Open access • Journal Article • DOI:10.1109/JPROC.2015.2404941

Signal Processing Approaches to Minimize or Suppress Calibration Time in Oscillatory Activity-Based Brain–Computer Interfaces — Source link

Fabien Lotte

Published on: 18 May 2015

Signal processing approaches to minimize or suppress calibration time in oscillatory activity-based Brain-Computer Interfaces

Fabien LOTTE

Abstract—One of the major limitations of Brain-Computer Interfaces (BCI) is their long calibration time, which limits their use in practice, both by patients and healthy users alike. Such long calibration times are due to the large between-user variability and thus to the need to collect numerous training electroencephalography (EEG) trials for the machine learning algorithms used in BCI design. In this paper, we first survey existing approaches to reduce or suppress calibration time, these approaches being notably based on regularization, user-to-user transfer, semi-supervised learning and a-priori physiological information. We then propose new tools to reduce BCI calibration time. In particular, we propose to generate artificial EEG trials from the few EEG trials initially available, in order to augment the training set size. These artificial EEG trials are obtained by relevant combinations and distortions of the original trials available. We propose 3 different methods to do so. We also propose a new, fast and simple approach to perform user-to-user transfer for BCI. Finally, we study and compare offline different approaches, both old and new ones, on the data of 50 users from 3 different BCI data sets. This enables us to identify guidelines about how to reduce or suppress calibration time for BCI.

Index Terms—Brain-Computer Interfaces (BCI), ElectroEncephaloGraphy (EEG), signal processing, machine learning, small sample settings, calibration

1 Introduction

Brain-Computer Interfaces (BCI) are systems that enable their users to interact with a computer by means of brain activity only, this activity being typically measured by ElectroEncephaloGraphy (EEG) [1]. A typical BCI example would be a system with which a user can move a computer cursor on a screen towards left or right by imagining left or right hand movements respectively, see, e.g., [2][3]. BCI have proven to be a very promising technology, e.g., to provide some communication abilities to severely disabled users [4], as a new control device for gaming for healthy users [5][6] or to design adaptive human-computer interfaces that can react to the user's mental state [7], among many other promising applications [8][1]. However, most of these applications are prototypes and current BCI are still scarcely used outside laboratories. Among the different reasons currently preventing BCI from being widely used outside laboratories for practical applications, one can cite their lack of robustness and reliability, their cumbersome setup and their long calibration time [1][8][9].

This last point, namely the long calibration time of BCI, is due to the fact that many examples of the user's EEG signals must be recorded in order to calibrate the BCI using machine learning [10]. Indeed, most current BCI are based on machine learning and are organized as described in Figure 1. First, EEG signals are acquired from the BCI user. Then, they are preprocessed, usually with different types of filters, to remove noise and enhance the relevant information. The relevant information contained in the signals is extracted under the form of values called *features*. Such features are given as input to a classifier whose aim is to identify which feature values correspond to which class of EEG signals (e.g., EEG signals corresponding to left hand or right hand movement imagination). The signals class is then

- *F. Lotte is with Inria Bordeaux Sud-Ouest, 200 avenue de la vieille tour, 33405, Talence Cedex, France. E-mail: fabien.lotte@inria.fr*

Manuscript received September 15, 2014.

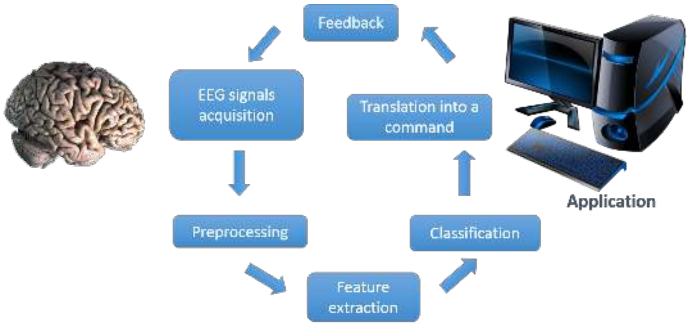

Fig. 1. A typical architecture of a BCI based on machine learning

translated into a command for the computer (e.g., an imagined left hand movement is translated into moving a cursor towards the left) before providing feedback to the user about the mental command recognized by the system. Typically, the preprocessing, feature extraction and/or classification algorithms are all calibrated specifically for each user. This need for user-specific calibration of a BCI is due to the large inter-user variability, making the creation of a universal BCI difficult, if not impossible. Unfortunately such calibrations and associated necessary data collection are both inconvenient and time consuming. For instance, a typical online BCI based on motor-imagery (i.e., imagination of movements) requires a calibration time of about 20 minutes [11], which is still far too long. As a comparison, nobody would indeed use a computer mouse if it required a 20 minute-long calibration before each use. Therefore, an ideal BCI system should require a calibration time that is as short as possible or even do not require calibration at all.

Interestingly enough, there are 2 main types of BCI systems, which exploit 2 different types of information: 1) Event-Related Potentials (ERP)-based BCI, which are based on brain responses to attended stimulus. A typical example of an ERP is the P300, which is a Positive deflection of the EEG signals occurring about 300ms after a rare, relevant and attended stimulus. ERP have been extensively used to design BCI-based spellers exploiting the user's brain responses to attended visual stimulus, including the P300 and the N200 [12]; 2) Oscillatory activity-based BCI which exploit changes in the amplitudes of EEG oscillations due to spontaneously performed mental tasks (e.g., a movement imagination). Figure 2 gives an example of EEG oscillations (here the so-called sensorimotor rhythms in the 8-30Hz frequency band) whose amplitude is changing due to hand movement imagination.

It should be noted that although the necessary long calibrations mentioned above affect all types of BCI, ERP-based BCI, e.g., spellers, do not suffer from this issue as much as oscillatory activity-based BCI do. Indeed, ERP are usually more stable across users than EEG oscillations [13][14], and the specific structures of ERP-spellers can be exploited in a smart way to reduce or suppress the need for calibration [15][16]. As a consequence, there are now a number of efficient solutions to drastically reduce or suppress calibration time for ERP-spellers, see e.g., [17][18][19][20][21][15][16][22]. In contrast, reducing or suppressing calibration time of oscillatory activity-based BCI such as mental imagery-based BCI is still an open and challenging research question. Moreover, oscillatory activity-based BCI can be useful for a number of applications for which ERP-based BCI cannot be used, including (but not limited to) self-paced BCI control [23][24], stroke rehabilitation [25][26] or passive BCI that monitor and adapt an application based on users' mental states such as attention or workload [27][7][28].

Therefore, this paper explores signal processing and machine learning tools to reduce or suppress calibration times in oscillatory activity-based BCI. More particularly, this paper first contributes a survey of the calibration time reduction or suppression methods proposed so far. Then, it contributes a couple of new approaches to address this objective. Finally, it compares several of these approaches in order to identify guidelines about how to efficiently reduce or suppress BCI calibration times.

This paper is organized as follows: Section 2 reviews existing approaches to reduce (Section 2.2) or suppress (Section 2.3) calibration times for oscillatory-activity BCI. Then, Section 3 proposes a couple of new approaches to reduce calibration time, notably three methods

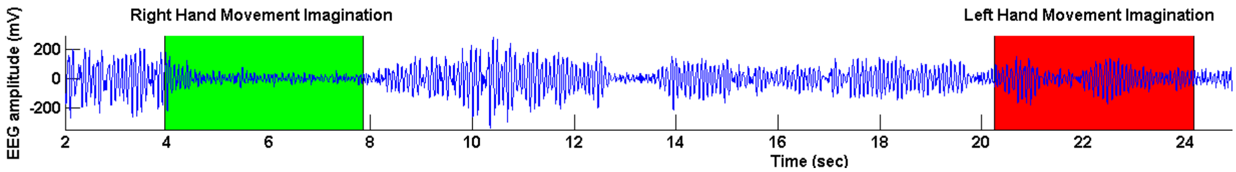

Fig. 2. Example of EEG oscillations in the 8-30Hz band (i.e., the sensorimotor rhythm), over the left motor cortex (after spatial filtering). The decrease in signal amplitude in that band during right hand motor imagery is clearly visible and can thus be exploited to operate a BCI.

based on artificial EEG data generation and a simple user-to-user transfer approach. Section 4 evaluates and studies the newly proposed approaches as well as several existing ones. Finally, Section 6 extracts guidelines from these analyses about which method to use. The paper is concluded in Section 7.

2 State-of-the-art

BCI calibration time reduction being a difficult problem to solve, it has been only scarcely addressed in the literature so far. Nevertheless, a few authors proposed interesting approaches to do so. However, before discussing advanced technical approaches, the following section first presents the standard way to design oscillatory activity-based BCI. Indeed, understanding this standard design and its limitations is essential to design and understand more advanced approaches that address these limitations.

2.1 Standard oscillatory activity-based BCI design

A typical oscillatory activity-based BCI is designed around two main algorithms: the Commom Spatial Patterns (CSP) algorithm to optimize spatial filters[1] and the Linear Discriminant Analysis (LDA) algorithm for classification. The CSP algorithm aims at learning spatial filters such that the variance of the spatially filtered signals is maximized for one class (e.g., one mental imagery task) and minimized for the other class. Since the variance of a band-pass filtered signal corresponds to the band-power of the signal in that band, CSP optimizes spatial filters that lead to optimally discriminant band-power features [29]. This is particularly interesting and relevant for the design of oscillatory activity-based BCI since such BCI exploit changes in EEG oscillations amplitude, i.e., changes in the EEG signals band power. As an example, during left hand motor imagery (i.e., the imagination of a left hand movement), the EEG band power in the mu (8-12Hz) and beta (16-24Hz) frequency bands will decrease in the right sensorimotor cortex [30]. More formally, optimizing CSP spatial filters w (w being a weight vector[2]) consists in extremizing the following function:

$$J_{CSP}(w) = \frac{wC_1w^T}{wC_2w^T} \tag{1}$$

where C_i is the average spatial covariance matrix of the band-pass filtered EEG signals from class i, and T denotes transpose. Typically, these spatial covariance matrices are obtained by computing the spatial covariance matrix C_i^j from each trial T_i^j from class i, and then averaging them:

$$C_i = \frac{1}{N_i}\sum_{j}^{N_i} C_i^j = \frac{1}{N_i}\sum_{j}^{N_i} T_i^j (T_i^j)^T \tag{2}$$

with N_i the number of trials in class i and $T_i^j \in \Re^{C\times S}$ is the j^{th} EEG trial from class i, with S the number of samples in a trial, and C the number of channels. Note that the EEG signals are assumed here to be band-pass filtered and thus to have zero mean. This optimization problem is solved by Generalized Eigen Value Decomposition (GEVD) of the two matrices C_1

1. A spatial filter is a (usually linear) combination of the original channels. Performing spatial filtering helps to overcome the EEG spatial blurring that occurs due to EEG signals travelling through the skull and scalp [29]

2. In this manuscript, all vectors are assumed to be row vectors

and C_2 [29]. The spatial filters which maximize/minimize $J_{CSP}(w)$ are the eigen vectors corresponding to the largest and smallest eigen values of this GEVD, respecticely. Once the filters w are obtained, CSP feature extraction consists in filtering the EEG signals using the w and then computing the resulting signals variance. In other words, a feature f is computed as $f = log(wC_{ct}w^T)$, where C_{ct} is the current trial covariance matrix. It is common to select 3 pairs of CSP spatial filters, corresponding to the 3 largest and smallest eigenvalues, hence resulting in a trial being described by 6 CSP features.

The LDA classifier uses a linear hyperplane to separate feature vectors from two classes [10]. The intercept b and normal vector a of this hyperplane are computed as follow:

$$a^T = C^{-1}(\mu_1 - \mu_2)^T \tag{3}$$

and

$$b = -\frac{1}{2}(\mu_1 + \mu_2)a^T \tag{4}$$

with μ_1 and μ_2 being the mean feature vectors for each class and C the covariance matrix of both classes. With LDA, for an input feature vector x, the classification output is $ax^T + b$. If this output is positive, the feature vector is assigned to the first class, otherwise it is assigned to the second class. The whole process is summarized in Figure 3.

It is interesting to note that both algorithms require the estimation of covariance matrices. If few training data is available, or if the data available does not reflect most of the variability that can occur during BCI use, the covariance matrices may be poorly estimated and/or not representative of the EEG during use. This would lead to inadequate classifiers or spatial filters. This explains why many examples of EEG signals should be collected in order to calibrate BCI systems, thus making BCI calibration long and tedious. For instance, in the study in [31], the authors found that at least 40 trials per class were necessary to obtain reasonnable performances with their motor imagery-based BCI.

2.2 Calibration time reduction

So far, reducing BCI calibration time has been achieved using four main types of approaches:

- by using *Regularization approaches* that enable to perform machine learning efficiently even with few training data.
- by relying on *user-to-user transfer*, i.e., by using relevant data from other users to improve calibration for the target user for which few training data is available.
- by using *Semi-supervised learning*, which adapt and refine the initially optimized model (e.g., CSP+LDA) based on data obtained during BCI use.
- by using *a priori physiological information* about which features are expected to be relevant to design the BCI.

These different methods are surveyed hereafter.

2.2.1 Regularization approaches

As mentioned aboved, what makes CSP and LDA vulnerable to the lack of training data is the need for both algorithms to estimate representative and reliable covariance matrices. More precisely, it is known that when covariance matrices are estimated from too few training data, their largest and smallest eigen values will be respectively over and under estimated [32]. To mitigate this estimation bias, two main approaches, based on regularization, have been used: shrinkage and divide-and-conquer.

