR from the LibSVM toolbox [34]. The spatio-spectral features were extracted before these methods under comparison were applied such that every method processed the same set of features.

4.2. Convergence of the optimization algorithm

We tested the optimization algorithm described in section 3 in each fold during the cross-validation, and observed the number of iterations it took before this stop criterion was met: the increase in the mutual information estimate was less than 10^{-4}. Figure 2 plots the results for the cross-validation runs on the data from subject ‘b’, where each fold produced a particular curve of mutual information estimate over the number of iterations. It can be seen that the mutual information picked up by approximately 10% in the first few iterations, and converged quickly in about ten iterations. Our analysis shows that, in all the cross-validation folds from every subject, the iteration took minimum 3 and maximum 30 iterations to meet the stop criterion, with a mean of 6.10 and a standard deviation of 5.81 iterations.

4.3. Area under the ROC curve analysis

The area under the ROC curve (AUC) is a widely used measure for assessing signal detection systems. It is equal to the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative one [35]. In this study, we pay attention to the AUC for low false positive rates, since prior studies on self-paced BCI have indicated the importance of low FPR [36] for practical BCI operation. Of particular interest here is the low FPR range $\leqslant$ 10%, which is selected according to [8] that reported a false positive rate around 10% in most runs of a motor imagery BCI, as well as [36] that emphasized FPR<8% in the analysis graphs (though it was not a motor imagery BCI).

For AUC analysis we need to consider the definition of FPR and TPR. The definition can be on a sample-by-sample basis or on an event-by-event basis [3]. Sample-by-sample approach breaks each event or nonevent up into a number of pieces (like the overlapping EEG segments described earlier), and assesses the outputs from each piece independently. On the other hand, the event-by-event approach determines the occurrence of events (each representing a discrete intent by the user) and nonevents [37].

This study took the sample-by-sample approach to evaluate the FPR and TPR measures and the AUC, for the following reasons. First, as suggested in [37] (section 3.2), the sample-by-sample metrics such as FPR can serve as a reliable performance measure for comparing different systems on the same data, provided that the sample-by-sample labeling is reasonable and is done prior to the system outputs. Second, event-by-event analysis generally requires additional post-processing such as dwelling [3] that converts sample-based outputs to event-based outputs. The additional process will introduce more variables to the system performance metrics. Furthermore, event-by-event analysis depends strongly on the definition of the event intervals, which are entirely up to the user and are subject to change during BCI operation.

The study adopted a multi-class estimation method from [38] to compute the AUC in this three-class setting (including two motor imagery classes plus NC). Please see the appendix for details.

We conducted two cross-validation tests to evaluate the method’s generalization performance. The first test ran the cross-validation procedure as described earlier in section 4.1, using data in the calibration session only. The second one ran almost the same procedure except that the test set was replaced by the evaluation session data. The two tests were denoted respectively by ‘Calib’ and ‘Calib-Eval’. We also ran paired t-tests to assess whether the mean AUC scores were statistically different between OLM and LMS/SVR_L. Besides, a virtual subject was created and named as ‘All’, which aggregated all the scores from the four subjects.

The results are shown in tables 1 and 2. In terms of the AUC for the full range of FPR, OLM achieved a significantly higher mean AUC score in all the tests except that on the calibration session from subject ‘g’. For the low range of FPR, OLM yielded significant improvement in the AUC in all the tests. Besides, the scores were comparable between the ‘Calib’ and the ‘Calib-Eval’ tests.

Table 1. AUC scores, for full false positive rate range. In mean (std) format. Refer to section 4.3 for related information. The first column denotes the subjects; the second column denotes two types of cross-validation study (see section 4.3): 'Calib CV' stands for cross-validation in the calibration session; 'Calib-Eval' replaces the test set in 'Calib CV' by the evaluation session. The two rightmost columns, under p-value, show the t-test results for the hypothesis that the proposed method OLM produced the same mean AUC as LMS (least mean-square error regression) or SVR_L (linear support vector regression). Significant p-values (rejection at the 5% significance level) are in bold.

