Data-efficient Motor Imagery Decoding in Real-time for the Cybathlon Brain-Computer Interface Race

Eduardo G. Ponferrada[1], Anastasia Sylaidi[1] and A. Aldo Faisal[1,2,3]
[1]*Department of Bioengineering, Imperial College London, London, U.K.*
[2]*Department of Computing, Imperial College London, London, U.K.*
[3]*Data Science Institute, London, U.K.*

Keywords: Brain-Computer Interfaces, Machine Learning, Cybathlon.

Abstract: Neuromotor diseases such as Amyotrophic Lateral Sclerosis or Multiple Sclerosis affect millions of people throughout the globe by obstructing body movement and thereby any instrumental interaction with the world. Brain Computer Interfaces (BCIs) hold the premise of re-routing signals around the damaged parts of the nervous system to restore control. However, the field still faces open challenges in training and practical implementation for real-time usage which hampers its impact on patients. The Cybathlon Brain-Computer Interface Race promotes the development of practical BCIs to facilitate clinical adoption. In this work we present a competitive and data-efficient BCI system to control the Cybathlon video game using motor imageries. The platform achieves substantial performance while requiring a relatively small amount of training data, thereby accelerating the training phase. We employ a static band-pass filter and Common Spatial Patterns learnt using supervised machine learning techniques to enable the discrimination between different motor imageries. Log-variance features are extracted from the spatio-temporally filtered EEG signals to fit a Logistic Regression classifier, obtaining satisfying levels of decoding accuracy. The systems performance is evaluated online, on the first version of the Cybathlon Brain Runners game, controlling 3 commands with up to 60.03% accuracy using a two-step hierarchical classifier.

1 INTRODUCTION

Individuals suffering from severe neuromotor disorders, such as Multiple Sclerosis, Spinal Cord Injury or Amyotrophic Lateral Sclerosis, face extraordinary barriers in communicating with their environment in their daily lives. Overcoming such barriers through the use of technology, while achieving successful adoption by the end-users remains an open problem (Makin et al., 2017). Previous work in assistive technology towards addressing this challenge includes the development of a robotic arm controlled by 3D eye-tracking which is able to assist reaching movements (Maimon-Mor et al., 2017; Dziemian et al., 2016; Tostado et al., 2016). This system identifies movement targets based on binocular gaze points and selectively employs a trigger to activate the robotic arm. In a follow-up study, different methods to control such triggers were evaluated, inferring that blink detection can achieve better performance in robotic control than electromyography (EMG) or voice (Noronha et al., 2017). Despite this available range of triggering approaches, the overall robust control of robotic interfaces, remains substantially difficult for severely paralyzed individuals, where movement is extremely limited. To address such cases, our lab has previously developed an online non-invasive Brain-Computer Interface (BCI) to control external environmental entities with no muscle activity involved, using deep learning techniques for multi-class closed-loop motor imagery decoding (Ortega et al., 2018b; Ortega et al., 2018a). Similar decoding strategies have also been proven successful for offline BCIs in previous studies (Yang et al., 2015; Tabar and Halici, 2017; Lu et al., 2017). Despite its decoding advantages, this type of BCI uses supervised machine learning to match brain activity to user intents, requiring a significant amount of training data obtained throughout lengthy training sessions. However, long training times prior to BCI control pose a rejection risk from the end-users' side and can impair a mainstream adoption process.

To address this limitation, here, we develop a data-efficient BCI system, which maintains decoding per-

Ponferrada, E., Sylaidi, A. and Faisal, A.
Data-efficient Motor Imagery Decoding in Real-time for the Cybathlon Brain-Computer Interface Race.
DOI: 10.5220/0006962400210032
In *Proceedings of the 6th International Congress on Neurotechnology, Electronics and Informatics (NEUROTECHNIX 2018)*, pages 21-32
ISBN: 978-989-758-326-1

formance, while reducing the amount of required training data, thereby facilitating successful BCI adoption in a clinical setting. Data-efficient BCI systems have been previously proposed primarily for offline motor imagery decoding (Ferrante et al., 2015). Generally, to this date, the majority of BCI research is dominated by offline decoding studies (Yang et al., 2015; Tabar and Halici, 2017; Lu et al., 2017; Zhang L. and Wang, 2010; Obermaier et al., 2001), with only a few online testing cases of non-invasive BCIs (Holz et al., 2015). Our system builds upon these emerging approaches.

The Cybathlon (Riener and Seward, 2014) is a competition in which people with disabilities compete in a series of motor control activities using advanced assistive technology. The Cybathlon's Brain-Computer Interface Race is one of the presented activity tests intended for individuals with neuromotor disorders, that consists of a competitive racing video game, during which participants have to control an avatar using strictly their brain activity.

Our system is particularly focused towards developing a competitive platform for the Cybathlon BCI Race task, motivated by encouraging recent online decoding results within the same context (Ortega et al., 2018a) (Schwarz et al., 2016).

The data-efficient BCI system we developed in order to tackle the Cybathlon challenge, is based on Common Spatial Patterns (Ramoser et al., 2000) (CSP) and a logistic regression classifier (Tomioka et al., 2007) to decode motor imageries.

The video game mechanics are simple. The players of the game run on a given track at steady velocity throughout the whole duration of the race. The track occasionally changes colors to indicate switches in desired commands and the players have to execute these commands efficiently in order to obtain an advantage. Executing the wrong command whilst on a coloured section, as well as sending any command whilst on the grey (baseline) area of the track will make the avatar fall behind by a few seconds. The first player reaching the finish line wins the game.

Our system was developed and adjusted for the first version of the Cybathlon Brain Runners videogame. It is worth noting that this version was the first release of the Cybathlon videogame, which became significantly harder in the final version used in the competition.

The Cybathlon BCI Race rules state that the game must be controlled using strictly brain activity. We decided to control the execution of each one of the commands using motor imageries, namely, thoughts of executing different intuitive body movements. We thus developed a system able to decode motor thoughts and execute their associated video game command in real time. We treated the no-command -necessary when the avatar is on the grey sections of the track- as an additional command, amounting to four alternative motor imageries altogether, available for decoding. The motor imageries we decoded for control purposes consisted on imaginations of opening and closing the right hand, opening and closing the left hand, flexing and relaxing both feet at the ankle (dorsiflexion) and contracting and relaxing the abdominal muscles. Figure 1 presents the association between section colour and motor imageries.