2.2.1.1 Shrinkage: Shrinkage consists in using a regularized estimate of the covariance matrices:

$$\tilde{C}_i = C_i + \lambda I \tag{5}$$

where I is the identity matrix. This regularized estimate requires to choose the extent of the regularization with the free parameter λ. Fortunately, a closed-form solution to obtain the optimal value of λ has been proposed, hence avoiding the need for costly procedures such as cross-validation [32]. Incorporating this automatically regularized estimator into CSP and LDA algorithms leads to BCI that can be trained with less training data than the standard estimator [33], hence reducing calibration time.

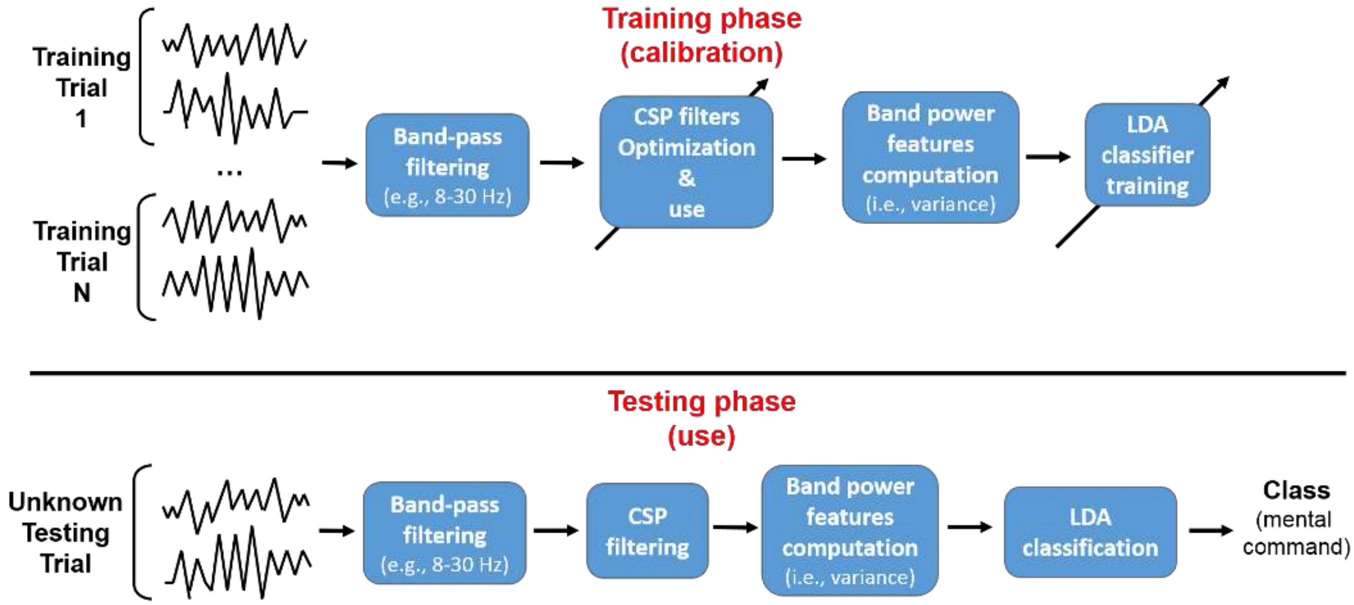

Fig. 3. The training (a.k.a. calibration) and testing steps involved in the standard design of an oscillatory activity-based BCI. During calibration, both the CSP spatial filters and the LDA classifier are optimized based on training data. During testing (i.e., actual use of the BCI with new EEG signals), the CSP filters are used to spatially filter the signals from which band power features are computed. These features are classified using the optimized LDA to identify the mental task performed by the user.

2.2.1.2 Divide-and-Conquer: Like any other parameter estimation problem, the higher the dimensionality of the covariance matrix estimations, the more difficult and thus the poorer the estimation from few data. As such, performing several easier estimations on sub-parts of the problem, thus with a lower-dimensionality, and then combining them (hence the "divide-and-conquer" name) is likely to give better results than performing a single estimation on all the data (with thus a high dimensionality). This is notably true for covariance estimation which requires to estimate d^2 parameters, with d being the dimensionality of the problem (e.g., the number of channels for CSP or the number of features for LDA). This was the idea explored with the CSP patches approach, in which several CSP filters were optimized on subsets of the available channels, and then used together, effectively outperforming CSP in small training set settings [34].

2.2.2 User-to-User transfer

Another approach that has been used to reduce calibration times in BCI, is to perform user-to-user transfer, i.e., to transfer data (EEG signals), features and/or classifiers from users for which many data are available to the target user for which there are few data. This approach has notably been used to learn CSP spatial filters or LDA classifiers in 3 different ways: multi-users covariance matrix estimation, multi-task learning and domain adaptation. They are described below.

2.2.2.1 Multi-users covariance matrix estimation: With this approach, the covariance matrix estimate used in CSP and/or LDA for each class can be regularized to ressemble the covariance matrices of other users as follows:

$$\hat{C} = \lambda C + (1 - \lambda)P \qquad (6)$$

where P can be the average covariance matrix of the other users [35], a weighted average of them [36], or an average of the covariance matrices of selected users [33]. These regularization approaches guide the optimization algorithms towards good solutions, similar to that of the other users, thus enabling a better learning from few data.

2.2.2.2 Multi-task learning: Multi-task learning consists in learning simultaneously multiple tasks that share some constraints or prior information [37]. For BCI, each such task is usually to learn a classification model for a given user, which ensures a similarity between users, and thus a better learning even for users with few training data. This has been explored successfully for multi-task linear classifier learning [38] and multi-task CSP learning [39][40][41].

2.2.2.3 Domain adaptation: Domain adaptation consists in transfering classifiers or features (and in particular spatial filters), that were optimized on other users, to the target user. In [42], spatial filters were optimized separately on the data of each other user available, and then the spatial filters that work well across users and for the target user were identified. They were finally combined with the features optimized from the target user's training data. In [43] and [44], data features optimized for the other users were transformed in order to match the feature distribution of the target user and thus increase the number of training data. In [45], classifiers were optimized on different users, and combined in a weighted scheme whose weights depend on how well the classifier can perform on the data available for the target user.

2.2.3 *Semi-supervised learning*

If few training data is available to calibrate the BCI for a given user, another approach that can be used to still reach reasonable classification performance is to use an adaptive approach [46][47], i.e., to refine the calibration of the BCI during use, as new EEG data (labelled or not) become available.

To do so, the main approach that has been used so far is semi-supervised learning. Semi-supervised learning consists in learning a model from both labelled and unlabelled data [48]. This can be achieved by learning an initial model from the labelled training data available, and then using this model prediction (typically the classifier output class) to label the test data encountered during model use. The newly labelled data are then incorporated into the already available training set and used to retrain the model, hence generally improving it. This principle is illustrated in Figure 4. For BCI, this approach has been used with CSP features and an LDA or SVM (support vector machine) classifier [49][50][51]. More precisely the CSP and LDA/SVM were first optimized on the few labelled training data available. Then, they were used on the unlabeled testing data, to predict the class label of these data. The test data classified and labelled (with the predicted class) were then incorporated into the training set, and the CSP and LDA/SVM retrained. This process was repeated for successive blocks of testing data, hence generally adaptively improving the quality of the CSP and classifiers as more testing data are encountered. Along the same line of ideas, [52] also adapted the CSP spatial filters in a unsupervised way. To do so, they measured three types of differences/similarities between the (unlabeled) test trials and the training trials in order to adapt the estimated covariance matrices from each class with the test trials; with a stronger adaptation for the class for which the test trials were more similar.

2.2.4 *A-priori physiological information*

Another way to reduce calibration time is to use a-priori information about which features or channels are likely to be useful, in order to guide the optimization of the spatial filters or classifier towards a good solution even with few training data. For instance, inverse solutions or beamforming have been used to specify prior regions of interest from which relevant features are likely to be found, hence making BCI calibration possible with fewer training trials [53][54]. Similarly, one can use as prior information which channels are the most likely to be useful and use that as a regularizer to optimize spatial filters [55].

2.3 Calibration time suppression

In order to completely suppress BCI calibration time, it is necessary to build a user-independent BCI system, i.e., to have features

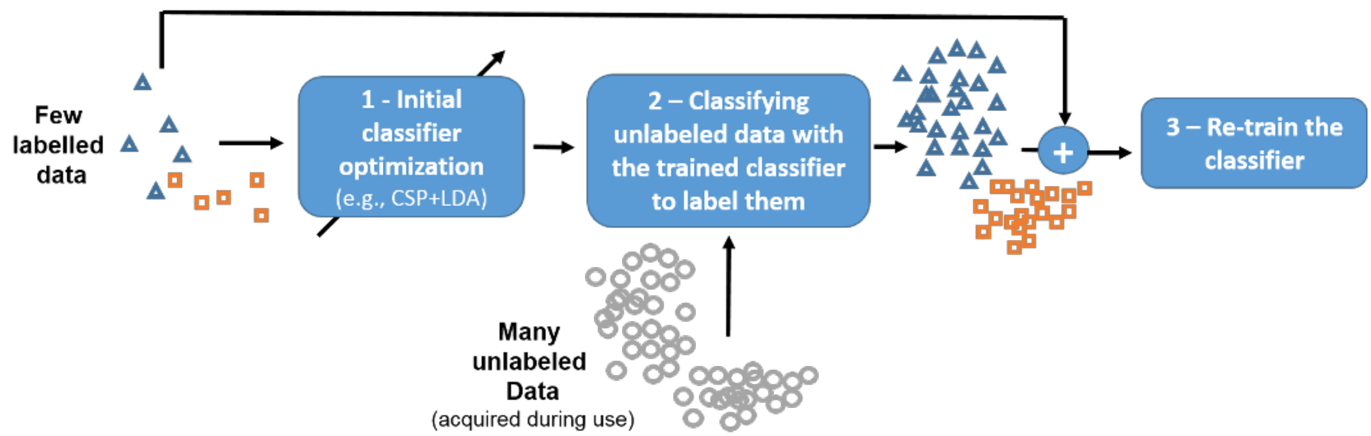

Fig. 4. Principle of semi-supervised learning: 1) a model (e.g., CSP+LDA classifier) is first trained on the few available labelled training data. 2) This model is then used to classify and thus label the many unlabeled data (the test data) available. 3) The newly labelled data are combined with the originally available labelled ones to retrain the model with many more data, and thus hopefully to obtain a better model.

and classifiers that work well across users. This is a difficult challenge due to the large between-user variability in oscillatory activity-based BCI. So far, it has been addressed using two main approaches: 1) pooled designs, i.e., calibrating a BCI on the pooled data (possibly transformed) from multiple users, and 2) ensemble designs, in which user-specific BCI are combined together to create a user-independent one.

2.3.1 Pooled design

A straightforward approach to user-independent BCI design is to optimize features (typically CSP spatial filters) and the classifier on the combined data from multiple users [56] (see Figure 5, top). Note that it may be necessary to use feature parameters (e.g., frequency band width) that are broad enough to be relevant for multiple users [56]. Interestingly enough, some authors suggested that when designing such user-independent BCI, using gender-specific BCI (e.g., a BCI calibrated on EEG signals from female users only for a target female user) may improve performances [57]. To address the between-user variability, a spatial normalization scheme, based on unlabeled data, can be used to make the data of different users spatially similar, and thus improve the performance of a user-independent BCI system optimized on data from multiple users [58]. In [59], a clustering approach was used to suppress re-calibration for a user for which data from previous days were available. CSP filters optimized on this user's data from previous days were clustered in order to identify filters that were robust across days and could thus be re-used for a new day without recalibration. To the best of our knowledge, this approach was not used to completely suppress calibration for a completely new user though.

2.3.2 Ensemble design

More advanced and efficient approaches consist in using ensemble methods. With ensemble methods, one can learn a BCI model (typically CSP+LDA) for each one of the users available, and then combine them to be efficient across users, as in [60] (see Figure 5, bottom). More recently, the same group explored $l1$-penalized linear mixed-effect models to identify and separate within-user variability from between-user variability, hence extracting a more efficient user-independent model [61]. Alternatively, among the different models learnt for each user, the most relevant ones to use to classify the data from the target user (not part of the training set) can be dynamically selected using a meta-classifier that has been trained to identify whether or not a given

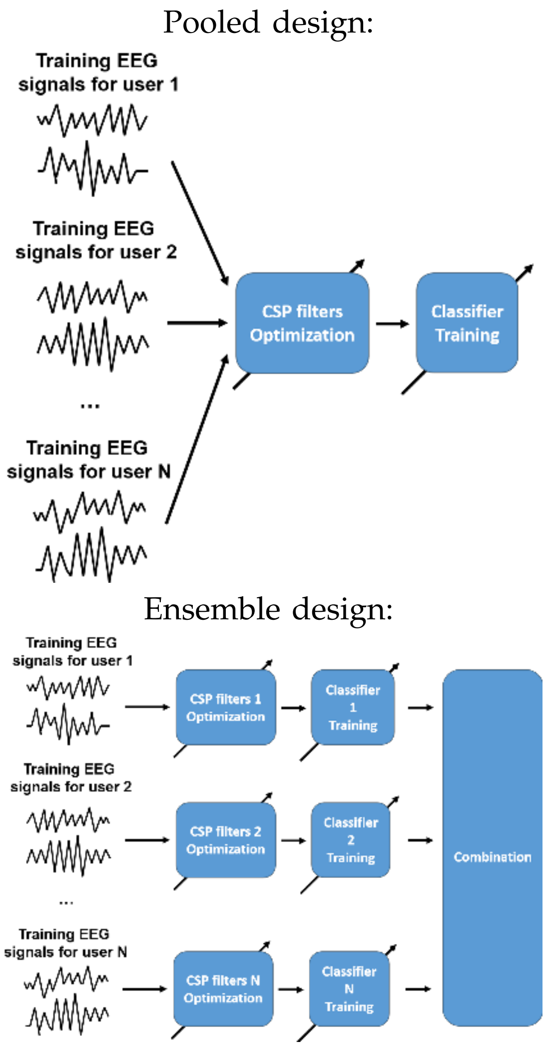

Fig. 5. Two approaches to user-independent BCI design for calibration suppression: Top: pooled design optimizing a single model (CSP+classifier) on the combined data of multiple users. Bottom: ensemble design, for which a model is optimized for each user and these models are then combined. These two approaches are assessed in Section 4

classifier is relevant to classify the target data [62].