		Mapping method			p-value	
Subject	Test	LMS	SVR_L	OLM	OLM = LMS	OLM = SVR_L
a	CalibCV	41.7 (4.16)	42.0 (4.36)	**49.0 (6.98)**	**0.02**	**0.02**
	Calib_Eval	36.3 (0.13)	36.3 (0.15)	**48.8 (7.04)**	**0.01**	**0.01**
b	CalibCV	38.7 (6.21)	43.0 (8.17)	**54.2 (3.06)**	**<0.01**	**<0.01**
	Calib_Eval	36.2 (1.53)	39.3 (1.94)	**44.3 (0.58)**	**<0.01**	**<0.01**
f	CalibCV	39.8 (3.09)	39.6 (3.13)	**49.6 (4.69)**	**<0.01**	**<0.01**
	Calib_Eval	34.42 (0.14)	34.51 (0.22)	**40.5 (3.46)**	**<0.01**	**<0.01**
g	CalibCV	42.6 (4.62)	42.4 (4.66)	**45.2 (4.59)**	0.21	0.19
	Calib_Eval	36.7 (1.12)	36.7 (1.17)	**41.1 (3.44)**	**0.01**	**0.01**
All	CalibCV	40.7 (4.73)	41.8 (5.36)	**49.5 (5.83)**	**<0.01**	**<0.01**
	Calib_Eval	35.9 (1.28)	36.7 (2.07)	**42.4 (4.45)**	**<0.01**	**<0.01**

Table 2. AUC scores, for low false positive ⩽10%. Refer to section 4.3 and the caption of table 1 for related information.

		Mapping method			p-value	
Subject	Test	LMS	SVR_L	OLM	OLM = LMS	OLM = SVR_L
a	CalibCV	3.27 (1.42)	3.36 (1.48)	**10.7 (6.65)**	**0.01**	**0.01**
	Calib_Eval	3.48 (0.06)	3.48 (0.12)	**11.6 (6.80)**	**<0.01**	**<0.01**
b	CalibCV	5.77 (3.26)	6.75 (3.38)	**11.4 (3.70)**	**<0.01**	**<0.01**
	Calib_Eval	5.40 (0.78)	6.29 (0.48)	**8.72 (0.79)**	**<0.01**	**<0.01**
f	CalibCV	2.86 (1.53)	2.77 (1.42)	**10.8 (3.65)**	**<0.01**	**<0.01**
	Calib_Eval	6.03 (0.17)	6.02 (0.21)	**10.2 (1.83)**	**<0.01**	**<0.01**
g	CalibCV	2.35 (0.87)	2.39 (0.84)	**6.30 (3.56)**	**0.01**	**0.01**
	Calib_Eval	2.64 (0.53)	2.63 (0.51)	**9.34 (5.95)**	**<0.01**	**<0.01**
All	CalibCV	3.56 (2.32)	2.79 (2.63)	**9.81 (4.86)**	**<0.01**	**<0.01**
	Calib_Eval	4.39 (1.47)	4.61 (1.64)	**10.0 (4.59)**	**<0.01**	**<0.01**

5. Discussions

Mutual information has been generally introduced to BCI as an evaluation method for given outputs and class labels [21], or as a feature selection criterion [21, 39]. In the feature extraction case, however, only simple uni-modal Gaussian models were considered. Furthermore, no numerical solution based on mutual information was developed to learn and optimize 1D signal design. In contrast, the purpose of this work was not only to introduce a non-parametric mutual information estimate that can account for high complexity of the EEG data in self-paced BCI, but also to derive, from the mutual information estimate, a numerical solution to the optimization of BCI, especially in the linear transformation from EEG features to a control signal.

The experimental results show that the proposed method (OLM) can significantly improve the performance in terms of the AUC. Here we discuss how OLM will affect the 1D control signal and possibly why it can significantly improve over the regression method. To this end, figure 3 illustrates a comparison of OLM versus LMS in forms of their output control signals. In the upper part of the figure, LMS mapped the three classes into three pre-set target values, while OLM was approaching maximum mutual information instead of any target values for the linear mapping. In LMS, there was apparently a discrepancy between training and test in the output distributions. On the other hand, OLM has produced more consistent results.