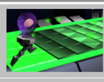
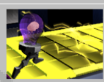
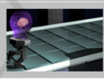
Game Event		Motor Imagery
	Green Pad	Open/Close Right Hand
	Purple Pad	Open/Close Left Hand
	Yellow Pad	Contract/Expand Feet
	Grey Pad	Contract Abdominal Muscles

Figure 1: Association of game events with motor imageries.

2 METHODS

The BCI system is composed of two main modules. The first one is the offline training pipeline in which the parameters of the BCI model are learn using user-specific supervised machine learning models trained on EEG recordings. The second module consists on the online decoding system, which performs real-time data streaming and inference using the model learnt during the offline training process to decode the player's intention continuously.

2.1 Training Pipeline

The aim of the BCI Training pipeline is to learn the parameters of a machine learning model for motor imagery detection that will later be used for real-time motor imagery decoding. These parameters are a set of CSP filters (Ramoser et al., 2000) for feature extraction and the parameters of a Logistic Regression classifier. The training pipeline is shown on Figure 2. The system was trained on two healthy subjects who had never participated in BCI studies or used any BCI system before - one of them (LG) was a twenty one year-old left-handed male and the other one (EG) was a twenty three year-old right-handed male. Informed consent was obtained and the study was performed according to institutional standards for the protection of human subjects. In this section we describe the details of the training pipeline.

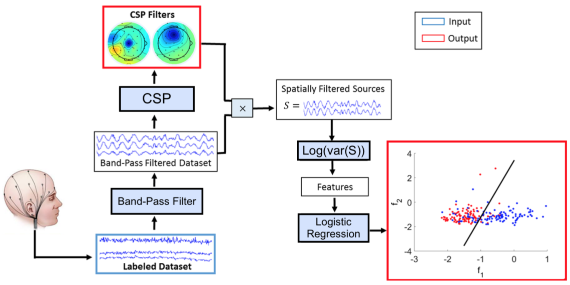


Figure 2: BCI model training: Labeled EEG recordings of motor imageries are fed to the training pipeline. The signals are band-pass filtered between 7 and 30 Hz to capture the mu and beta frequency bands associated to sensorimotor rhythms. The band-pass filtered data is used to fit the Common Spatial Patterns, obtaining pairs of filters, such that one maximizes the variance of one specific motor imagery and minimizes the variance of another one, and the other filter does the opposite. The logarithm of the variance of the spatially filtered sources are used as the features to fit a Logistic Regression classifier. Variations of the model were performed using 2-4 motor imageries to classify.

Experimental Setup for Training Data Acquisition

The subject sits in a comfortable chair at approximately 70 centimeters away from a computer screen. The EEG signal is recorded using a Brain Products' Easy-Cap with 32 Ag-AgCl active actiCAP electrodes located according to the International 10-20 system. High viscosity electrolyte gel for active electrodes is used to reduce the impedance between the electrode and the scalp under a tolerance level of 50 KOhm. 5 electromyogram (EMG) bipolar pairs of electrodes are located on the subject's right and left forearm (around the extensor retinaculum area), on the right and left feet (around the extensor retinaculum area) and on the torso (around the rectus abdominis area). One pair of bipolar electrooculogram (EOG) electrodes are located around the vertical vEOG up and vEOG down locations. Figure 3 shows the experimental setup. The BCI user watches game-play videos of the game and executes motor thoughts when the avatar reaches different sections of the race track. The motor imageries consist on imaginations of opening and closing the right hand, opening and closing the left hand, flexing and relaxing both feet at the ankle (dorsiflexion) and contracting and relaxing the abdominal muscles and their associated game events are detailed on Figure 1. This way we are able to gather and label motor imagery data from a 32-electrode EEG cap. Note that for this study, we used the first version of the Cybathlon Brain Runners videogame, which differs from the final one used in the Cybathlon competition. In order to homogenize the number of pads of each class, the five videos that maximized the homogeneity on the number of pads of each type were selected, yielding a total number of 90 trial events (22 green, 23 purple, 23 yellow and 22 grey). Each of these five videos was repeated four times during the experiment, thus, a total of 20 videos were played containing a total of 360 trial events (88 green, 92 purple, 92 yellow and 88 grey). Performing motor imageries is a tiring task, so after each trial, the subject is allowed to rest and a black screen appears informing him how many videos he has already completed and asking him to press a key to continue with the new video whenever he feels prepared.

Figure 3: Experimental setup for training data acquisition (the first author shown using the systems in the photograph). The subject sits on a comfortable chair with the 32 active electrodes cap recording his EEG signal, 5 pairs of bipolar EMG recording his muscle activity and one pair of bipolar EOG electrodes recording his eye movements and blinks.

Training the BCI Model

After collecting and labelling training data for the different motor imageries for a given user, we feed it to the training pipeline. Our approach focuses on spectral and spatial filtering, using a band-pass filter and learnt Common Spatial Patterns respectively in event-locked time windows in order to capture Event-Related-Desynchronisation (ERD) and Event-Related-Synchronisation (ERS) features associated to the motor imageries (Neuper et al., 2006). Once the features of interest are extracted through spectral and spatial filtering, we use them to fit a Logistic regression classifier to discriminate across the different motor imageries. For clarity reasons, this paper first describes the training platform to decode two motor imageries and later on, an adaptation for for multi-class classification. The training process as shown on Figure 2, consists on band-pass filtering, selecting a time-window, fitting the CSP filters, spatially filtering the signal using the learnt CSP filters, extracting log-variance features and fitting a classifier.

Band-pass Filtering and Time-window Selection

The first step on the training process is band-pass filtering the labeled EEG recordings. ERD and ERS on the sensorimotor cortex are features observed on EEG signals associated with motor imageries that are prominent on the μ and β frequency bands. In order to capture these events, the EEG recording is band-pass filtered using a Finite Impulse Response (FIR) filter with cutoff frequencies between 7 and 30 Hz and using a pass-band ripple of -20 dB and a stop-band ripple of -40 dB.