3 New methods

To further contribute to reducing BCI calibration time, we propose here a couple of new approaches. In particular we propose a new type of method to reduce calibration time based on Artificial Data Generation (ADG). The idea is to generate multiple trials from the few training trials available in order to increase the training set size. We also propose a new and simple user-to-user transfer approach. These new methods are described herafter.

3.1 Artificial EEG data generation

The problematic need for large amounts of training data is not unique to the BCI field, and can also be encountered in other fields in which machine learning is involved, although the problem might be more severe for BCI. In these fields, some authors proposed to deal with this issue by generating numerous artificial training data from the few original training data available, and use it to augment the training set. This has been shown to lead to increased classification accuracies in fields such as speech processing [63] or hand-writing recognition [64]. We therefore propose here to explore this idea for BCI design. In particular, we propose three ways to generate artificial EEG trials for BCI: 1) using signal segmentation and recombination in the time domain, 2) using signal segmentation and recombination in the time-frequency domain and 3) using analogies. Note that all three methods are applied on the already band-pass filtered EEG signals.

3.1.1 Signal segmentation and recombination in the time domain:

The idea of this first simple approach to ADG[3] is to first divide each training EEG trial into several segments, and then generate new artificial trials as a concatenation of segments coming from different and randomly selected training trials from the same class. More formally, let us denote as $\Omega = \{T_i\}, i \in [1, N]$ as the set of N band-pass filtered EEG trials that are available for training for a given class (note that this ADG is performed separately for each class), $T_i \in \Re^{C \times S}$, with S the number of samples in a trial, and C the number of channels. The first step consists in dividing the signals (from each channel) of each training trial T_i into K consecutive and non-overlapping segments $T_i^k \in \Re^{C \times S/K}, k \in [1, K]$ (each segment containing the same number of EEG samples). Then, from these segments, we can generate a

3. We already presented preliminary results with this approach in a short conference paper [65].

new artificial trial $\tilde{T}_j$ as $\tilde{T}_j = [T^1_{R_1} T^2_{R_2} \dots T^K_{R_K}]$, where $[AB]$ denote the concatenation of the samples from segment A and B (in other words a concatenation of the columns, i.e., along the time dimension), and R_k is a randomly selected integer (random selection with replacement) from $[1, N]$. The whole process is schematized in Figure 6. This simple approach enables us to generate a large number of new trials, different from the original ones, but still relevant and likely to be similar to future trials, as they were made from parts of real trials and have the same temporal structure. Therefore, each new artificial trial will have a spatial covariance matrix different from that of the original trials, but with covariance values (and thus signal power - which is the only information used by the CSP and LDA) that are within a credible range for the current user as they were computed from data of this very same user. It will thus lead to a different average covariance matrix $\tilde{C}_i$ for CSP optimization, that will be in fact regularized towards the average covariance matrix of the artificial trials, as the estimation of this average covariance matrix can be rewritten as follows:

$$\tilde{C}_i = \frac{N_{orig}}{N_{orig} + N_{art}} C_i^{orig} + \frac{N_{art}}{N_{orig} + N_{art}} C_i^{art} \quad (7)$$

where C_i^{orig} is the average spatial covariance matrix for class i estimated from the original trials, C_i^{art} is the average spatial covariance matrix estimated from the artificial trials, and N_{orig} and N_{art} are the number of original and artificial trials, respectively. Similarly, the feature covariance matrix of LDA will take into account these new trials, and thus the resulting LDA classifier will be able to deal with this increased variability better.

3.1.2 Signal segmentation and recombination in the time-frequency domain:

While the previous approach is extremely simple, and yet effective (see Section 4), it is also brutal. Indeed, simply concatenating segments from different trials may result in inadequate matching between the EEG samples at the boundary between two consecutive segments and thus add some unwanted high frequency noise. To avoid this issue, it could be useful to perform the trial segmentation and random recombinations into the time-frequency domain rather than directly in the time domain. To do so, we first transform each band-pass filtered training trial T_i in a time-frequency representation TF_i by using a Short-Time Fourier Transform (STFT) for each channel (to do so we used 250ms-long hamming windows with 50% overlap). We denote as TF_i^k the k^{th} time window (containing a Fourier spectrum for each channel) of trial T_i in the time-frequency domain. Then, from these time windows, we can generate a new artificial trial in the time-frequency domain $\tilde{TF}_j$ as $\tilde{TF}_j = [TF^1_{R_1} TF^2_{R_2} \dots TF^K_{R_K}]$, i.e., by concatenating together STFT windows from different trials from the same class. The final artificial trial $\tilde{T}_j$ is obtained by using inverse STFT on $\tilde{TF}_j$. This process, illustrated in Figure 7, is repeated multiple times to generate multiple artificial trials.

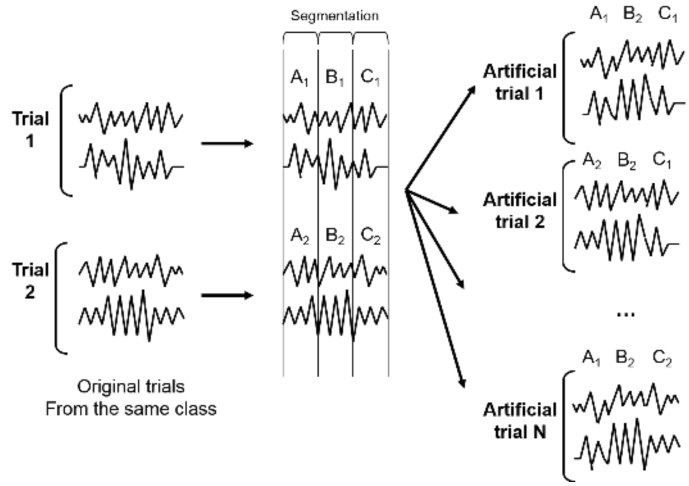

Fig. 6. Principle of artificial EEG data generation in the time domain

3.1.3 Artificial trial generation based on analogy:

The last approach we propose for ADG is based on analogies. The idea of analogy-based data generation, explored in [66] for hand-writing recognition, consists in first taking 3 data examples A, B and C, and in generating a 4^{th} example D which is to C what B is to A. In other words, we compute the analogy between A and B, and create a data D which has the same analogy to C. This is a way to generate

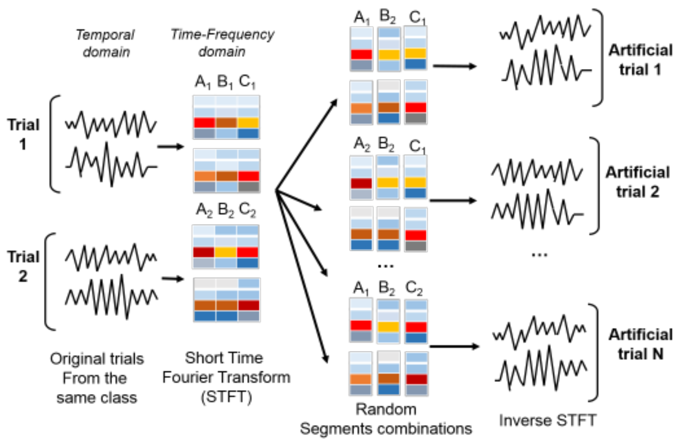

Fig. 7. Principle of artificial EEG data generation in the time-frequency domain

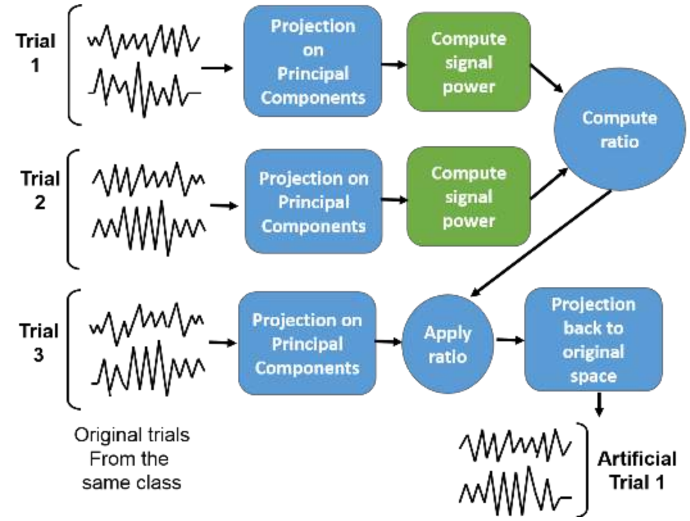

Fig. 8. Principle of artificial EEG data generation based on Analogies

artificial trials that are different from the existing ones but whose difference is relevant since it can already be found between other available trials. To apply this principle in practice for EEG signals, we used the following approach, for each class:

1) compute the covariance matrix C of all the available data for this class (using shrinkage [32] for a better estimate)
2) compute the eigen vectors V of this matrix ($C = VDV^T$ with D the diagonal matrix of eigen values), i.e., the Principal Components (PC) of the data [67]
3) randomly select 3 distinct trials X_A, X_B and X_C
4) project the first two on the PC, i.e., X_AV and X_BV, and compute the signal power p^i_A and p^i_B along each PC V^i (V^i being the i^{th} column of V)
5) create artificial trial X_D by transforming trial X_C as $X_CVdiag(p_A^{-1/2})diag(p_B^{1/2})V^T$, where $diag(p)$ is a purely diagonal matrix whose diagonal elements are those of vector p. This amounts to creating a trial D whose signal power along the PC is as different from those of trial C, as the signal power along the PC of trial B is different from those of trial A.
6) return to step 3 and repeat the process to generate more artificial trials.

In other words this approach consist in computing a transformation to make trial A similar to trial B (here similar in terms of PC signal power) and then applying this transformation to trial C to create artificial trial D. Note that we chose here to work with power along the PC because the signal power is what is used by CSP and LDA for classifying the EEG signals. Note also that here we expect PC that do not correspond to discriminant directions (i.e., the noise) to be more variable that those corresponding to discriminant direction (i.e., the signal). As such our analogy approach will mostly generate new trials with various levels of noise, thus helping the subsequent classifier to deal with such noise. Naturally, other transformations than the one used here could and should be explored in the future. This analogy-based ADG is illustrated in Figure 8.

3.2 Combining user-to-user transfer, ensemble approaches and Riemannian geometry for calibration time reduction

In order to perform a simple, computationally fast and efficient user-to-user transfer for calibration time reduction, we propose to re-use the idea of multi-users covariance matrix estimation (see Section 2.2.2), that we and others explored in [33][35][36]. However, rather than using all other users together as in [35] or weighting or selecting them using heuristics as in [33][36], we propose to use a sound theoretical framework to assess which other users' data should be used based on their similarity to the data of the target user. In particular, we propose to use Riemannian geometry [68], which recently proved very promising for BCI design [69], to measure how different the average covariance matrices of the other users are from that of the target user. Indeed, Riemannian geometry has been specifically designed to manipulate and measure distances between Symmetric and Positive Definite (SPD) matrices, which covariance matrices are. Formally, the Riemannian distance δ_R between two SPD matrices A and B is the following:

$$\delta_R(A,B) = \| \log(A^{-1}B)\|_F = [\sum_{i=1}^{n} \log(\lambda_i)^2]^{1/2} \tag{8}$$

where the λ_i are the eigen values of $A^{-1}B$. Using this distance, we can easily and quickly identify which other users have covariance matrices that are close to that of the target user and thus use them as regularizers. More precisely, we propose to perform user-to-user transfer by regularizing the target user covariance matrices (both CSP and LDA covariance matrices as we did in [33]) as follows:

$$\hat{C_{target}} = \lambda C_{target} + (1-\lambda)\sum_i \frac{1}{\gamma_i} C_{s_i} \tag{9}$$

with

$$\gamma_i = \frac{\delta_R(C_{target}, C_{s_i})}{\sum_j \delta_R(C_{target}, C_{s_j})} \tag{10}$$

where C_{target} is a covariance matrix estimated for the target user and C_{s_i} the covariance matrix estimated for the i^{th} other user. In other words, the Riemannian distance enables us to emphasize covariance matrices that are close to that of the target user in the regularization term, and de-emphasize those that are too different. These regularized covariance matrices are then plugged into the CSP (spatial covariance matrices) and LDA (feature covariance matrix) algorithms to perform user-to-user transfer. Naturally, this would only work if the target user's data available do contain some class-related information. If they do not (e.g., if the user cannot produce the desired EEG patterns, or if the covariance estimate is too poorly estimated, for instance due to outliers), then regularizing towards the closest covariance matrices from other users may not help. This approach therefore helps by 1) regularizing the covariance matrix towards well estimated covariance matrices, and thus performing some implicit shrinkage and 2) by adding variability (i.e., noise) that is likely to occur (as it was observed in other users) to the data, to help the CSP and LDA to deal with it. To avoid the selection of the regularization parameter λ (which is difficult when few training data is available - thus preventing cross-validation use), we re-use the trick introduced in [35] and optimize several CSP and LDA pairs, one for each value of λ among $[0.1, 0.2, \ldots, 0.9]$. When classifying a new trial, these different CSP+LDA pairs are combined by summing the LDA outputs (signed distance of the feature vector from the LDA separating hyperplane) from each of them to determine the final class. This results in a simple, fast, theoretically sound and parameter free multi-user BCI design.