If we took the means of the control signal generated by OLM as the target values for LMS, LMS would produce similar results to that of OLM. This suggests that manipulation of the target values in an *ad hoc* manner for a regression machine may lead to improved BCI performance. On the other hand, regression using intuitive but improper target values may degrade the BCI system.

To further compare the outputs of different methods, we plot in figure 4 the time courses of the 1D signal generated by SVR_L and OLM, respectively. The first half of the calibration session was used for training and the rest for testing. Only the outputs from testing are displayed here. As can be seen, the outputs varied both among the subjects and between the methods. Apparently, the outputs show large difference between the two methods in all subjects except 'g'. This may link to the abovementioned finding that the calibration AUC score showed significant difference between SVR_L and OLM for the first three subjects but not the last

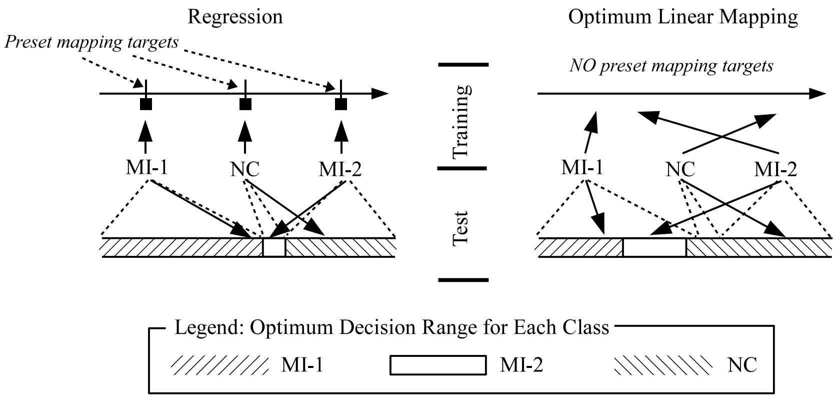

Figure 3. Control signal generated by regression and the proposed OLM method. The data were from a cross-validation fold of subject 'b' from the BCI Competition IV Data Set I. See section 5 for related discussions. The arrows starting from each class point to the respective mean values after mapping.