ERD/ERS are shown to be active from 0.5 seconds after the stimulus onset and last between 2 and 3 seconds (Neuper et al., 2006). Thus, the EEG chunk of interest for our motor imagery control purpose is contained within this time-window. We were also interested on building a BCI system that reacts as fast as possible to the user's intent, for online decoding purposes. To achieve this, ideally, we would use the shortest possible time-window while keeping enough information on the EEG signal to provide a reliable decoding. Considering this, we selected a time window from 0.5 seconds after the motor imagery onset until 2.5 seconds after the motor imagery onset, obtaining multiple samples of 2-seconds long labeled time-series to train the system capturing ERD/ERS events.

Learning Optimal CSP Filters

Common Spatial Patterns (Ramoser et al., 2000) are pairs of filters that linearly transform the multichannel time series of an EEG signal into lower-dimensional time series in such a way that the first filter maximizes the voltage variance across time of signals of Class A, while minimizing the variance of signals of Class B and the second filter does the opposite, it minimizes the variance of signals of Class A and maximizes the variance of signals of class B. Class A and B are two different motor imageries.

We show this in Figure 4, on a left-hand versus right hand motor imagery trial. If we observe the voltage distribution on a given trial on the C3 and C4 EEG channels, located on the left and right hemisphere of the motor cortex respectively (Figure 4A), we can observe that the voltage distribution of both classes have an important amount of overlap and thus, the variance of both clusters on each channel would not be a good feature to separate the different-class trials using a linear classifier. After filtering with a pair of learnt spatial filters on Figure 4B, the signals obtained for the first source S1 (after filtering with the first CSP filter 1), are maximal in variance for the blue cluster, corresponding to left hand movement and minimal for the red cluster, corresponding to right hand movement. The opposite happens with the second source S2 (after filtering with the converse CSP filter 2). In this case the red cluster, corresponding to right hand movement event has maximal variance and the blue cluster, corresponding to left hand movement have minimal variance. The signals are now much more linearly separable by extracting variance-based features.

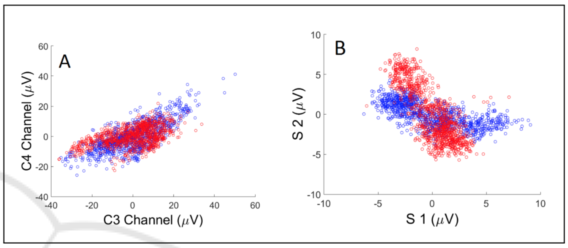


Figure 4: A) The distribution of a raw EEG signal of a right hand trial (red) and left hand trial (blue) before the Common Spatial Filters are applied. B) Same trials after spatial filtering using a single pair of optimized CSP filters.

The supervised Machine Learning algorithm to learn the CSP filters are is described as the following generalized eigenvalue problem, using the left versus right hand motor imageries for illustration purposes:
We compute the average covariance matrices for each class over all trials of our band-pass filtered 2-second long time-series training dataset, obtaining $\Sigma_\mathbf{L}$ and $\Sigma_\mathbf{R}$, for left and right hand classes respectively, and find the simultaneous diagonalizer $\mathbf{W}$ of $\Sigma_\mathbf{L}$ and $\Sigma_\mathbf{R}$:

$$\begin{aligned} \mathbf{W}^\mathbf{T}\Sigma_\mathbf{L}\mathbf{W} &= \mathbf{D}_\mathbf{L} \\ \mathbf{W}^\mathbf{T}\Sigma_\mathbf{R}\mathbf{W} &= \mathbf{D}_\mathbf{R} \end{aligned} \tag{1}$$

Such that the diagonal matrices $\mathbf{D}_\mathbf{L}$ and $\mathbf{D}_\mathbf{R}$, follow:

$$\mathbf{D}_\mathbf{L} + \mathbf{D}_\mathbf{R} = \mathbf{I} \tag{2}$$

This formulates a generalized eigenvalue problem of the form:

$$\Sigma_\mathbf{L}\mathbf{W} = \lambda\Sigma_\mathbf{R}\mathbf{W} \tag{3}$$

This is satisfied for $\mathbf{W}$ consisting of the generalized eigenvectors $\mathbf{w_j}$ (j=1,, C) and λ being: $\lambda = \lambda_L/\lambda_R$ and ${\lambda_j}^C (c \in \{L,R\})$, the diagonal elements of $\mathbf{D}_\mathbf{L}$ and $\mathbf{D}_\mathbf{R}$, respectively. It is important to note that $\lambda_j^C \geq 0$, is the variance of class c in the CSP source j, and that

${\lambda_j}^L + {\lambda_j}^L = 1$. Thus, a value ${\lambda_j}^L$ close to one, means that the spatial filter $\mathbf{w_j}$ yields a high variance in the left hand class and a low variance on the right hand class and the opposite if it is close to zero. This makes both classes easily discriminative by computing the variance across time on the CSP filtered time series.

Once we obtain the eigenvectors matrix, we take the n first and the n last columns of $\mathbf{W}$ ($n = 3$ in our case), as they are maximally informative to extract variance features. The pair of filters composed by the first and last columns are the ones that provide maximal discriminative information between the variance of the two classes, but the rest $n-1$ pairs provide robustness for classification.

Since the Common Spatial Patterns are vectors of dimension equal to the number of EEG channels, they can be visualized using bipolar plots, which provide useful information regarding the EEG channels that are enhanced or suppressed to provide maximal discrimination between classes and gives us a good intuition about which areas of the brain the CSP filter pairs are trying to extract information from. An example of the bipolar plot of a pair of common spatial filters optimized for subject EG2608's right hand versus left hand data is shown on Figure 5. CSP Pattern 1 shows that the filter is amplifying the depolarization associated with the activation of the left hemisphere on the motor cortex, which is known to be related to right hand movements. CSP Pattern 2, in contrast, amplifies the depolarization of a similar area of the motor cortex but in the right hemisphere, associated to left hand movements. The visualization of these filters, thus, provide a good interpretation on whether the CSP filters are capturing ERD/ERS features or other irrelevant EEG patterns, such as artifacts, in which case we would observe a strong weighting on the frontal electrodes. The visualization of spatial filters also provide with an estimation on which parts of the brain are being activated associated to different motor imageries, which can vary across different subjects and can be taken into account to decide which motor imageries a specific subject should use to control the BCI video game.

Fitting a Logistic Regression Classifier

After fitting the CSP filters, we extract variance features on the spatially filtered signal that are informative of the underlying motor imagery.