Overall, a number of methods have been proposed to reduce or suppress calibration time in oscillatory activity-based BCI. The different methods categories and sub-categories are summarized in Table 1, together with the new methods proposed in this paper. This table also indicates which references correspond to which method category, as well as whether a given family of method is evaluated in this paper (see Section 4).

4 Evaluations and analyses

In order to study and compare different approaches to BCI calibration time reduction or suppression we analyzed the performance of various methods on EEG data from 50 users, from 3 data sets, for different number of training trials, offline. We aimed at finding out how few training data were necessary to achieve a desired level of performance, and thus how long the calibration time would be for different methods. The data sets used, the methods studied and the evaluation results are described in the following sections.

TABLE 1
Overview of the different methods used to reduce or suppress calibration time in BCI.

Method type	Sub-category	References	Evaluated here
Calibration reduction			
Regularization approaches	Shrinkage	[33]	X
	Divide-and-conquer	[34]	
User-to-user transfer	Multi-users covariance matrix estimation	[35], [36], [33]	X
	Multi-task learning	[38], [39], [40], [41]	
	Domain adaptation	[43], [44], [45], [42]	
Semi-supervised learning		[49], [50], [51], [52]	X
A priori physiological information		[53], [54], [55]	
Artificial data generation		[65]	X
Calibration suppression			
Pooled design		[56], [57], [58], [59]	X
Ensemble design		[60], [61], [62]	X

4.1 EEG Data Sets

4.1.1 Data set 1, Motor Imagery data:

The first data set used is data set 2a from BCI competition IV, provided by the Graz group [70]. This data set comprises EEG signals from $N_1 = 9$ users who performed left hand, right hand, foot and tongue Motor Imagery (MI). Users were instructed to perform a given motor imagery task following the appearance of an arrow pointing left, right, up or down, and to do so repeateadly for 4 seconds. The EEG signals were recorded using 22 EEG channels located over the motor cortex. For the purpose of this study, only EEG signals corresponding to left and right hand MI were used. EEG signals were band-pass filtered in the 8-30 Hz frequency band. Features were extracted from the 2-second time window starting 0.5 s after the cue. 72 trials per class were available for both training and testing, as provided for the competition.

4.1.2 Data set 2, Workload data:

This data set is an in-house data set, recorded while $N_2 = 21$ users performed two tasks involving different mental workload levels (easy tasks vs difficult tasks). The cognitive difficulty of the task was manipulated using the N-back task. With this task, users saw a sequence of letters on screen, the letters being displayed one by one, every 2 seconds. For each letter the user had to indicate with a mouse click whether the displayed letter was the same one as the letter displayed N letters before. Each user participated into 12 blocks, alternating between *easy* blocks with the 0-back task (the user had to identify whether the current letter was the letter 'X') and *difficult* block with the 2-back task (the user had to identify whether the current letter was the same letter as the one displayed 2 letters before). Each block contained 30 letter presentations. EEG signals were recorded using 30 EEG channels. This data set is described in more details in [27][71]. We used each 2-second long time window of EEG data immediately following a letter presentation as a trial, as in [27]. The first 6 blocks were used as the training set (180 trials per class) and the last 6 blocks as the testing set (180 trials per class as well). All EEG signals were band-pass filtered in 8-12Hz as this band was identified as the most useful one for mental workload discrimination in [27]. For such workload classification, the relevant discriminative activity is expected to originate mainly from the occipital area.

4.1.3 Data set 3, Mental Imagery data:

This last data set is another in-house data set which comprises EEG data from $N_3 = 20$ users who performed mental imagery tasks [72]. More precisely, following a protocol similar to the one used for data set 1, users were prompted to perform either left hand motor imagery, mental rotation of a geometric figure (the figure being displayed on screen) or mental subtractions (successive subtraction of a two digits number from a three digits number, both numbers being displayed on screen) following the appearance of an arrow pointing left, right or up respectively. The arrow was pointing towards a picture representing the mental imagery task to be performed. Users had to perform the instructed mental imagery task for 4.25 seconds starting from the arrow appearance. Each user participated to 5 runs, each run containing 15 trials per mental imagery task, presented in a random order. For the purpose of this study, only EEG signals corresponding to left motor imagery and mental rotation of a geometric figure were used. EEG signals were band-pass filtered in the 8-30 Hz frequency band, as in [73]. For each user, the first two runs were used as the training set (i.e., 30 trials per class) and the last three runs as the testing set (i.e., 45 trials per class). As for data set 1, features were extracted in a 2-second long time window starting 0.5s after the cue.

4.2 Methods analysed and compared

In order to identify and understand the signal processing tools that can effectively reduce or suppress BCI calibration times, we studied and compared several of them for different number of training data. In particular we studied:

Baseline: We used the standard BCI design, described in Section 2.1, which simply consists in training CSP filters and a LDA classifier on the available training trials.

Regularization approach with Shrinkage: We studied the performance of the automatic covariance matrix shrinkage approach described in Section 2.2.1, for both CSP and LDA. Note that for CSP, we used it to estimate the covariance matrices of each trial in equation 2.

User-to-User transfer: We studied the Multi-User BCI design approach that we proposed in Section 3.2. With this approach, we used $\lambda \in [0.1, 0.2, \ldots, 0.9]$ in equation 9.

Semi-supervised learning: We studied semi-supervised learning (see Section 2.2.3). To do so we straightforwardly applied the semi-supervised learning principle to a BCI design based on CSP+LDA. First, we trained the CSP filters and LDA classifier on the available training trials. Then, we used them to classify the first 5 trials from the testing set. We labelled these 5 trials with the estimated class given by the CSP+LDA, added them to the training trials, and retrained the CSP+LDA on this newly augmented training set. We repeated the process for each subsequent block of 5 testing trials until the whole testing set was classified.

User-independent design: For user-independent BCI design we studied:

- A *pooled user-independent design*, studied in [56], which simply consists in pooling together the EEG data of all available users, and then training CSP filters and a LDA classifier on them. We studied this user-independent design both with and without automatic covariance matrix shrinkage (for both CSP+LDA).
- An *ensemble approach*, similar to that of [60], which consists in first training a set of CSP filters and a LDA classifier on the data of each available user separately, and then combining the outputs of this ensemble of CSP+LDA to classify the data of a new unseen user. This is achieved by simply using each CSP+LDA for each training user on the new unseen trial, then by concatenating the LDA outputs (signed distance to the LDA hyperplane) together and used them as input to an higher level LDA (also trained previously) which takes the final decision. Automatic covariance matrix shrinkage was used for both CSP and LDA with this approach.

Artificial data generation: For ADG, we studied all three approaches newly proposed in Section 3.1, i.e., ADG in the time domain, in the time-frequency domain and based on analogies. For each method, we generated 100 artificial trials[4] from the available training trials and added them to this initial training set before training the CSP+LDA. For ADG in the time and time-frequency domains, the 2 second long EEG trials were divided into 8 segments before being recombined into artificial trials. Due to the random process involved (random trial selection from which generating a new artificial trial), this process was repeated 10

4. According to our preliminary study in [65], using more than 100 artificial trials does not lead to further significant performance improvements.

times and the performance results obtained averaged over the 10 repetitions.

For each method, we used 6 filters for CSP (corresponding to the 3 largest and 3 smallest eigen values of the GEVD problem). We studied the performance of each calibration time reduction method for different numbers of training trials, starting from only the first 5 training trials per class, to all available training trials, by step of 5 training trials per class. For the calibration time suppression approaches, for each data set, the user-independent BCI were trained on the training EEG trials from all available users expect one, and tested on the testing set of this remaining user. This process was repeated so that each user is used once as the testing set. In other words we used a leave-one-user-out cross-validation scheme. The reported classification performance (percentage of trials whose class was correctly estimated) are those obtained on the testing sets of each user, for which all available trials were used.

4.3 Results

Figure 9 first displays the average classification performances (averaged over users) obtained by the standard design (Baseline) and the different ADG approaches. Results first suggest, as expected, that for the standard design, for all data sets, the less trials used for training, the lower the performances. In particular for small training sets, typically with less than 20 trials per class, the performances are very low, near or at chance level (50%), and decrease dramatically when the number of training trials decreases. This confirms the need for numerous training trials for each user, and thus the resulting long calibration time. Then, what can be observed is that ADG approaches often increase classification performances, particularly when few training data is available. These differences are statistically significant for data set 3 (paired t-test[5], $p < 0.01$) for both ADG in the time domain and ADG based on analogy, and show a trend for ADG in the time-frequency domain ($p = 0.07$). On Data set 2, both ADG based on analogy and ADG in the time-frequency domain are significantly better than the baseline ($p < 0.05$). Overall, ADG in the time-frequency domain is the best method, significantly better than ADG based on analogy ($p < 0.01$), and on average better than ADG in the time-domain, although not significantly so. Overall, these results indeed support that ADG methods can be used to reduce BCI calibration time, since they can achieve a given performance with less training data than the baseline approach.

Then, Figure 10 displays the results obtained by the baseline and the different user-independent BCIs. As could be expected, the User-Independent (UI) methods have lower performances than the user-specific methods when all available training data are used. However, when very few training data are available (i.e., less than 10-15 trials per class), then UI methods often outperform the baseline on average. When comparing the different UI methods, no clear picture emerges. The pooled design reached reasonnable, better-than-chance performance, on Data sets 1 and 3, but completely failed (chance level) on Data set 2. The ensemble approach is better than the pooled approach on Data set 2, but poorer on the other two data sets. Overall, there is no significant differences between the three UI designs.

Finally, Figure 11 compares the methods from each category, evaluating only the best one from a given category if this category contained several methods. In particular for ADG we compared only the method based on time-frequency information to other categories since this was the most efficient ADG method. In the following we denote this method as ADG-TF. For ADG-TF, we also used automatic covariance matrix shrinkage for CSP+LDA, since the two methods can be combined. For the UI methods, we studied the performance of the pooled design with shrinkage. Overall, we compared the baseline design, automatic covariance matrix shrinkage, ADG-TF with and without shrinkage, semi-supervised learning, multi-user design and UI design based on pooled data with shrinkage.

First, results showed that automatic covariance matrix shrinkage significantly outperformed the baseline ($p << 0.01$), on all data

5. The paired t-test was applied here on the average performance of each user, i.e., averaged across the different numbers of training trials used for each user.

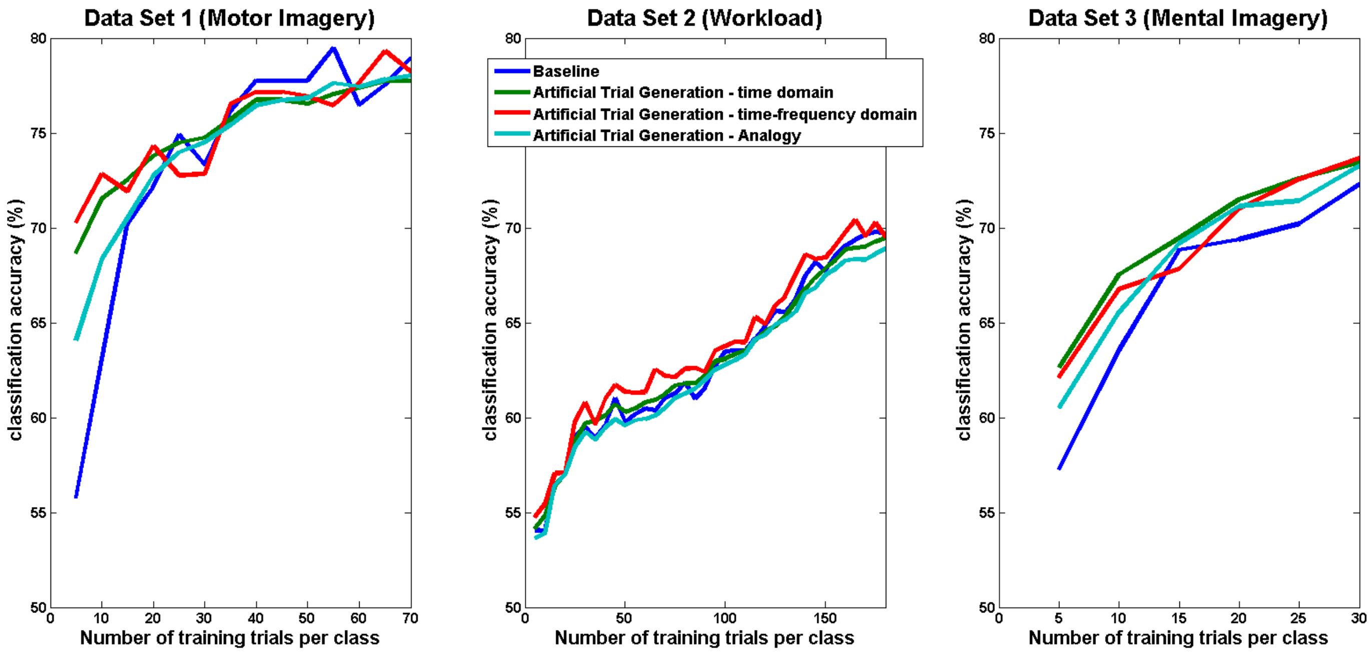

Fig. 9. Average classification performances (over users) for the artificial data generation (ADG) methods (i.e., ADG in the time domain, ADG in the time-frequency domain and ADG based on analogy) and the baseline. Standard deviation bars are omitted for clarity. See text for statistical analyses.