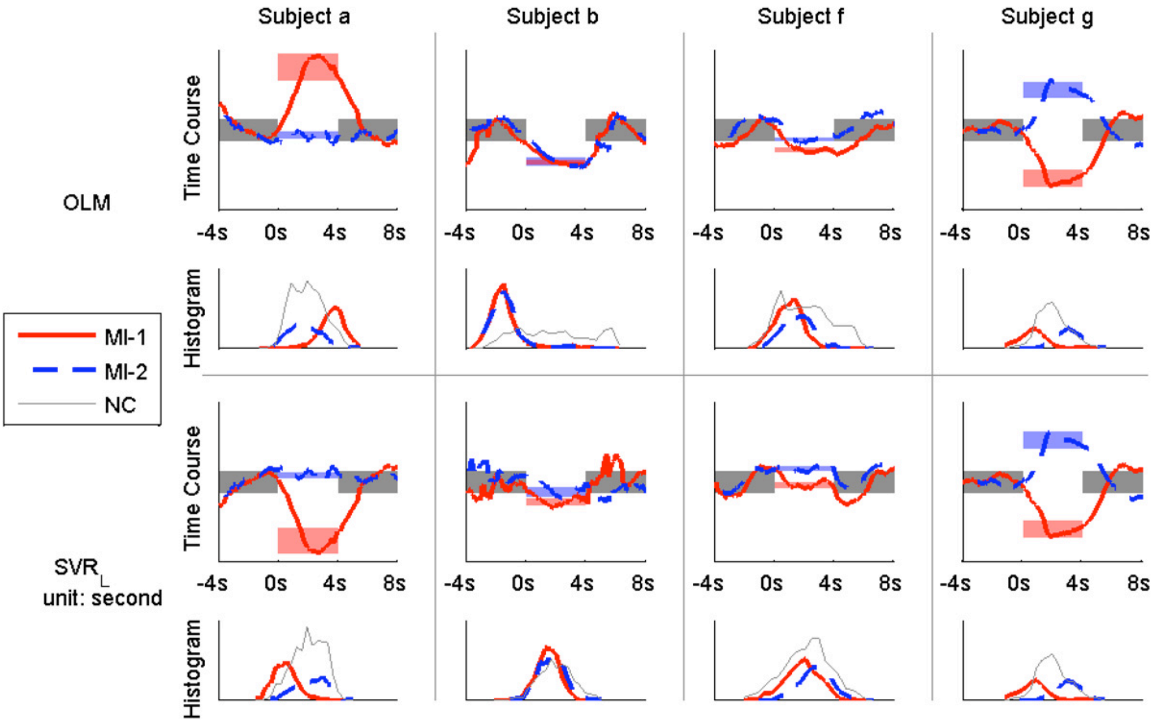

Figure 4. Average time courses of output signal. The upper and lower panels show the average output signal produced by the proposed method (OLM) and a regression method (SVR_L), respectively. In the upper row of each panel, each color curve represents the average time course (unit: second) of output signal for a particular class (MI-1 or MI-2) of motor imagery trials, where time point 0 represents the cue time and the subjects were performing motor imagery in the [0s 4s] period, and the color rectangles denote the standard deviation by height, and the time range of motor imagery or NC by horizontal span. The output signal in each graph is normalized such that the standard deviation of NC output is 1. The lower panel plots the histograms of the output signal for the three classes separately.

(This figure is in colour only in the electronic version)

one (see table 1). Especially prominent difference in the time courses can be found in subject 'b' (corresponding to figure 3. From the histograms of the output, it appears that SVR_L was not able to differentiate any class, whereas OLM managed to automatically distinguish the two motor imagery classes from NC.

In the above report, we considered a sample-by-sample AUC. Practically, event-by-event performance measure will be of greater interest to the end user. As section 4.3 describes, converting sample-based measures to event-based measures would introduce new variables that are difficult to handle rigorously. Nevertheless, it is worthwhile to consider the low sample-based FPR (detection rate at 2 Hz) at 10% together with a simple event detection using a dwelling period [3] of 2 s. Hence, a positive event will be detected if the sample-by-sample detections in the last 2 s time period are all positive. Consider the worst case in which the false sample-based detections are coherent in the 2 s time frame, the system would produce three false detections per minute during NC. In the ideal case where the sample-based detections are independent, the false event detection rate would be very low: our Monte

Carlo simulation showed that it would be 0.1±0.3 per minute during NC.

It is also interesting to compare the computational cost of the method with the conventional ones. Both LMS and the proposed method perform a simple linear transformation. The computational cost is minimal. The training process, on the other hand, depends on the complexity of the learning algorithm and the implementation. We measured the time cost of the proposed method implemented in MATLAB without optimization, and found that it could finish optimization in each cross-validation training in <15 s. That seems to be quite acceptable to offline calibration.

It is worthwhile to discuss the proposed mutual information estimate in relation to evaluation criteria for BCI research. For assessing synchronous system performance, various criteria have been well established and studied [40, 41], including classification accuracy, Cohen's Kappa coefficient [42], information transfer rate [43, 44], and more [45]. For assessing self-paced BCI performance, the evaluation criterion is still an ongoing research topic [21, 37]. Some studies used information transfer rate [13, 18], while others resorted to the ROC analysis [3, 9, 36]. The proposed estimate has been used to optimize the BCI control signal and successfully improved the performance in terms of the AUC. However, since it is based on a non-parametric sample-based estimation method, the estimate cannot serve as a metric for comparing different data and/or different methods. Nevertheless, future extension of the proposed estimate by appropriate normalization may solve the problem.

Lastly, it would be interesting to extend the proposed method to two- or multi-dimensional control signals [46]. Although it poses considerable difficulties to the user and challenges to BCI system design, controlling two variables simultaneously bears important potential for high-performance BCI systems [46].

6. Conclusion

In this paper we considered the issue of designing a continuous control signal for a self-paced BCI. We proposed a mutual information estimate as the objective function for the design, and derived from it a gradient-based optimization algorithm to learn the OLM from EEG features to a univariate. Results of our offline simulation showed that the optimization algorithm converged quickly, and the proposed method yielded significantly better results in terms of the AUC than the conventional regression approach. Therefore, we expect that continuing research will further explore the mutual information estimate as the criterion for designing and optimizing one- or even multi-dimensional BCI.