First, we filter all time series of each class with the spatial filters. An EEG time-series is $\mathbf{X} \in \mathbb{R}^{t \times c}$ being t, the number of time steps contained on the 2 seconds epoch, ($t = 1000$ in our case, as signals are sampled at 500 Hz) and c the number of channels of the EEG signal (31, in our case). Our learnt CSP filters are, $\mathbf{W} \in \mathbb{R}^{t \times 2n}$, being n the number of pairs of filters being used (columns of the eigenvector matrix), which is $n = 3$ in our case. Thus, each filtered time-series $\mathbf{S} \in \mathbb{R}^{t \times 2n}$ becomes:

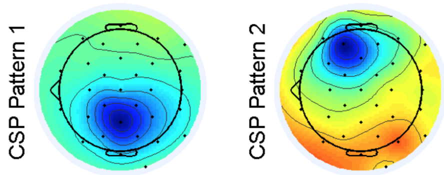


Figure 5: Bipolar plot of the first pair of Common Spatial Patterns from subject EG2608 optimized for the classification between right hand (CSP Pattern 1) and left hand (CSP pattern 2). CSP Pattern 1 shows that the filter is strongly weighting the left hemisphere of the motor cortex, associated to right hand movements. The second filter of the pair, CSP Pattern 2, in contrast, strongly weights the the right hemisphere of the motor cortex, associated to left hand movements. The colour map represents the weights assigned to each of the channels by the spatial filters, dark blue being the highest and red being the lowest. Bipolar plots created using BCILAB (Kothe and Makeig, 2013) using a cubic interpolation of $\mathbf{W^{-1}}$.

$$\mathbf{S} = \mathbf{XW} \tag{4}$$

We then extract features from the spatially filtered signals, which are discriminative with respect to their variance. Thus, we build a feature space based on the logarithm of the variance of S across time. The logarithm compresses the range and reduces the impact of outliers.
The features $\mathbf{f} \in \mathbb{R}^{1 \times 2n}$, are thus:

$$\mathbf{f} = \log(\mathrm{var}(\mathbf{S})) \tag{5}$$

We finally used the extracted features to train the L2-regularized Logistic Regression classifier using BCILAB (Kothe and Makeig, 2013), which is built upon the LIBLINEAR (Fan et al., 2008) and CVX (Grant and Boyd, 2008) packages.

In order to evaluate the classification performance, we ran a 5-fold cross-validation on the training dataset. Importantly, to avoid correlation between nearby trials, we performed a block-wise cross validation leaving 5 trials out between the training and test blocks. Doing so attempts to remove correlated trials between training and test sets on each cross-validation split to avoid unrealistic offline decoding accuracy results.

Multi-class Motor Imagery Classification

CSP generates optimized spatial filters to discriminate between two motor imageries by linearly transfor-

ming the EEG signal to maximize or minimize the variance of one of the pair of motor imageries. Despite its binary nature, it can be extended to work with more than two classes by computing the filters for all possible pairs of classes and fitting classifiers for all possible pairs. We then compute the normalized probability of each class across all binary Logistic Regression classification tasks and select the class with maximum probability.
As an alternative, we also developed a two-step hierarchical binary classifier. For the first step, we fit a model to classify the motor imagery associated to the no-command signal against all the rest of motor imageries. The second step is the standard classification problem as described before for the remaining motor imageries.

The decoding then starts on the first step, using a Logistic Regression classifier. If the rest of the classes wins against the no-command class, then the system enter the second decoding step to classify which is the most likely motor imagery out of the rest of the classes using a Logistic Regression classifier.

2.2 Real-time BCI Module

Once the parameters of the BCI model have been estimated running the training pipeline, they are used to decode the mental state of the BCI user in real time and send commands to control the video game accordingly. Figure 6 summarizes the transition from the real-time acquisition of EEG data to the translation of this information into a videogame command. In this section we describe the details of real-time BCI module.

From an EEG Signal to a Video Game Command

In this section, we describe the specific case for two classes and a Logistic Regression classifier is explained, although variations using more than two classes were also implemented.

We used a modified version of the RDA Server for the BrainVision Recorder, streaming the data directly from the PyCorder recording software into MATLAB. A 10 seconds ring buffer was implemented and updated every 50 milliseconds.

The 10 seconds-long RDA server buffer is filtered using a Finite Impulse Response (FIR) filter with cutoff frequencies between 7 and 30 Hz and using a pass-band ripple of -20 dB and a stop-band ripple of -40 dB. Once the buffer is band-pass filtered, just the last two seconds are used for decoding, matching the time window length used to train the BCI model. The reason for using a 10 seconds buffer for filtering is that the band-pass filter transients might distort the first time steps of the signal, but the last two seconds, which are the part we are interested on, are free of distortions.

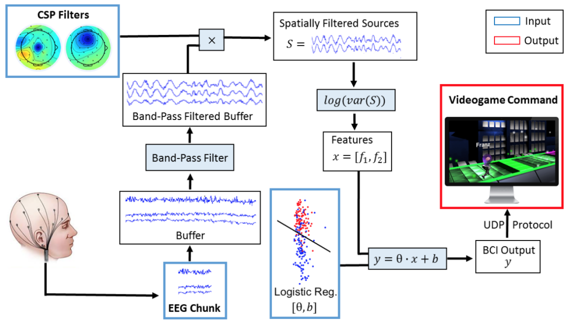


Figure 6: Real-Time Decoding BCI module. A chunk of EEG data is streamed into a buffer of 10 seconds length every 50 milliseconds. This buffer is accessed by the decoding thread, which first applies a band-pass filter between 7 and 30 Hz. Then, it takes the most recent 2 seconds of the buffer and applies the CSP filters computed on the training pipeline to it, obtaining a set of spatially filtered signals. The feature vector is obtained by computing the log-variance of the signals across time. The log-variance feature vector is classified using the Logistic Regression classifier. The BCI system finally sends the corresponding command to the video game system via UDP Protocol.