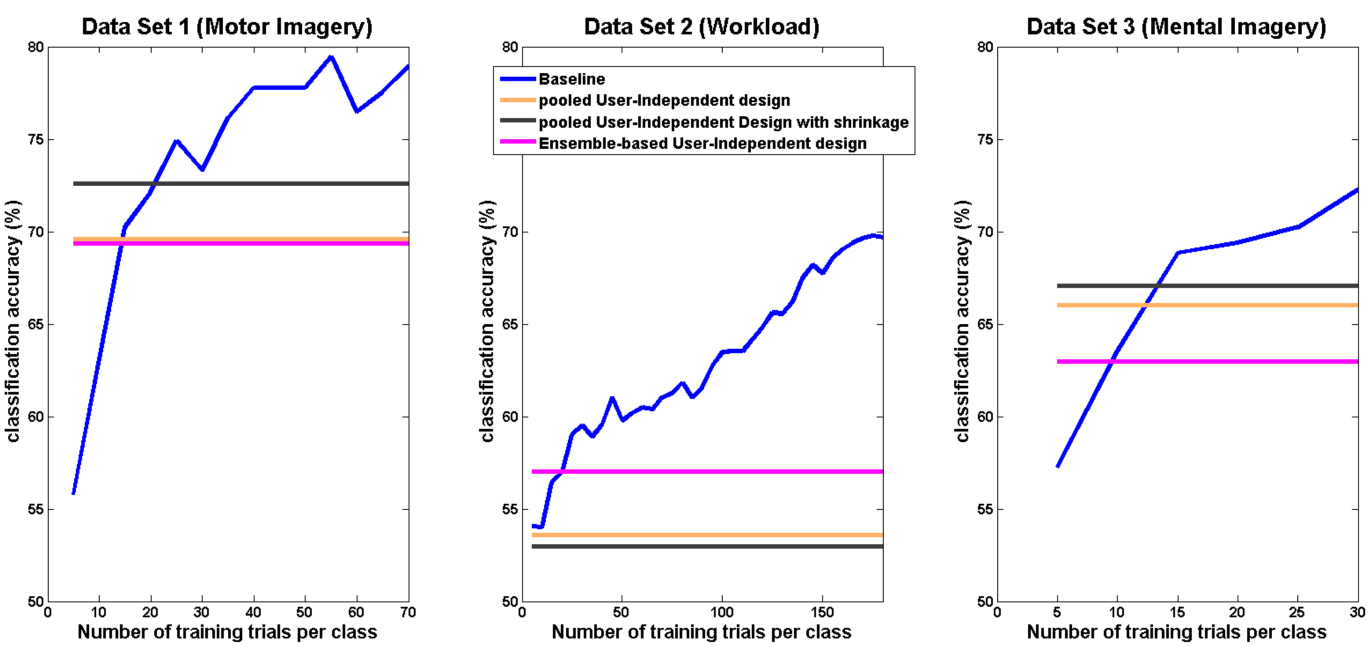

Fig. 10. Average classification performances (over users) for the user-independent (UI) BCI design methods (i.e., pooled UI design, pooled UI with shrinkage and ensemble UI design) and the baseline. Standard deviation bars are omitted for clarity. See text for statistical analyses.

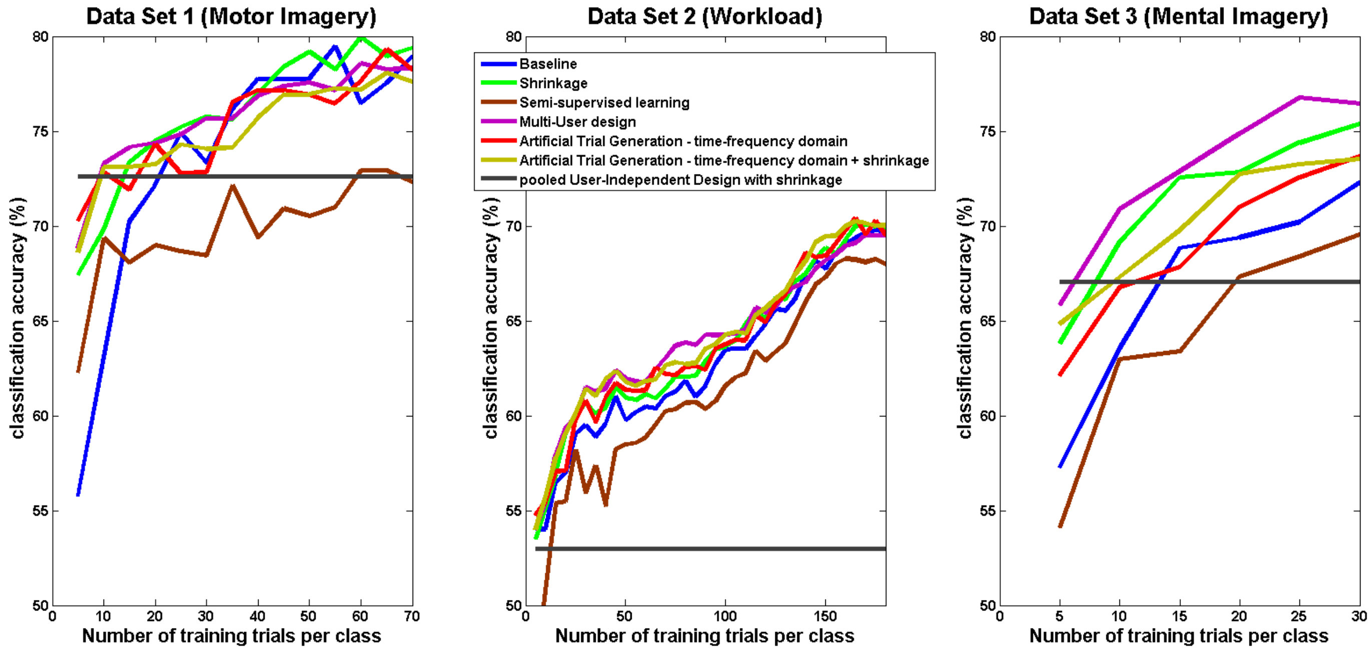

Fig. 11. Average classification performances (over users) for the different methods from each category (i.e., shrinkage, semi-supervised learning, multi-user design, ADG-TF, ADG-TF with shrinkage, pooled UI design with shrinkage) and the baseline. Standard deviation bars are omitted for clarity. See text for statistical analyses.

sets. This increase of classification performance is particularly clear when the number of training trials is very low, but is still present even when the maximum number of training trials is used. When comparing ADG-TF and shrinkage, there were no significant differences between the two methods on data set 1 and 2, but shrinkage was significantly better than ADG-TF on data set 3 ($p < 0.05$). However, combining ADG-TF with shrinkage makes the performance of ADG-TF better. Indeed, ADG-TF+shrinkage outperformed simple shrinkage on data set 2 ($p < 0.05$), whereas there was no significant differences between ADG-TF+shrinkage and shrinkage on the two other data sets. This suggests that shrinkage and ADG-TF regularize the covariance matrices in different ways, and thus might be complementary. This also suggests that ADG indeed seems to had noise (and/or variability) to the training set, making the CSP/LDA more robust to deal with this noise/variability. Semi-Supervised Learning (SSL) applied to CSP and LDA, on the other hand, essentially failed. Indeed, it was poorer than the baseline most of the time, except sometimes with very few training data. The fact that CSP and LDA are sensitive to noise and mislabels can explain this failure, since mislabels are bound to occur using SSL. A more robust approach to SSL should probably be used, e.g., using an SVM instead of an LDA as in [51]. The proposed Multi-User BCI (MU-BCI) design also proved very efficient. Even when the maximum number of training trials is used, this approach can substantially improve performances, suggesting that it is not only useful for calibration time reduction but also for performance improvement in general. On average over all data sets, MU-BCI notably significantly outperformed all other methods ($p < 0.05$), except shrinkage which was significantly outperformed by MU-BCI only on data set 3 ($p < 0.05$). Overall, the best three methods for calibration time reduction are shrinkage, ADG-TF with shrinkage, and MU-BCI. Regarding UI-BCI designs, they are most of the time clearly outperformed by the other approaches, except when no training trial is available (obviously) or only 5 training trials per class. For 10 trials per class onwards, the different methods studied (except the baseline) are more efficient than the UI designs. This suggests that such UI-BCI could be used to provide feedback to the user for the first 5 trials per class, and that a calibration time reduction method should be used once these 5 trials per class acquired. Overall, it is interesting to notice that with only 10 trials per class, several calibration time reduction methods can reach a classification performance equivalent to that of the baseline with 30 to 40 trials per class, hence effectively reducing the calibration time by 3 or 4. When only 5 trials per class are available, all calibration time reduction approaches except SSL can increase classification performance by 10 to 15%, e.g., from 55% to 70% in Data set 1, or from 55% to 65% in Data set 3.

5 Further Analyses

In order to gain more insights into the obtained results, we conducted some additional experiments. First, we explored what was the impact of the lack of training data on CSP and LDA separately. Then, we present a case study about what makes ADG work. We also studied the impact of the segment size on the performances of ADG. Finally, we provide a glimpse on how different methods impact the performances of different users.

5.1 Impact of training data scarcity on CSP and LDA separately

As seen in the evaluation results so far, and as could be theoretically expected, the fewer the training data available, the lower the classification performances of the BCI. This is due to both the CSP and LDA being optimized on these training data, and therefore being poorly designed when too few data are available. However, are CSP and LDA affected by training data scarcity in the same way? To find out, we studied the performance of the BCI, when only CSP, only LDA or both are faced with training data scarcity, for different number of training data, on data set 3. We also performed the same analysis when using shrinkage for LDA or for CSP, in order to assess the impact of regularization on the capability of these two algorithms to deal with training data scarcity. The obtained results are displayed in Figure 12.

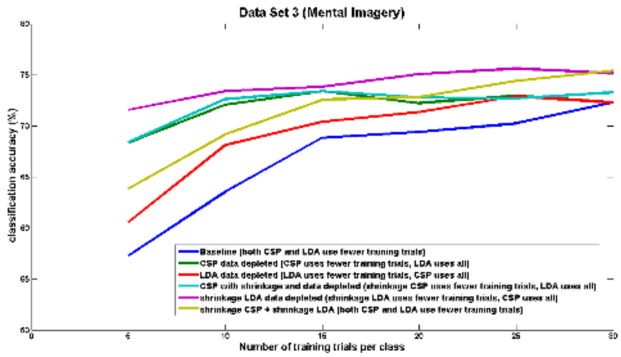

Fig. 12. Classification performances obtained on data set 3, when 1) only CSP uses fewer training trials (but LDA uses all training trials available), 2) only LDA uses few trials (but CSP uses all), 3) both CSP and LDA uses fewer training trials, for different numbers of training trials used, 4) CSP is regularized with shrinkage and uses fewer trials (but regular LDA uses all training trials available), 5) LDA is regularized with shrinkage and uses fewer trials (but regular CSP uses all training trials available) and 6) both CSP and LDA use regularization with shrinkage and use fewer training trials, for different numbers of training trials used.

As can be seen on this figure, both CSP and LDA do suffer from training data scarcity, as their performances decrease with the number of available training data. However, it is interesting to observe that LDA appears to be more sensitive to training data scarcity than CSP. Indeed, when only CSP uses fewer training data, the performances of the resulting BCI are clearly higher than when only LDA uses fewer training data. When using regularization with shrinkage, both CSP and LDA performances improve, although LDA benefits much more than CSP from regularization. This further confirms that LDA is the most sensitive algorithm to training data scarcity. This may be explained by the fact that the covariance matrices used in CSP are defined as the average covariance matrices over training trials. As such adding training trials essentially improves the averaging process, but does not involve representing the variability of the covariance matrices. In other words, CSP does not consider the variability over trials, but only the average trial (represented by the average spatial covariance matrix). As for LDA, the covariance matrix used is the covariance of the training feature vectors, as such adding training trials improves the estimation of this covariance matrix, which thus involves representing the variability of these training feature vectors. Moreover, although the input dimensionality of CSP is higher than that of LDA, the CSP covariance matrices are estimated from hundreds of EEG samples per trial, whereas the covariance of LDA is estimated only from one example per trial. Overall, this could explain why LDA benefits more than CSP from increased training data, and is therefore the most sensitive algorithm to data scarcity.