Acknowledgments

We thank the provider of the BCI Competition Data Set I [15]. We also appreciate constructive comments given by Dr Fabien Lotte.

Appendix. Multi-class AUC

As a system performance measure, AUC has the attractive property that it avoids specifying the costs of the different kinds of misclassification [38]. Besides, so far most of the techniques for computing AUC are non-parametric, which is insensitive to varying class distributions. Finally, it has been demonstrated in [47] that the AUC is superior to the often used misclassification rate for the evaluation of system performance.

The basic form of the AUC is only applicable to two-class cases. Various methods have been proposed to extend the AUC for multi-class problems [38, 48]. In this work we adopt the algorithm proposed in [38]. The algorithm is described below.

Suppose there are n_c classes. The multi-class AUC is a weighted average of AUC over all pairs of classes:

$$\bar{A} = \frac{2}{n_c(n_c-1)} \sum_{i=1}^{n_c} \sum_{j=i+1}^{n_c} \rho(i,j) A(i,j), \tag{A.1}$$

where $A(i, j)$ is the AUC for the ith and the jth classes. $\rho(i, j)$ is the weight that equals the *a priori* probability that the true class label is either i or j. The probability can be estimated from the data.

To compute $A(i, j)$, we first note that the area under the curve is an integral [38]:

$$A = \int G_1(u) \mathrm{d}G_0(u) = \int G_1(u) g_0(u)\, \mathrm{d}u. \tag{A.2}$$

Here u is a threshold, $G_1(u)$ $(G_0(u))$ is the cumulative distribution function that describes the probability that a random positive (negative) class sample is larger than u and $g_0(u)$ is the density function that a random negative class sample equals u.

Therefore, A can be viewed as an expectation, whose sample-based approximation is

$$A = E\left[G_1(u) | u \in g_0(u)\right] \cong \frac{1}{N_0} \sum_{u_k} \hat{G}_1(u_k), \tag{A.3}$$

where u_k is the kth sample from the negative class, N_0 is the number of negative class samples and $\hat{G}_1$ is the percentile of the positive class samples larger than u_k.

For the AUC in the range of low false positive rate, e.g. $G_0(u) < f_{\mathrm{low}}$, it is straightforward to adapt the above equation as follows:

$$\begin{aligned} A_{\mathrm{low}} &= E\left[G_1(u) | u \in g_0(u) \text{ AND } G_0(u) < f_{\mathrm{low}}\right] \\ &\cong \frac{1}{N_0} \sum_{u_k} \hat{G}_1(u_k) \delta(G_0(u_k) - f_{\mathrm{low}}), \end{aligned} \tag{A.4}$$

where $\delta(\leqslant 0) = 0$ and $\delta(> 0) = 1$.

References

[1] Wolpaw J R 2007 Brain–computer interfaces as new brain output pathways *J. Physiol.* **579** 613–9

[2] Nijholt A and Tan D 2008 Brain–computer interfacing for intelligent systems *IEEE Intell. Syst.* **23** 72–9

[3] Townsend G, Graimann B and Pfurtscheller G 2004 Continuous EEG classification during motor imagery simulation of an asynchronous BCI *IEEE Trans. Neural Syst. Rehabil. Eng.* **12** 258–65

[4] Pfurtscheller G and Neuper C 2001 Motor imagery and direct brain–computer communication *Proc. IEEE* **89** 1123–34

[5] Li Y Q, Li H Q, Guan C T and Chin Z Y 2008 An self-training semi-supervised SVM algorithm and its application in an EEG-based brain computer interface speller system *Pattern Recognit. Lett.* **29** 1285–94

[6] Wu W, Gao X R, Hong B and Gao S K 2008 Classifying single-trial EEG during motor imagery by iterative spatio-spectral patterns learning (ISSPL) *IEEE Trans. Biomed. Eng.* **55** 1733–43

[7] Donchin E, Spencer K M and Wijesinghe R 2000 The mental prosthesis: assessing the speed of a P300-based brain–computer interface *IEEE Trans. Rehabil. Eng.* **8** 174–9

[8] Scherer R, Lee F, Schlögl A, Leeb R, Bischof H and Pfurtscheller G 2008 Toward self-paced brain–computer communication: navigation through virtual worlds *IEEE Trans. Biomed. Eng.* **55** 675–82