The 31-channel, 2 seconds-long EEG signal is spatially filtered with the 3 pairs of CSP filters producing a time series of 2 seconds and 6 source channels. The logarithm of the variance of each of these time series is computed to produce a feature vector, which is finally classified using the logistic regression classifier:

$$\mathbf{y} = \frac{1}{1 + e^{-(\mathbf{xw}+\mathbf{b})}} \tag{6}$$

Being $\mathbf{x} \in \mathbb{R}^{1\times 2n}$ the log-variance feature, n the number of pairs of filters (3 in our case) and $\mathbf{w}$ and $\mathbf{b}$ the parameters of the Logistic Regression Classifier. $\mathbf{y}$ is a probabilistic output informative about the probability of the input signal belonging to class 1 (e.g Right Hand motor imagery) and the converse $1-\mathbf{y}$ is the probability of the input signal to belonging to class 2 (e.g. Left Hand motor imagery).

Finally, we map the classification output to a video game command. The video game has three commands that execute different actions. In order to send these commands to the video game, a specific value has to be sent via UDP protocol to the specific IP address of the game. In order to achieve a good performance on the Cybathlon video game, it is important not to decode the wrong command, especially during the grey pads within which no command must be sent for a long period of time. For this reason, we adopted a conservative strategy to avoid false positives and we only sent a command when a motor imagery is decoded with a probability of 0.8 or higher and a no-command otherwise.

Feedback to Improve Control of the System

The Real-Time BCI system provides feedback to the user by displaying bars, representing the probability of each of the motor imageries. This visual feedback is very important in order for the user to be able to adapt to the system. As in any natural learning task, after some practice using the system, the user will be able to adapt his brain activity to improve his skills controlling the system.

A Brain-Computer interface is a system in which both the computer and user's brain activity must adapt to each other to achieve an optimal performance. The computer learns from the user through supervised machine learning and the user learns how to best control the system through continuous training.

Once the users feel comfortable controlling the BCI, they start using the system to control the Cybathlon video game.

3 RESULTS

In the first part of his section, we introduce our results evaluating our decoding accuracy offline, using cross-validation on the acquired training data. In the second part, we characterise the online decoding system. Finally, we evaluate the real time decoding performance.

3.1 Offline Classification

We performed the described 5-fold block-wise cross-validation to evaluate the training pipeline performance. In this section we show the accuracy results for all possible combinations of two and three motor imagery decoding and the results for four motor imagery decoding on two different subjects. Accuracy is defined as:

$$Accuracy(\%) = (Correct/Total) \times 100 \quad (7)$$

For two motor-imagery classification, the results on all combinations of two pairs of motor imageries using CSP with a Logistic Regression classifier are reported on this section. The bars represent the accuracy on the classification of pairs of classes. The classes are labeled as S for thinking about contracting abdominal muscles (stomach), RH for thinking about opening and closing the right hand, LH for thinking about opening and closing the left hand and F for thinking about expanding and contracting the feet. The red line represents the chance level and the pink area represents a 95% confidence interval for the chance level (Müller-Putz et al., 2008). The blue bars represent the standard deviation for the classification accuracy.

Results for subjects LG2807 and EG2608 are shown on Figure 7. In the case of LG2807, our system is able to differentiate between right hand and feet with an accuracy of up to an 84.44%, however it is unable to differentiate between some other motor imageries, such as the case of stomach versus feet, for which the classification accuracy is within the chance level range. For subject EG2608, our system is able to decode three of the pairs of motor imageries with up to an 81.67% accuracy for right hand versus left hand classification. However, it is unable to differentiate between other pairs, such as the case of stomach versus feet.

Similarly to the two-motor imagery case, the classification results using CSP with a Logistic Regression on all combinations of three motor imageries are shown on Figure 8 and the classification results for the four motor imageries is shown in Figure 9. The way in which the data is represented is equivalent to the two motor imageries case.

The commands that subject LG2807 can best control for practical BCI purposes are feet versus right hand, with an 84.44% accuracy for two classes, right hand versus left hand versus feet with a 62.4% accuracy and for the four classes the decoding accuracy is 48.61%.
In the case of subject EG2608, the commands that can be best controlled are right hand versus left hand with an 81.67% accuracy for two classes, right hand versus left hand versus feet with a 56.65% accuracy for three classes and the decoding accuracy for the four classes is 44.72%.

3.2 Real-time BCI Characterization

We describe the real-time BCI system characteristics as:
Decoding delay: $t_d = 0.1214 \pm 0.0031$ seconds.
Decoding time-window length: $t_w = 2$ seconds.
ERD/ERS delay: $t_{ERD/ERS} = 0.5$ seconds.

The decoding delay refers to the minimum period of time that it takes for our system to perform a full decoding task, this is, since the moment in which the raw EEG data enters ours system until the moment in which the decoded command is sent to the video

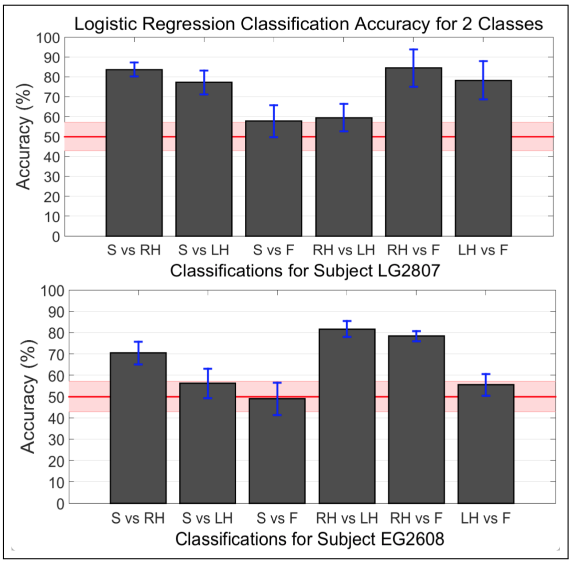


Figure 7: Classification accuracy between two motor imageries for subjects LG2807 (top) and EG2608 (bottom) using CSP with a Logistic Regression classifier. Motor imageries are labeled as: S = Stomach, RH = Right Hand, LH = Left Hand, F = Feet. The red line represents chance label and the pink area is a confidence interval for chance level. The blue bars represent the standard deviation for the classification accuracy. (These apply to all other figures).