5.2 A case study of artificial data generation impact

In this paper, we have proposed to generate artificial training EEG data to reduce BCI calibration time. In this section, we present a case study to gain more insights about why it does work and how these artificial trials impact the CSP and LDA estimation. To do so, we looked at the CSP filters, features and LDA classifier obtained with or without artificial data generation, for user number 3 from data set 1, with 5 original training trials per class. Indeed, with only the original trials, this user reached a classification accuracy on the test set of 57.6%, whereas it reached 94.4% when using artificial data generation in the time domain. It is therefore interesting to study the reasons of this sharp performance increase. Figure 13 displays the CSP spatial filters obtained with and without artificial data generation. It can be observed that the CSP filters are actually very similar in the two conditions: ADG had little impact on CSP optimization.

The resulting CSP feature vectors (containing 6 CSP features) extracted from the training set can be visualized in Figure 14. Here we can see that ADG indeed populates the feature space in a plausible way, which seems to respect the structure of the original trials. ADG also increases the variability of the training data as compared to that of the original data. This is expected to help the subsequent classifier to learn to cope with this variability and thus to use the most invariant features.

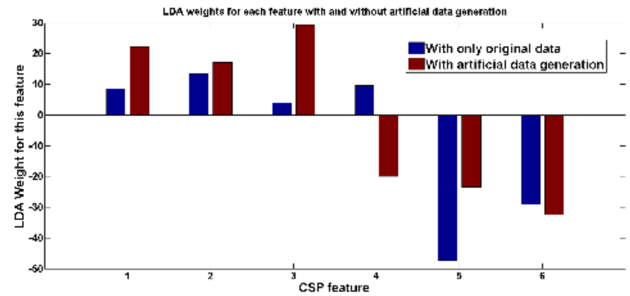

Fig. 15. Weights attributed to each of the 6 CSP features by the LDA classifier with or without artificial data generation, for user number 3 from data set 1, with 5 original training trials per class. It can be observed that the weights are very different between the two conditions.

The weights attributed to each CSP feature by the LDA classifier, with or without ADG, are displayed in Figure 15. It can be seen that the feature weights are rather different when using ADG or not. Feature 4 weight has even a different sign in the two conditions. Given that the CSP filters are almost identical in the two conditions, the LDA differences between the two conditions are the one which explain the sharp increased in performance obtained when using ADG. This is also consistent with the study in Section 5.1 in which we found out that LDA was much more sensitive than CSP to data scarcity.

Overall it seems that ADG can improve BCI design with few training data mostly by improving the estimation of the LDA covariance matrix and increasing the training data variability to make the LDA classifier more robust to such variability.

5.3 Impact of the segment size on artificial data generation

Another interesting question regarding ADG, is how the number of segments, used to segment trials before recombining them into new artificial trials, impacts the performances obtained. To study this point, we computed the performances of ADG in the time and time-frequency

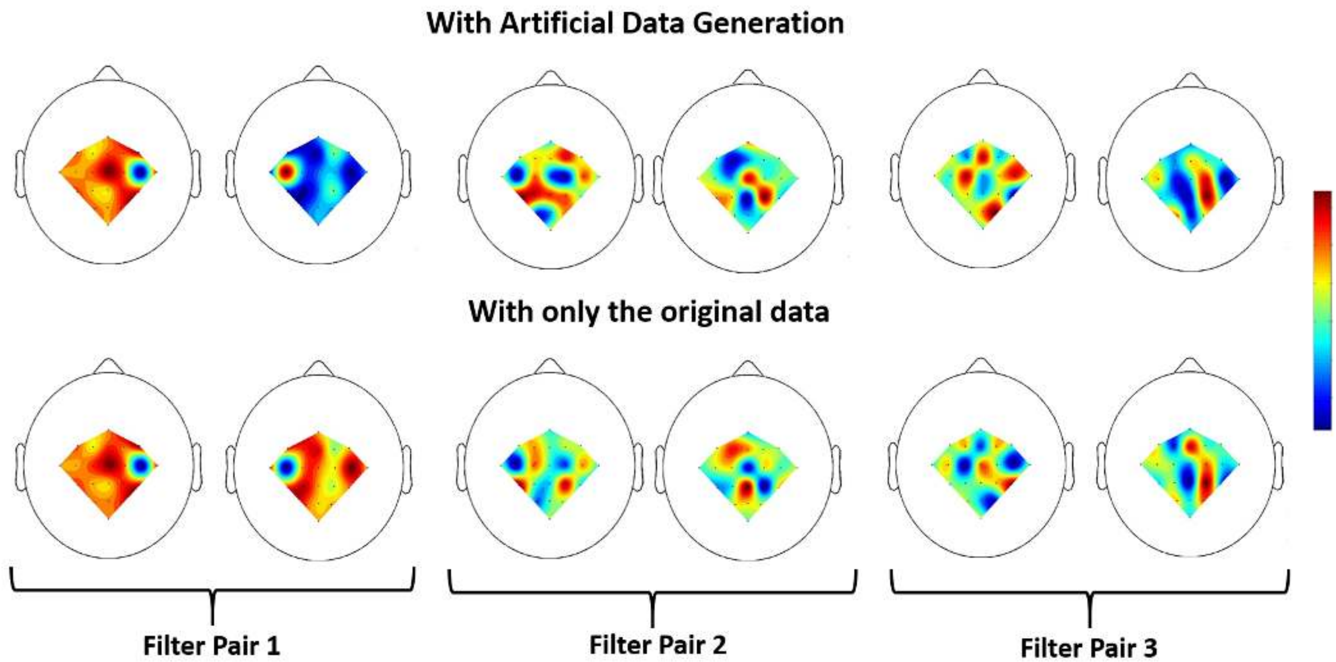

Fig. 13. CSP Spatial Filters (i.e., weights attributed to each channel) obtained with and without artificial data generation for user number 3 from data set 1, with 5 original training trials per class. It can be observed that these filters are very similar between the two conditions. (Note: the sign of the CSP filter is arbitrary, as filters w and $-w$ would give exactely the same features.)

domain, on data set 3, for a 4, 8 or 12 segments. The results can be seen in Figure 16

It can be observed that the higher the number of segments, i.e., the finer the segmentation, the better the performances. However, the differences between the performances obtained for 8 and 12 segments are very small and thus not substantial. In practice, it therefore makes sense to use more than 4 segments, but 8 or 12 segments would give similar performances.

5.4 Individual variations between users

In order to obtain a glimpse about how different methods impact different types of users, we looked at the individually obtained performances for the 20 users of data set 3, for 10 original training trials per class. Indeed, for this number of training trials, many of the calibration reduction methods lead to a substantial improvement. The obtained results are displayed on Figure 17.

What the results suggest is that for users with poor performances with the baseline design (e.g., users 7, 8, 17, 19, 20), most of them can increase substantially their performances by using a user-independent BCI design. For users with already high performances with the basic design (e.g., users 1, 3, 6, 12), they can further improve their performances by using the multi-user transfer approach we proposed. These two observations suggest that different methods affect differently different types of users. Nevertheless, most users can improve their performances using multi-user transfer and shrinkage.

6 Discussion and guidelines

From the results obtained, we can identify a number of useful guidelines about which tools to use to reduce or suppress calibration time in which context. Overall, the first important and interesting point is that these results suggest that there are a number of signal processing tools that can significantly and subtantially improve classification performance as compared to the standard BCI design when very few training data are available. This also suggests that good classification performances can be

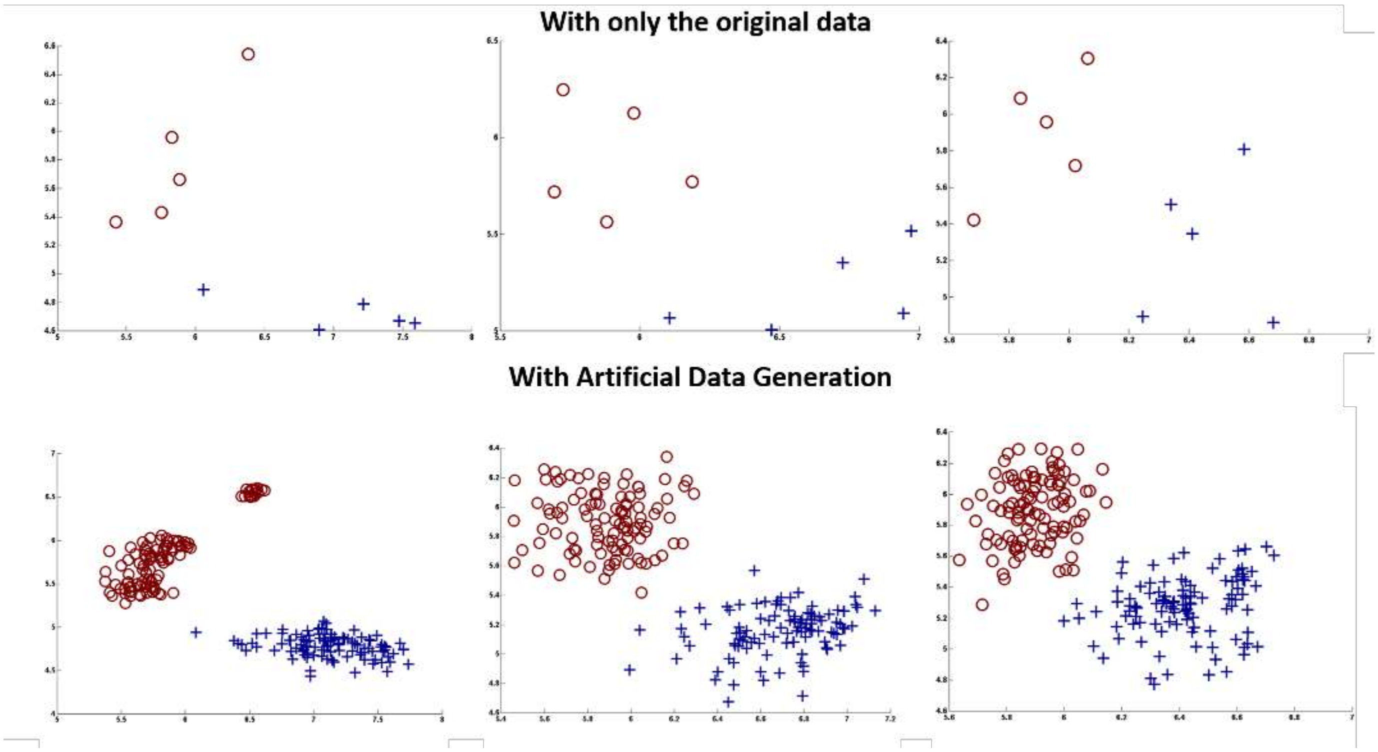

Fig. 14. Training set with and without artificial data generation, for user number 3 from data set 1, with 5 original training trials per class. The two classes are displayed with different shapes and colors. Here each pair of CSP features are diplayed (left: CSP feature 1 versus CSP feature 6, center CSP feature 2 versus CSP feature 5, right: CSP feature 3 versus CSP feature 4). It can be seen that artificial data generation indeed leads to different training examples and thus populate in a relevant way the feature space.

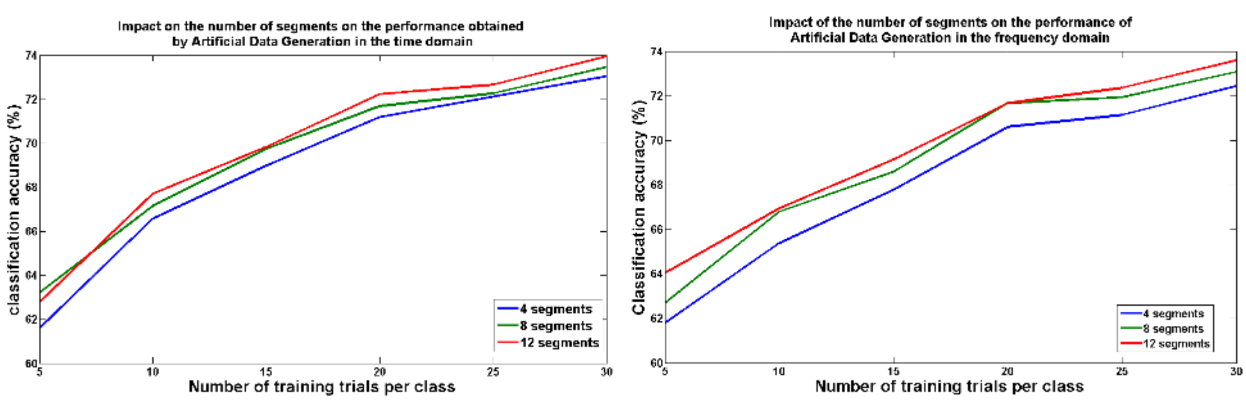

Fig. 16. Performances obtained by ADG methods for different number of segments used for data generation, on data set 3. Left: ADG in the time domain, Right: ADG in the time-frequency domain.