[9] Fatourechi M, Ward R K and Birch G E 2008 A self-paced brain–computer interface system with a low false positive rate *J. Neural Eng.* **5** 9–23

[10] Bashashati A, Mason S G, Borisoff J F, Ward R K and Birch G E 2007 A comparative study on generating training-data for self-paced brain interfaces *IEEE Trans. Neural Syst. Rehabil. Eng.* **15** 59–66

[11] Millán J d R and Mouriño J 2003 Asynchronous BCI and local neural classifiers: an overview of the adaptive brain interface project *IEEE Trans. Neural Syst. Rehabil. Eng.* **11** 159–61

[12] Galán D R, Nuttin M, Lew E, Ferrez P W, Vanacker G, Philips J and Millán J d R 2008 A brain-actuated wheelchair: asynchronous and non-invasive brain–computer interfaces for continuous control of robots *Clin. Neurophysiol.* **119** 2159–69

[13] Zhang H, Guan C and Wang C 2008 Asynchronous p300-based brain–computer interfaces: a computational approach with statistical models *IEEE Trans. Biomed. Eng.* **55** 1754–63

[14] Obermaier B, Muller G R and Pfurtscheller G 2003 Virtual keyboard controlled by spontaneous EEG activity *IEEE Trans. Neural Syst. Rehabil. Eng.* **11** 422–6

[15] Blankertz B, Dornhege G, Krauledat M, Müller K-R and Curio G 2007 The non-invasive Berlin brain–computer interface: fast acquisition of effective performance in untrained subjects *NeuroImage* **37** 539–50

[16] Blankertz B, Losch F, Krauledat M, Dornhege G, Curio G and Müller K-R 2008 The Berlin brain–computer interface: accurate performance from first-session in BCI-naïve subjects *IEEE Trans. Biomed. Eng.* **55** 2452–62

[17] Kübler A, Nijboer F, Mellinger J, Vaughan T M, Pawelzik H, Schalk G, McFarland D J, Birbaumer N and Wolpaw J R 2005 Patients with ALS can use sensorimotor rhythms to operate a brain–computer interface *Neurology* **64** 1775–7

[18] Scherer R, Müller G R, Neuper C, Graimann B and Pfurtscheller G 2004 An asynchronously controlled EEG-based virtual keyboard: improvement of the spelling rate *IEEE Trans. Biomed. Eng.* **51** 979–84

[19] BCI Competition IV http://www.bbci.de/competition/

[20] Cover T M and Thomas J A 2006 *Elements of Information Theory* 2nd edn (New York: Wiley)

[21] Schlögl A, Kronegg J, Huggins J and Mason S G 2007 Evaluation criteria for BCI research *Toward Brain–Computer Interfacing* ed G Dornhege, J d R Millan, T Hinterberger, D McFarland and K-R Müller (Cambridge, MA: MIT Press) pp 327–42

[22] Pfurtscheller G, Neuper C, Flotzinger D and Pregenzer M 1997 EEG-based discrimination between imagination of right and left hand movement *Electroencephalogr. Clin. Neurophysiol.* **103** 642–51

[23] Muller-Gerking J, Pfurtscheller G and Flyvbjerg H 1999 Designing optimal spatial filtering of single trial EEG classification in a movement task *Clin. Neurophysiol.* **110** 787–98

[24] Blankertz B, Tomioka R, Lemm S, Kawanabe M and Müller K-R 2008 Optimizing spatial filters for robust EEG single-trial analysis *IEEE Signal Process. Mag.* **25** 41–56

[25] Lemm S, Blankertz B, Curio G and Müller K-R 2005 Spatio-spectral filters for improving the classification of single trial EEG *IEEE Trans. Biomed. Eng.* **52** 1541–8

[26] Zhang H, Guan C and Wang C 2009 Spatio-spectral feature selection based on robust mutual information estimate for brain–computer interfaces *Ann. Int. Conf. of the IEEE Engineering in Medicine and Biology Society* pp 2391–8