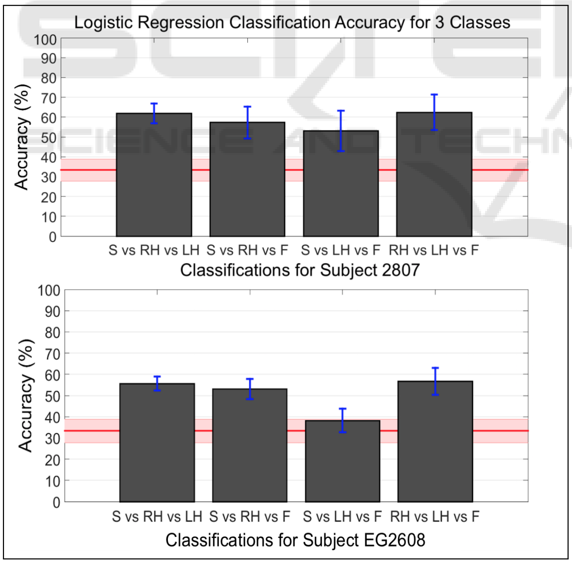


Figure 8: Classification accuracy between three motor imageries for subject LG2807 (top) and EG2608 (bottom) using CSP with a Logistic Regression classifier. Using the same legend described in Figure 7.

game system. This time is on average $t_d = 121$ milliseconds. The decoding response, however is limited by the length of the time window that we need to use for the decoding. The EEG features that we are using are relatively slow, at a maximum frequency of

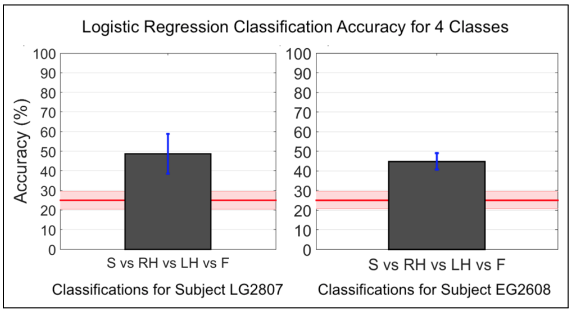


Figure 9: Classification accuracy between four motor imageries for subject LG2807 (top) and EG2608 (bottom) using the CSP with an Logistic Regression classifier. Using the same legend described in Figure 7.

30 Hz and even in the ideal case in which we would be able to identify an ERD/ERS feature instantly, we would still have a $t_{ERD/ERS} = 0.5s$ lag, which is an intrinsic delay associated to these specific EEG features (Neuper et al., 2006). For training, we use a time window between 0.5 seconds until 2.5 seconds after the event onset, in order to capture ERD/ERS events. For this reason, we need to keep a consistent in the time-window length for real-time decoding, $t_w = 2s$.

3.3 Real-time BCI Decoding Results

The real-time system was tested on one right-handed subject, EG2608 using CSP with a Logistic Regression classifier using two and three commands respectively. This was the second session for this subject. First, the subject completed the training phase to optimise the CSP filters and Logistic Regression classifier for this second session. This time, however, his offline performance was significantly lower than the during the first session, obtaining a maximum of 62.5% accuracy on two-class decoding for right hand versus feet, a maximum of 52.29% accuracy for three-class decoding for right hand versus left hand versus feet. The subject then played the Cybathlon game under three different conditions:

The first condition consisted on two commands decoding using a standard binary classification. Right hand motor imagery was assigned to control the baseline command and stomach motor imagery to control the accelerate command. The average accuracy across 10 game rounds was: 68.62 %

The second condition consisted on three commands using a three class classification. Right hand motor imagery was assigned to control the accelerate command, stomach motor imagery to control the jump command and left hand motor imagery to control the baseline command. The average accuracy across 3 game rounds was: 47.66%

The third condition consisted on three commands decoding using a two-step hierarchical classifier. Right hand motor imagery was assigned to control the accelerate command and stomach motor imagery was used to control the jump command. The first step of the hierarchical classifier decides wether any of the imageries are active and if the threshold is greater than 0.8, it makes a prediction using the right hand versus stomach classifier. If any of the motor imageries are decoded, its respective command is sent, otherwise no command is sent. The average online-decoding accuracy across 3 game rounds was: 60.03 %

The subject was not able to complete the online 4-class decoding task as he struggled to achieve control for 4 commands on the feedback phase. A snapshot of the subject playing the Cybathlon video game using motor thoughts is shown on Figure 10.

Figure 10: A subject plays the Cybathlon video game using thoughts. The three bars at the right of the screen represent the probability of a specific motor thought to be decoded. In this snapshot the subject is successfully executing a jump.

4 DISCUSION

In this work we demonstrate how a data-efficient EEG-based online BCI system can achieve competitive results on the Cybathlon BCI race task. The main advantage of our system is that it requires a small amount of training data, effectively translating in short data acquisition sessions prior to live BCI usage and potentially improving the BCI user experience. Our system was able to achieve an average decoding accuracy across subjects of 86%, 62% and 49% for the offline classification of two, three and four motor imageries respectively, evaluated using a correlation-corrected cross-validation approach. Our system was also able to achieve an online decoding accuracy of up to 60.03% controlling 3 commands, while only requiring the acquisition of 360 motor imagery trials.

Considerations Regarding CSP-based BCIs

Common Spatial Patterns have some drawbacks (Tomioka et al., 2007). CSP were originally described as a decomposition technique (Koles, 1991) rather than a tool for classification based on variance features. Nonetheless they have been extensively used for classification purposes in motor imagery based BCI applications with success.

One important problem, however, is that the simultaneous diagonalization of the covariance matrices suffers greatly from the presence of even a small amount of outlier trials on the EEG dataset, such as in cases of artifact presence.

Moreover, the nature of the CSP filters is binary. They attempt to maximize the difference in variance between pairs of classes. They can be extended to multi-class tasks as we described on the methods section but they are not directly optimized for multi-class classification. Besides this workaround, other alternatives to multi-class CSP have been attempted in the past and might pose an appropriate choice for the given problem (Grosse-Wentrup and Buss, 2008).

Previous Results on the Cybathlon BCI

Ortega et al. proposed an alternative decoding route on their BCI for the Cybathlon using a Convolutional Neural Network, consisting of a convolutional layer, followed by a fully connected layer and a softmax (Ortega et al., 2018a). They reported results for 4-class decoding, using a 4-fold cross-validation, both offline and online, obtaining up to 54.5% accuracy offline, which shows that optimizing the data acquisition approach to gather a larger dataset (9000 samples), allows deep neural networks to improve performance compared to simpler linear classifications on smaller datasets. Additionally, they reported 47% online accuracy for 4-class decoding.