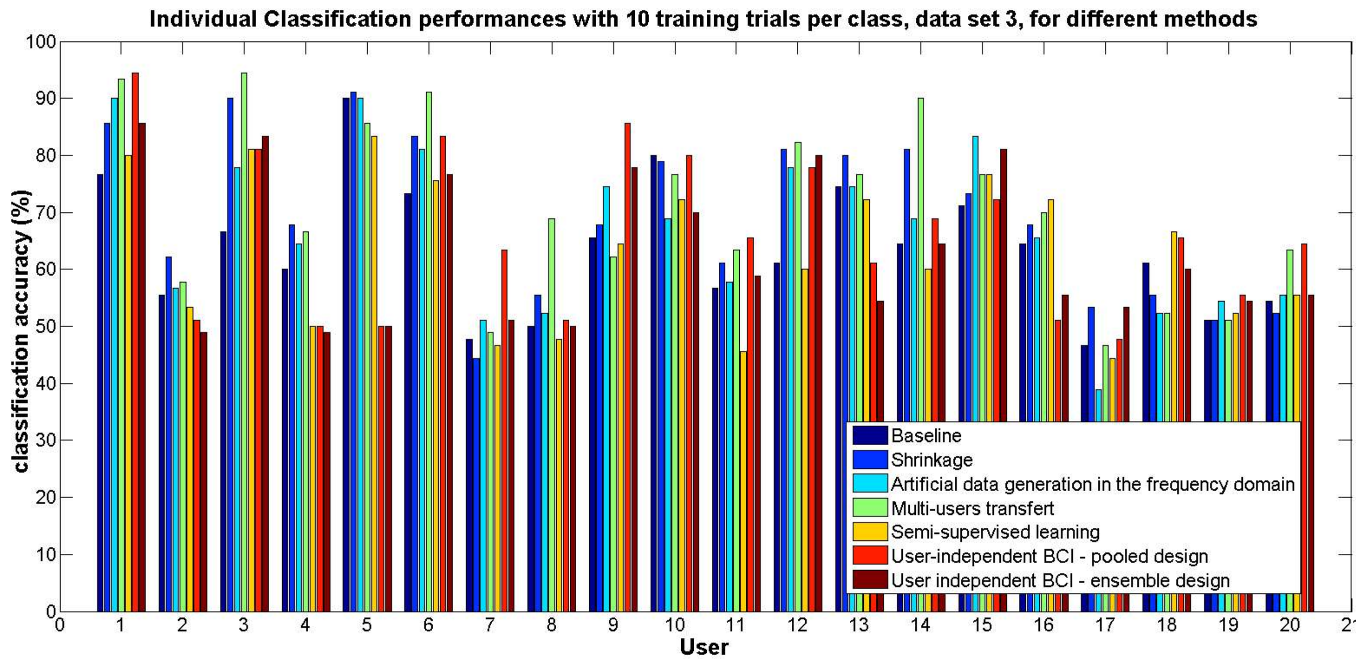

Fig. 17. Individual performances obtained by the 20 users from data set 3, with 10 original training trials per class, for different calibration reduction methods.

maintained with much fewer training data, thus effectively reducing BCI calibration time. On a more detailled level, we propose here the following guidelines:

- Automatic covariance matrix shrinkage should always be used, whatever the number of training trials. Indeed, it does not only enable calibration time reduction but also overall performance improvement, even when many training trials are used. Since this is a simple, computationally efficient and parameter free method, we advocate that it should become a standard tool for BCI design, including for oscillatory activity-based BCI (it is already becoming a rather standard tool for ERP classification [74]).
- If data from other users are available, user-to-user transfer is an efficient way to further reduce calibration time or even boost classification performance irrespectively of the number of training data, and should be used as a method of choice. The MU-BCI design proposed in this paper is a fast, simple and efficient method to do so.
- If no data from other users is available, artificial EEG trial generation combined with shrinkage can be considered to further reduce calibration time.
- Semi-supervised learning should NOT be used straightforwardly with CSP and LDA, due to their lack of robustness to mislabelling.
- There is not yet a gold standard for user-independent BCI design, no method significantly outperforming the other ones.
- Although user-independent BCI design is possible, performances are still rather poor and need further research to be improved.

Overall, although reducing BCI calibration time with the different approaches mentioned in this paper is possible and useful, calibration time reduction is still not a solved problem. Indeed, as the results suggested, the more the training data, the better the performances, even with the best calibration time reduction approaches so far. This is therefore still an open challenge. In the future, a number of points would be interesting to explore to further improve calibration time reduction or suppression, including 1) exploring optimal frequency band(s) selection as well (most methods discussed here only considered spatial filters and classifier optimization from few training data), several algorithms being available to optimize such bands with CSP and LDA, see, e.g., [75], [76], [77], [78], [79], [80], [81], 2) outlier/artifact removal in small sample size settings, as when few data are available, a single outlier (artefacts being unfortunately rather common in EEG measurements [82]) can have dramatic negative effects on the quality of the optimized BCI, 3) combining user-independent BCI design followed by online adaptation to completely remove calibration and quickly reach high performances, as suggested in [69], [83]. It would also be interesting to study the impact of calibration reduction technique on BCI training. Indeed, on the one hand, calibration reduction enables to provide BCI users with feedback as early as possible, such feedback being necessary for efficient training [84]. On the other hand, a poor feedback, due to a poorly calibrated BCI, may actually impede successful training. Online studies with calibration reduction techniques and online BCI training would thus need to be conducted.

7 Conclusion

In this paper we studied and analysed BCI design approaches that aim at reducing or suppressing calibration time for oscillatory activity-based BCI. We first surveyed existing approaches, which we identified as belonging to five categories of approaches: regularization approaches, user-to-user transfer approaches, semi-supervised learning, a-priori physiological information-based approaches and user-independent design. We then proposed a new category of methods to reduce calibration time, based on artificial data generation (ADG), and proposed three methods to do so. We also proposed a new and simple method for user-to-user transfer. Finally, we studied and compared methods from the different categories in order to identify guidelines about which method to use in which context. In short, automatic covariance matrix shrinkage should be used at all times, if data from other users are available user-to-user transfer approaches should be used to further boost performances and reduce calibration time, otherwise ADG with shrinkage can prove useful. User-independent approaches still have too modest performances to be used on the long-term, and are only useful for the very beginning of BCI use, before about 5 to 10 training trials per class become available.

Overall, this study suggested that there are a number of simple, fast and efficient tools that can and should be used routinely to reduce oscillatory activity-based BCI calibration time. For several of the calibration time reduction methods analyzed we can observe that only 10 trials per class are enough to reach the same performances as that obtained with 30 trials per class with the baseline design, hence effectively reducing BCI calibration time by 3. To encourage the adoption of such methods, the Matlab code of the different methods studied in this paper will be made available for free and open-source on http://sites.google.com/site/fabienlotte/code-and-softwares.

Nevertheless there is still a lot of room for improvements to further reduce calibration time and even more to suppress it altogether. New methods should be identified to alleviate between user-variability, e.g., by designing invariant features or efficient unsupervised adaptive BCI. Gathering and working with very large data bases of EEG signals may also help. Indeed, in the infancy of speech or hand-writing recognition, the pattern recognition systems were also user-specific. However, when data-bases with thousands of users' data became available, efficient user-independent systems could be created. So far, BCI data bases only very rarely reach 100 users, so we may also expect improvements in this direction.

Acknowledgments

The author would like to thank Camille Jeunet and Julia Schumacher for the collection of the Workload and Mental Imagery EEG data sets.

References

[1] J. Wolpaw and E. Wolpaw, *Brain-computer interfaces: principles and practice*. Oxford University Press, 2012.

[2] G. Fabiani, D. McFarland, J. Wolpaw, and G. Pfurtscheller, "Conversion of EEG activity into cursor movement by a brain-computer interface (BCI)," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 12, no. 3, pp. 331–338, 2004.

[3] J. R. Wolpaw, D. J. McFarland, G. W. Neat, and C. A. Forneris, "An EEG-based brain-computer interface for cursor control," *Electroencephalography and clinical neurophysiology*, vol. 78, pp. 252–259, 1991.

[4] G. Pfurtscheller, G. Müller-Putz, R. Scherer, and C. Neuper, "Rehabilitation with brain-computer interface systems," *IEEE Computer*, vol. 41, no. 10, pp. 58–65, 2008.

[5] A. Lécuyer, F. Lotte, R. Reilly, R. Leeb, M. Hirose, and M. Slater, "Brain-computer interfaces, virtual reality and videogames," *IEEE Computer*, vol. 41, no. 10, pp. 66–72, 2008.

[6] A. Nijholt, D. Plass-Oude Bos, and B. Reuderink, "Turning shortcomings into challenges: Brain-computer interfaces for games," *Entertainment Computing*, vol. 1, no. 2, pp. 85–94, 2009.

[7] T. Zander and C. Kothe, "Towards passive brain-computer interfaces: applying brain-computer interface technology to human-machine systems in general," *Journal of Neural Engineering*, vol. 8, 2011.

[8] J. van Erp, F. Lotte, and M. Tangermann, "Brain-computer interfaces: Beyond medical applications," *IEEE Computer*, vol. 45, no. 4, pp. 26–34, 2012.

[9] B. Blankertz, M.Tangermann, C. Vidaurre, S. Fazli, C.Sannelli, S.Haufe, C. Maeder, L. Ramsey, I. Sturm, G. Curio, and K. Mueller, "The berlin brain-computer interface: Non-medical uses of BCI technology," *Frontiers in Neuroprosthetics*, vol. 5, 2010.

[10] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi, "A review of classification algorithms for EEG-based brain-computer interfaces," *Journal of Neural Engineering*, vol. 4, pp. R1–R13, 2007.

[11] B. Blankertz, G. Dornhege, M. Krauledat, G. Curio, and K.-R. Müller, "The non-invasive Berlin brain-computer interface: Fast acquisition of effective performance in untrained subjects," *NeuroImage*, vol. 37, no. 2, pp. 539–550, 2007.

[12] H. Cecotti, "Spelling with non-invasive brain-computer interfaces - current and future trends," *Journal of Physiology-Paris*, 2011.

[13] J. Elshout and G. G. Molina, "Review of brain-computer interfaces based on the P300 evoked potential," Koninklijke Philips Electronics, Tech. Rep. PR-TN 2009/00066, 2009.

[14] R. Fazel-Rezai, B. Allison, C. Guger, E. Sellers, S. Kleih, and A. Kübler, "P300 brain computer interface: current challenges and emerging trends," *Frontiers in Neuroengineering*, vol. 5, no. 14, 2012.

[15] P.-J. Kindermans, D. Verstraeten, and B. Schrauwen, "A bayesian model for exploiting application constraints to enable unsupervised training of a P300-based BCI," *PloS one*, vol. 7, no. 4, p. e33758, 2012.

[16] P.-J. Kindermans, M. Tangermann, K.-R. Müller, and B. Schrauwen, "Integrating dynamic stopping, transfer learning and language models in an adaptive zero-training ERP speller," *Journal of neural engineering*, vol. 11, no. 3, p. 035005, 2014.

[17] S. Lu and C. Guan, "Boosting-based subject-independent brain computer interface," in *Proc. ICPR*, 2008.

[18] S. Lu, C. Guan, and H. Zhang, "Unsupervised brain computer interface based on inter-subject information and online adaptation," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 17, no. 2, pp. 135–145, 2009.

[19] F. Lotte and C. Guan, "An efficient P300-based brain-computer interface with minimal calibration time," in *Assistive Machine Learning for People with Disabilities symposium (NIPS'09 Symposium)*, 2009.

[20] J. H. Metzen, S. K. Kim, and E. A. Kirchner, "Minimizing calibration time for brain reading," in *Pattern Recognition*. Springer, 2011, pp. 366–375.

[21] Y. Li, C. Guan, H. Li, and Z. Chin, "A self-training semi-supervised SVM algorithm and its application in an EEG-based brain computer interface speller system," *Pattern Recognition Letters*, vol. 29, no. 9, pp. 1285–1294, 2008.

[22] M. Spüler, W. Rosenstiel, and M. Bogdan, "Online adaptation of a c-VEP brain-computer interface (bci) based on error-related potentials and unsupervised learning," *PloS one*, vol. 7, no. 12, p. e51077, 2012.

[23] R. Scherer, A. Schloegl, F. Lee, H. Bischof, J. Janŝa, and G. Pfurtscheller, "The self-paced Graz brain-computer interface: Methods and applications," *Computational Intelligence and Neuroscience*, vol. 2007, 2007.

[24] F. Lotte, H. Mouchère, and A. Lécuyer, "Pattern rejection strategies for the design of self-paced EEG-based brain-computer interfaces," in *International Conference on Pattern Recognition (ICPR)*, 2008, pp. 1–5.

[25] A. Ramos-Murguialday, D. Broetz, M. Rea, L. Läer, Ö. Yilmaz, F. L. Brasil, G. Liberati, M. R. Curado, E. Garcia-Cossio, A. Vyziotis *et al.*, "Brain–machine interface in chronic stroke rehabilitation: A controlled study," *Annals of neurology*, vol. 74, no. 1, pp. 100–108, 2013.

[26] K. K. Ang and C. Guan, "Brain-computer interface in stroke rehabilitation," *Journal of Computing Science and Engineering*, vol. 7, no. 2, pp. 139–146, 2013.

[27] C. Mühl, C. Jeunet, and F. Lotte, "EEG-based workload estimation across affective contexts," *Frontiers in Neuroscience section Neuroprosthetics*, vol. 8, p. 114, 2014.