[27] Ramoser H, Muller-Gerking J and Pfurtscheller G 2000 Optimal spatial filtering of single trial EEG during imagined hand movement *IEEE Trans. Rehabil. Eng.* **8** 441–6

[28] Viola P and Wells W M III 1997 Alignment by maximization of mutual information *Int. J. Comput. Vis.* **24** 137–54

[29] Bowman A W and Azzalini A 1997 *Applied Smoothing Techniques for Data Analysis: The Kernel Approach with S-Plus Illustrations* (New York: Oxford University Press)

[30] Nocedal J and Wright S J 1999 *Numerical Optimization* (Berlin: Springer)

[31] Fukunaga K 1990 *Introduction to Statistical Pattern Recognition* (New York: Academic)

[32] Fatourechi M, Fatourechi A, Ward R K and Birch G E 2007 EMG and EOG artifacts in brain–computer interface systems: a survey *Clin. Neurophys.* **118** 480–94

[33] Zhang H, Guan C, Ang K K and Wang C 2009 Winning algorithm of BCI Competition IV Data Set I: uncued motor imagery classifier. Learning discriminative patterns for EEG-based motor imagery detection *Lecture Notes on Computer Science: Special Volume on the BCI Competition IV* (Berlin: Springer) at press

[34] Chang C-C and Lin C-J 2001 LIBSVM: a library for support vector machines http://www.csie.ntu.edu.tw/∼cjlin/libsvm

[35] Fawcett T 2006 An introduction to ROC analysis *Pattern Recognit. Lett.* **27** 861–74

[36] Borisoff J F, Mason S G, Bashashati A and Birch G E 2004 Brain–computer interface design for asynchronous control applications: improvements to the LF-ASD asynchronous brain switch *IEEE Trans. Biomed. Eng.* **51** 985–92

[37] Mason S, Kronegg J, Huggins J, Fatourechi M and Schlöegl A 2006 Evaluating the performance of self-paced BCI technology *Technical Report* Neil Squire Society

[38] Hand D J and Till R J 2001 A simple generalisation of the area under the ROC curve for multiple class classification problems *Mach. Learn.* **45** 171–86

[39] Grosse-Wentrup M and Buss M 2008 Multiclass common spatial patterns and information theoretic feature extraction *IEEE Trans. Biomed. Eng.* **55** 1991–2000

[40] Kronegg J, Voloshynovskiy S and Pun T 2005 Analysis of bit-rate definitions for brain–computer interfaces *Int. Conf. on Human–Computer Interaction*

[41] Fatourechi M, Ward R K, Mason S G, Huggins J, Schlögl A and Birch G E 2008 Comparison of evaluation metrics in classification applications with imbalanced datasets *Int. Conf. on Machine Learning and Applications*

[42] Schlögl A, Lee F, Bischof H and Pfurtscheller G 2005 Characterization of four-class motor imagery EEG data for the BCI-competition 2005 *J. Neural Eng.* **4** L14

[43] Wolpaw J R, Birbaumer N, Heetderks W J, McFarland D J, Peckham G S P Hunter, Donchin E, Quatrano L A, Robinson C J and Vaughan T M 2000 Brain–computer interface technology: a review of the first international meeting *IEEE Trans. Biomed. Eng.* **8** 164–73

[44] Obermaier B, Neuper C, Guger C and Pfurtscheller G 2001 Information transfer rate in a five-classes brain–computer interface *IEEE Trans. Neural Syst. Rehabil. Eng.* **9** 283–8

[45] Seno B Dal, Matteucci M and Mainardi L 2010 The utility metric: a novel method to assess the overall performance of discrete brain–computer interfaces *IEEE Trans. Biomed. Eng.* **18** 20–8

[46] Wolpaw J R and McFarland D J 2004 Control of a two-dimensional movement signal by a noninvasive brain–computer interface in humans *Proc. Natl Acad. Sci.* **101** 17849–54

[47] Huang J and Ling C X 2005 Using AUC and accuracy in evaluating learning algorithms *IEEE Trans. Knowl. Data Eng.* **17** 299–310

[48] Van Calster B, Van Belle V, Condous G, Bourne T, Timmerman D and Van Huffel S 2008 Multi-class AUC metrics and weighted alternatives *IEEE Int. Joint Conf. on Neural Networks*