Schwartz et al. also showed relevant results on their Cybathlon BCI system, using two CSP projections, extracting 12 power-band features and fitting a shrinkage regularized Linear Discriminant Analysis (Schwarz et al., 2016). They collected 1230 trials and obtained a 66.1% median accuracy offline, on the best combination of motor imageries using right hand versus left hand versus feet versus rest, which interestingly indicates a large improvement compared to their results using four motor imageries. These results demonstrate that including a rest state and gathering more data can boost decoding performance. They also reported a drop in online performance for 4-class decoding, down to 51.2% accuracy across all user decisions.

One of the reasons why our results may differ from the previous studies discussed above is the choice of a different cross-validation method. In previous work, a standard block-wise cross-validation was chosen. Instead, we attempted to propose a more robust cross-validation technique, which corrects for highly correlated contiguous trials between training and test blocks by discarding the 5 trials in between them for each split of our block-wise cross-validation. Correlated trials between training and test sets might be one of the factors involved in the offline to online accuracy drop reported in previous work, but perhaps not the only one. Note, that, even by using this advanced cross-validation technique to correct for correlated trials across training and test sets, we also experienced an offline to online accuracy drop on 3-classes decoding (although better results were obtained online than offline for the 2-class decoding case). This suggests that there could be additional factors involved in the offline to online accuracy drop. On the whole our platform manages to offer correlation-corrected cross-validation results for offline decoding, which are at competitive levels of performance compared to previous approaches, while significantly reducing the amount of training data needed to operate the BCI (360 samples compared to thousands of samples employed in other studies). It thereby poses a solid decoding system that is aimed at managing more efficiently the trade-off between decoding accuracy and the length of training sessions.

For online BCI decoding tasks other than the Cybtahlon, previous studies have shown promising results on time-locked tasks. Friedrich et al. achieved accuracies for 4-class decoding ranging between 61% and 72% on 14 subjects, where 8 of them achieved performance over the chance level for all 4 classes, including motor and non-motor imageries (Friedrich et al., 2013). In the case of 3-class decoding, Milln et al. showed accuracies ranging between 55% and 76% across 5 subjects, also including both motor and non-motor imageries (Millán et al., 2004).

Comparing Offline and Online Results

Results on real time decoding accuracy for subject EG2608 showed a classification accuracy of 68.62%, surpassing the offline cross-validation results for the same subject and session (62.5%). After several games, the subject was able to achieve good control over the video game tracks where only two commands were needed.

In the case of three commands, the online accuracy was 47.66%, showing a drop compared to the 52.29% accuracy obtained offline, evidencing that the control of three commands is much more challenging. We also demonstrated that 3-class accuracy can be greatly improved by using a two-step classification scheme, which executes binary classification on each step, obtaining a 60.03% accuracy.

In the three commands case, the subject reported that control was possible provided enough concentration but significantly harder than the two commands case. The subject also attempted 4-class decoding, however, during the feedback phase, the subject was unable to achieve meaningful control. It is worth noticing that previous studies on BCIs applied to communication have shown that a minimum of 70% accuracy is necessary for the system to be usable (Kübler et al., 2001). We consider improvements on the 4-class gaming control as one of the primary outlook research areas for the given decoding context and as a prioritized extension of the present version of our platform.

We also observed a high variability in performance across the two subjects and across different sessions, verifying that consistency and personalization is an important challenge that needs to be addressed for practical BCI adoption in clinical settings.

Data Efficiency Assessment

Our approach builds upon previous work on offline data-efficient EEG BCI decoding (Ferrante et al., 2015), where results for 2-class motor-imagery decoding using a CSP-based linear system showed an average accuracy of 88% when trained on just a dozen of samples per-class and takes it one step further, proving satisfying decoding performance for multi-class motor imagery decoding, as well as suitability for practical online BCI tasks.

In particular, our system was able to achieve competitive results compared to previous Cybathlon systems using a number of training samples one order of magnitude smaller. More precisely, we trained our classifier using 360 samples, achieving an offline 4-class mean decoding accuracy across subjects of 49% on our correlation-corrected cross-validation assessment. When compared against the offline block-wise cross-validation results from previous Cybathlon systems on the same task, one of them (Ortega et al., 2018a) showed a 11% increase in accuracy at the expense of a 2400% increase in training data volume and a second one (Schwarz et al., 2016) obtained a 35% accuracy increase at the expense of increasing training data volume by 242%.

The main characteristic making our system data-efficient is that it only needs to optimize a very small set of model parameters, compared to other approaches using much more complex non-linear models (Ortega et al., 2018a; Yang et al., 2015; Tabar and Halici, 2017; Lu et al., 2017). In fact, our classifier only needs to learn 6 weights and one bias term.

Overall, our system aims to find an optimal trade-off between accuracy and volume of training data needed prior to BCI live usage, so as to obtain satisfying levels on the former, while reducing the latter drastically, effectively shortening the duration of training sessions. This could potentially lead to an improved BCI user experience, enabling mainstream adoption of such systems in the clinical setting.

5 CONCLUSION

In this work we demonstrate how a data-efficient EEG-based online BCI system can be deployed end-to-end to compete in the Cybathlon BCI race. The main advantage of our system compared to others from previous literature is that it requires a volume of training data one order of magnitude smaller while retaining a competitive accuracy, essentially reducing the amount of time needed for data acquisition prior to live BCI usage. In addition, we present methods to troubleshoot an EEG-based BCI system by examining CSP bipolar plots, pre-CSP and post-CSP filtered trials. We also demonstrate good practices to avoid over-optimistic offline results by using a block-wise cross-validation discarding a number of trials between the training and test blocks, effectively eliminating correlated trials.

The system we implemented was based on temporal filtering using band-pass filtering and spatial filtering using Common Spatial patterns to extract log-variance features to fit a linear classifier (Logistic Regression). We focused on the identification of sensorimotor rhythms associated with motor imageries and 4 different motor imageries were tested (right hand movement, left hand movement, both feet movement, and abdominal contraction). Our results showed comparable results in offline and online accuracy for 2-commands control, obtaining superior results on the online mode than on the offline mode, but a drop in performance for 3-class online decoding with respect to offline decoding. We were able to improve 3-command online control by using a two-step binary classification strategy.