[28] J. Frey, C. Mühl, F. Lotte, and M. Hachet, "Review of the use of electroencephalography as an evaluation method for human-computer interaction," in *Proc. of PhyCS*, 2014.

[29] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe, and K.-R. Müller, "Optimizing spatial filters for robust EEG single-trial analysis," *IEEE Signal Proc Magazine*, vol. 25, no. 1, pp. 41–56, 2008.

[30] G. Pfurtscheller and C. Neuper, "Motor imagery and direct brain-computer communication," *proceedings of the IEEE*, vol. 89, no. 7, pp. 1123–1134, 2001.

[31] B. Blankertz, F. Losch, M. Krauledat, G. Dornhege, G. Curio, and K.-R. Müller, "The berlin brain-computer interface: Accurate performance from first-session in BCI-naive subjects," *IEEE Trans Biomed Eng*, vol. 55, no. 10, pp. 2452–2462, 2008.

[32] O. Ledoit and M. Wolf, "A well-conditioned estimator for large-dimensional covariance matrices," *Journal of Multivariate Analysis*, vol. 88, no. 2, pp. 365–411, 2004.

[33] F. Lotte and C. Guan, "Learning from other subjects helps reducing brain-computer interface calibration time," in *Proc. ICASSP*, 2010, pp. 614–617.

[34] C. Sannelli, C. Vidaurre, K.-R. Müller, and B. Blankertz, "CSP patches: an ensemble of optimized spatial filters. an evaluation study," *J. Neural Eng.*, vol. 8, 2011.

[35] H. Lu, H.-L. Eng, C. Guan, K. Plataniotis, and A. Venetsanopoulos, "Regularized common spatial pattern with aggregation for EEG classification in small-sample setting," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 12, pp. 2936–2946, 2010.

[36] H. Kang, Y. Nam, and S. Choi, "Composite common spatial pattern for subject-to-subject transfer," *IEEE Signal Processing Letters*, vol. 16, no. 8, pp. 683 – 686, 2009.

[37] R. Caruana, “Multitask learning,” *Machine Learning*, vol. 28, pp. 41–75, 1997.

[38] M. Alamgir, M. Grosse-Wentrup, and Y. Altun, “Multitask learning for brain-computer interfaces,” in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS 2010)*, 2010, pp. 17–24.

[39] D. Devlaminck, B. Wyns, M. Grosse-Wentrup, G. Otte, and P. Santens, “Multisubject learning for common spatial patterns in motor-imagery BCI,” *Computational Intelligence and Neuroscience*, vol. 2011, 2011.

[40] H. Kang and S. Choi, “Bayesian common spatial patterns for multi-subject EEG classification,” *Neural Networks*, 2014.

[41] W. Samek, M. Kawanabe, and K. Muller, “Divergence-based framework for common spatial patterns algorithms,” *IEEE Reviews in Biomedical Engineering*, 2014.

[42] W. Tu and S. Sun, “A subject transfer framework for EEG classification,” *Neurocomputing*, vol. 82, pp. 109–116, 2012.

[43] D. Heger, F. Putze, C. Herff, and T. Schultz, “Subject-to-subject transfer for CSP based BCIs: Feature space transformation and decision-level fusion,” in *Proc. EMBC*. IEEE, 2013, pp. 5614–5617.

[44] M. Arvaneh, I. Robertson, and T. E. Ward, “Subject-to-subject adaptation to reduce calibration time in motor imagery-based brain-computer interface,” in *Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE*. IEEE, 2014, pp. 6501–6504.

[45] S. Dalhoumi, G. Dray, and J. Montmain, “Knowledge transfer for reducing calibration time in brain-computer interfacing,” in *International Conference on Tools with Artificial Intelligence*, 2014.

[46] C. Vidaurre, M. Kawanabe, P. Von Bunau, B. Blankertz, and K. Muller, “Toward unsupervised adaptation of LDA for brain–computer interfaces,” *Biomedical Engineering, IEEE Transactions on*, vol. 58, no. 3, pp. 587–597, 2011.

[47] P. Shenoy, M. Krauledat, B. Blankertz, R. Rao, and K.-R. Müller, “Towards adaptive classification for BCI,” *Journal of Neural Engineering*, vol. 3, no. 1, p. R13, 2006. [Online]. Available: http://stacks.iop.org/1741-2552/3/i=1/a=R02

[48] X. Zhu, “Semi-supervised learning literature survey,” Computer Sciences, University of Wisconsin-Madison, Tech. Rep. 1530, 2005.

[49] J. Meng, X. Sheng, D. Zhang, and X. Zhu, “Improved semi-supervised adaptation for small training data set in brain-computer interface,” *IEEE Journal of Biomedical and Health Informatics*, vol. 18, no. 4, 2014.

[50] W. Tu and S. Sun, “Semi-supervised feature extraction for EEG classification,” *Pattern Analysis and Applications*, vol. 16, no. 2, pp. 213–222, 2013.

[51] Y. Li and C. Guan, “Joint feature re-extraction and classification using an iterative semi-supervised support vector machine algorithm,” *Machine Learning*, vol. 71, no. 1, pp. 33–53, 2008.

[52] X. Song, S.-C. Yoon, and V. Perera, “Adaptive common spatial pattern for single-trial EEG classification in multisubject BCI,” in *Proc. IEEE-EMBS NER*. IEEE, 2013, pp. 411–414.

[53] M. Grosse-Wentrup, C. Liefhold, K. Gramann, and M. Buss, “Beamforming in non invasive brain computer interfaces,” *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 4, pp. 1209 – 1219, 2009.

[54] M. Ahn, H. Cho, and S. C. Jun, “Calibration time reduction through source imaging in brain computer interface (BCI),” in *HCI International 2011–Posters Extended Abstracts*. Springer, 2011, pp. 269–273.

[55] F. Lotte and C. Guan, “Regularizing common spatial patterns to improve BCI designs: Unified theory and new algorithms,” *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 2, pp. 355–362, 2011.

[56] F. Lotte, C. Guan, and K. Ang, “Comparison of designs towards a subject-independent brain-computer interface based on motor imagery,” in *Proc. EMBC*, 2009, pp. 4543 – 4546.

[57] J. Cantillo-Negrete, J. Gutierrez-Martinez, R. I. Carino-Escobar, P. Carrillo-Mora, and D. Elias-Vinas, “An approach to improve the performance of subject-independent BCIs-based on motor imagery allocating subjects by gender,” *Biomedical engineering online*, vol. 13, no. 1, p. 158, 2014.

[58] B. Reuderink, J. Farquhar, M. Poel, and A. Nijholt, “A subject-independent brain-computer interface based on smoothed, second-order baselining,” *Proc. EMBC*, 2011.

[59] M. Krauledat, M. Tangermann, B. Blankertz, and K.-R. Müller, “Towards zero training for brain-computer interfacing,” *PLoS ONE*, vol. 3, no. 8, p. e2967, 2008.

[60] S. Fazli, F. Popescu, M. Danóczy, B. Blankertz, K.-R. Müller, and C. Grozea, “Subject-independent mental state classification in single trials,” *Neural networks*, vol. 22, no. 9, pp. 1305–1312, 2009.

[61] S. Fazli, M. Danóczy, J. Schelldorfer, and K.-R. Müller, “l1-penalized linear mixed-effects models for high dimensional data with application to BCI,” *NeuroImage*, vol. 56, no. 4, pp. 2100–2108, 2011.

[62] W. Tu and S. Sun, “Dynamical ensemble learning with model-friendly classifiers for domain adaptation,” in *Proc. ICPR*. IEEE, 2012, pp. 1181–1184.

[63] R. Chakraborty and U. Garain, “Role of synthetically generated samples on speech recognition in a resource-scarce language,” *Proc. ICPR*, pp. 1618–1621, 2010.

[64] H. Mouchère, S. Bayoudh, E. Anquetil, and L. Miclet, “Synthetic on-line handwriting generation by distortions and analogy,” in *Proc. IGS*, 2007, pp. 10–13.

[65] F. Lotte, “Generating artificial EEG signals to reduce BCI calibration time,” *5th International Brain-Computer Interface Conference*, pp. 176–179, 2011.

[66] S. Bayoudh, L. Miclet, H. Mouchère, and E. Anquetil, “Learning a classifier with very few examples: knowledge based and analogy generation of new exemples for character recognition,” in *Proc. ECML*, 2007.

[67] L. Smith, “A tutorial on principal components analysis,” 2002. [Online]. Available: http://www.cs.otago.ac.nz/cosc453/student_tutorials/principal_components.pdf

[68] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten, “Multiclass brain–computer interface classification by riemannian geometry,” *Biomedical Engineering, IEEE Transactions on*, vol. 59, no. 4, pp. 920–928, 2012.

[69] M. Congedo, A. Barachant, and A. Andreev, “A new generation of brain-computer interface based on riemannian geometry,” *arXiv preprint arXiv:1310.8115*, 2013.

[70] M. Naeem, C. Brunner, R. Leeb, B. Graimann, and G. Pfurtscheller, “Seperability of four-class motor imagery data using independent components analysis,” *Journal of Neural Engineering*, vol. 3, pp. 208–216, 2006.

[71] C. Jeunet, F. Lotte, and C. Mühl, “Design and validation of a mental and social stress induction protocol: Towards load-invariant physiology-based detection,” in *Proc. PhyCS*, 2014, pp. 98–106.

[72] C. Jeunet, F. Lotte, M. Hachet, and B. N’Kaoua, “Impact

of cognitive and personality profiles on mental-imagery based brain-computer interface-control performance," in *IOP 2014*, 2014.
[73] E. V. Friedrich, C. Neuper, and R. Scherer, "Whatever works: A systematic user-centered training protocol to optimize brain-computer interfacing individually," *PloS one*, vol. 8, no. 9, p. e76214, 2013.
[74] B. Blankertz, S. Lemm, M. Treder, S. Haufe, and K.-R. Müller, "Single-trial analysis and classification of ERP components a tutorial," *Neuroimage*, 2010.
[75] K. Ang, Z. Chin, C. Wang, C. Guan, and H. Zhang, "Filter bank common spatial pattern algorithm on bci competition iv datasets 2a and 2b," *Frontiers in Neuroscience*, vol. 6, 2012.
[76] K. Thomas, C. Guan, T. Chiew, V. Prasad, and K. Ang, "New discriminative common spatial pattern method for motor imagery brain computer interfaces," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 11, 2009.
[77] G. Dornhege, B. Blankertz, M. Krauledat, F. Losch, G. Curio, and K.-R. Müller, "Combined optimization of spatial and temporal filters for improving brain-computer interfacing," *IEEE Trans. Biomed. Eng.*, vol. 53, no. 11, pp. 2274–2281, 2006.
[78] W. Wu, X. Gao, B. Hong, and S. Gao, "Classifying single-trial eeg during motor imagery by iterative spatio-spectral patterns learning (isspl)," *Biomedical Engineering, IEEE Transactions on*, vol. 55, no. 6, pp. 1733–1743, 2008.
[79] H. Zhang, Z. Y. Chin, K. K. Ang, C. Guan, and C. Wang, "Optimum spatio-spectral filtering network for brain–computer interface," *Neural Networks, IEEE Transactions on*, vol. 22, no. 1, pp. 52–63, 2011.
[80] H.-I. Suk and S.-W. Lee, "Subject and class specific frequency bands selection for multiclass motor imagery classification," *International Journal of Imaging Systems and Technology*, vol. 21, no. 2, pp. 123–130, 2011.
[81] ——, "A novel bayesian framework for discriminative feature extraction in brain-computer interfaces," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 2, pp. 286–299, 2013.
[82] M. Fatourechi, A. Bashashati, R. Ward, and G. Birch, "EMG and EOG artifacts in brain computer interface systems: A survey," *Clinical Neurophysiology*, vol. 118, no. 3, pp. 480–494, 2007.
[83] C. Vidaurre, C. Sannelli, K.-R. Müller, and B. Blankertz, "Machine-learning-based coadaptive calibration for brain-computer interfaces," *Neural computation*, vol. 23, no. 3, pp. 791–816, 2011.
[84] F. Lotte, F. Larrue, and C. Mühl, "Flaws in current human training protocols for spontaneous brain-computer interfaces: lessons learned from instructional design," *Frontiers in Human Neuroscience*, vol. 7, no. 568, 2013.

Fabien LOTTE obtained a M.Sc., a M.Eng. and a PhD degree in computer sciences, all from the National Institute of Applied Sciences (INSA) Rennes, France, in 2005 (MSc. and MEng.) and 2008 (PhD) respectively. As a PhD candidate he was part of Inria Rennes Bretagne-Atlantique and member of the OpenViBE project dedicated to brain-computer interfaces and virtual reality. His PhD Thesis received both the PhD Thesis award 2009 from AFRIF (French Association for Pattern Recognition) and the PhD Thesis award 2009 accessit (2nd prize) from ASTI (French Association for Information Sciences and Technologies). In 2009 and 2010, he was a research fellow at the Institute for Infocomm Research (I2R) in Singapore, working in the Brain-Computer Interface Laboratory. Since January 2011, he is a Research Scientist (with tenure) at Inria Bordeaux Sud-Ouest, France, in team Potioc (http://team.inria.fr/potioc/). His research interests include brain-computer interfaces, human-computer interaction, pattern recognition and signal processing.