We also observed great variability in performance across subjects and sessions, which suggests that multiple sessions would be ideally needed to identify which motor imageries can be best controlled by each subject, so that training can be focused on those specific ones. The data-efficient nature of our approach makes it well suited for fast identification of optimal motor-imageries per subject. Finally, the Cybathlon testing task and performance conditions of our system align it with contemporary assistive technologies' standards and afford it the potential to become a good benchmark and a promising tool for the advancement of non-invasive BCI research and applications.

REFERENCES

Dziemian, S., Abbott, W. W., and Faisal, A. A. (2016). Gaze-based teleprosthetic enables intuitive continuous control of complex robot arm use: Writing drawing. In *2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pages 1277–1282.

Fan, R., Chang, K., Hsieh, C., Wang, X., and Lin, C. (2008). LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874.

Ferrante, A., Gavriel, C., and Faisal, A. A. (2015). Data-efficient hand motor imagery decoding in EEG-BCI by using morlet wavelets and common spatial pattern algorithms. In *2015 7th International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 948–951.

Friedrich, E. V., Scherer, R., and Neuper, C. (2013). Long-term evaluation of a 4-class imagery-based braincomputer interface. *Clinical Neurophysiology*, 124(5):916 – 927.

Grant, M. C. and Boyd, S. P. (2008). Graph implementations for nonsmooth convex programs. In *Recent Advances in Learning and Control. Lecture Notes in Control and Information Sciences*, volume 371, pages 95–110, London. Springer.

Grosse-Wentrup, M. and Buss, M. (2008). Multiclass common spatial patterns and information theoretic feature extraction. *IEEE Transactions on Biomedical Engineering*, 55(8):1991–2000.

Holz, E. M., Botrel, L., Kaufmann, T., and Kübler, A. (2015). Long-term independent brain-computer interface home use improves quality of life of a patient in the locked-in state: A case study. *Archives of Physical Medicine and Rehabilitation*, 96(3):S16–S26.

Koles, Z. (1991). The quantitative extraction and topographic mapping of the abnormal components in the clinical EEG. *Electroencephalography and Clinical Neurophysiology*, 79(6):440 – 447.

Kothe, C. A. and Makeig, S. (2013). BCILAB: a platform for braincomputer interface development. *Journal of Neural Engineering*, 10(5):056014.

Kübler, A., Neumann, N., Kaiser, J., Kotchoubey, B., Hinterberger, T., and Birbaumer, N. P. (2001). Brain-computer communication: Self-regulation of slow cortical potentials for verbal communication. *Archives of Physical Medicine and Rehabilitation*, 82(11):1533–1539.

Lu, N., Li, T., Ren, X., and Miao, H. (2017). A deep learning scheme for motor imagery classification based on restricted boltzmann machines. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(6):566–576.

Maimon-Mor, R. O., Fernandez-Quesada, J., Zito, G. A., Konnaris, C., Dziemian, S., and Faisal, A. A. (2017). Towards free 3D end-point control for robotic-assisted human reaching using binocular eye tracking. In *2017 International Conference on Rehabilitation Robotics (ICORR)*, pages 1049–1054.

Makin, T. R., de Vignemont, F., and Faisal, A. A. (2017). Neurocognitive barriers to the embodiment of technology. *Nature Biomedical Engineering*, 1:0014 EP.

Milln, J. R., Renkens, F., Mourio, J., and Gerstner, W. (2004). Brain-actuated interaction. *Artificial Intelligence*, 159(1):241 – 259.

Müller-Putz, G. R., Scherer, R., Brunner, C., Leeb, R., and Pfurtscheller, G. (2008). Better than random: a closer look on BCI results. *International Journal of Bioelectromagnetism*, pages 52–55.

Neuper, C., Wrtz, M., and Pfurtscheller, G. (2006). ERD/ERS patterns reflecting sensorimotor activation and deactivation. In Neuper, C. and Klimesch, W., editors, *Event-Related Dynamics of Brain Oscillations*, volume 159 of *Progress in Brain Research*, pages 211 – 222. Elsevier.

Noronha, B., Dziemian, S., Zito, G. A., Konnaris, C., and Faisal, A. A. (2017). Wink to grasp - comparing eye, voice and EMG gesture control of grasp with soft-robotic gloves. In *2017 International Conference on Rehabilitation Robotics (ICORR)*, pages 1043–1048.

Obermaier, B., Neuper, C., Guger, C., and Pfurtscheller, G. (2001). Information transfer rate in a five-classes brain-computer interface. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 9(3):283–288.

Ortega, P., Colas, C., and Faisal, A. A. (2018a). Compact convolutional neural networks for multi-class, personalised, closed-loop EEG-BCI. In *2018 IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob), 7th IEEE International Conference (forthcoming)*.

Ortega, P., Colas, C., and Faisal, A. A. (2018b). Convolutional neural network, personalised, closed-loop brain-computer interfaces for multi-way control mode switching in real-time. *bioRxiv*.

Ramoser, H., Muller-Gerking, J., and Pfurtscheller, G. (2000). Optimal spatial filtering of single trial EEG during imagined hand movement. *IEEE Transactions on Rehabilitation Engineering*, 8(4):441–446.

Riener, R. and Seward, L. J. (2014). Cybathlon 2016. In *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2792–2794.

Schwarz, A., Steyrl, D., and Mller-Putz, G. R. (2016). Brain-computer interface adaptation for an end user to compete in the Cybathlon. In *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 001803–001808.

Tabar, Y. R. and Halici, U. (2017). A novel deep learning approach for classification of EEG motor imagery signals. *Journal of Neural Engineering*, 14(1):016003.

Tomioka, R., Aihara, K., and Müller, K. (2007). Logistic regression for single trial EEG classification. In Schölkopf, B., Platt, J. C., and Hoffman, T., editors, *Advances in Neural Information Processing Systems 19*, pages 1377–1384. MIT Press.

Tostado, P. M., Abbott, W. W., and Faisal, A. A. (2016). 3D gaze cursor: Continuous calibration and end-point grasp control of robotic actuators. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3295–3300.

Yang, H., Sakhavi, S., Ang, K. K., and Guan, C. (2015). On the use of convolutional neural networks and augmented CSP features for multi-class motor imagery of EEG signals classification. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 2620–2623.

Zhang L., He W., H. C. and Wang, P. (2010). Improving mental task classification by adding high frequency band information. *Journal of Medical Systems*, 34:51 – 60.