

ACCEPTED MANUSCRIPT

Adaptive feature extraction in EEG-based motor imagery BCI: tracking mental fatigue

To cite this article before publication: Upasana Talukdar *et al* 2019 *J. Neural Eng.* in press <https://doi.org/10.1088/1741-2552/ab53f1>

Manuscript version: Accepted Manuscript

Accepted Manuscript is “the version of the article accepted for publication including all changes made as a result of the peer review process, and which may also include the addition to the article by IOP Publishing of a header, an article ID, a cover sheet and/or an ‘Accepted Manuscript’ watermark, but excluding any other editing, typesetting or other changes made by IOP Publishing and/or its licensors”

This Accepted Manuscript is © 2019 IOP Publishing Ltd.

During the embargo period (the 12 month period from the publication of the Version of Record of this article), the Accepted Manuscript is fully protected by copyright and cannot be reused or reposted elsewhere.

As the Version of Record of this article is going to be / has been published on a subscription basis, this Accepted Manuscript is available for reuse under a CC BY-NC-ND 3.0 licence after the 12 month embargo period.

After the embargo period, everyone is permitted to use copy and redistribute this article for non-commercial purposes only, provided that they adhere to all the terms of the licence <https://creativecommons.org/licenses/by-nc-nd/3.0>

Although reasonable endeavours have been taken to obtain all necessary permissions from third parties to include their copyrighted content within this article, their full citation and copyright line may not be present in this Accepted Manuscript version. Before using any content from this article, please refer to the Version of Record on IOPscience once published for full citation and copyright details, as permissions will likely be required. All third party content is fully copyright protected, unless specifically stated otherwise in the figure caption in the Version of Record.

View the [article online](#) for updates and enhancements.

Adaptive Feature Extraction in EEG-based Motor Imagery BCI: Tracking Mental Fatigue

Upasana Talukdar¹, Shyamanta M Hazarika² and John Q Gan³

¹Biomimetic & Cognitive Robotics Lab, Department of Computer Science & Engineering, Tezpur University, Tezpur, India.

²Mechatronics & Robotics Lab, Department of Mechanical Engineering, Indian Institute of Technology, Guwahati, India

³School of Computer Science & Electronic Engineering, University of Essex, United Kingdom

E-mail: upat123@tezu.ernet.in, upasanat123@gmail.com

Abstract. *Objective:* Electroencephalogram (EEG) signals are non-stationary. This could be due to internal fluctuation of brain states such as fatigue, frustration, etc. This necessitates the development of adaptive Brain-Computer Interfaces (BCI) whose performance does not deteriorate significantly with the adversary change in the cognitive state. In this paper, we put forward an unsupervised adaptive scheme to adapt the feature extractor of motor imagery (MI) BCIs by tracking the fatigue level of the user. *Approach:* Eleven subjects participated in the study during which they accomplished MI tasks while self-reporting their perceived levels of mental fatigue. Out of the 11 subjects, only 6 completed the whole experiment, while the others quit in the middle because of experiencing high fatigue. The adaptive feature extractor is attained through the adaptation of the common spatial patterns (CSP), one of the most popular feature extraction algorithms in EEG-based BCIs. The proposed method was analyzed in two ways: offline and in near real-time. The separability of the MI EEG features extracted by the proposed adaptive CSP (ADCSP) has been compared with that by the conventional CSP (C-CSP) and another CSP based adaptive method (ACSP) in terms of: Davies Bouldin Index (DBI), Fisher Score (FS) and Dunn's Index (DI). *Results:* Experimental results show significant improvement in the separability of MI EEG features extracted by ADCSP as compared to that by C-CSP and ACSP. *Significance:* Collectively, the results of the experiments in this study suggest that adapting CSP based on mental fatigue can improve the class separability of MI EEG features.

Keywords: Motor Imagery, EEG, BCI, Common Spatial Patterns, Adaptation, Mental Fatigue

1. Introduction

Brain-Computer Interface (BCI) is a communication methodology that translates the intent of a user measured through brain signals into control commands for devices such as neuroprosthesis or computer applications. Even though there are a number of modalities of capturing brain signals for BCIs, non-invasive electroencephalogram (EEG) for motor imagery (MI) remains one of the most popular. Imagination of movements known as motor imagery (MI) can induce changes in EEG signals and has attracted the attention of the researchers as a basis of BCIs. Deterioration of motor function due to spinal cord injuries, accident or stroke leads to motor disabilities and to this end EEG-based MI BCI provides promising solutions.

A lot of issues have been reported on the reliability of MI tasks and accompanying changes in EEG signals [1]. One such predicament is the non-stationary nature of EEG signals. One of the major causes of such non-stationarity is the change in the cognitive state of the users like attention, concentration, workload, fatigue, etc. [2]. In EEG-based MI BCIs, significant mental effort is required to concentrate on MI tasks. Repeated accomplishment of MI tasks for a prolonged period can be tedious. Subjects may experience loss of attention, drowsiness or fatigue. This change in the cognitive state may lead to signal fluctuations or variations/shifts in the statistics of EEG signals from trial to trial and from day to day. This may cause degradation in the MI EEG features, declining their class separability.

Different methods have been developed to address the non-stationary nature of EEG signals. These methods focused on the adaptation of feature extractors [3–9], classifiers [10–12] or on post-processing level [13]. Adaptation of feature extractors has been attained in the literature mostly through adaptation of common spatial patterns (CSP) [4, 5, 8, 9], one of the most popular methods of extracting features in MI BCI. It is a two-class supervised and subject specific method that constructs spatial filters based on training data which maximize the variance of one class while minimizing the variance of the other [14]. However, due to non-stationary nature of EEG, the CSP computed on the training data may not be optimal for the evaluation data, which in turn may lead to the significant shift in the feature space between the training and the evaluation sessions. Most of the adaptation of CSPs

have been attained in a supervised manner by estimating the target data's class label first and adapting the covariance matrix of the estimated class [3–6]. Talukdar et al. [15] adapted CSP through active learning which is a semi-supervised approach. Song et al. [7] proposed an unsupervised method of adapting CSP without estimating the target data class labels (named as ACSP) and portrayed its benefits over the existing supervised methods. Although there exist many methods for adapting CSP to address the non-stationary nature of EEG, none of them adapt it by taking the cognitive state into account.

Myrden et al. [2] have shown the effect of mental states on BCI performance. Recent work [16] has reported the inter-relationship between MI and mental fatigue, i.e., a prolonged session of MI induces mental fatigue which in turn decreases the separability of MI EEG features. In such cases, there is an ample necessity for continuously monitoring the change in the user's cognitive state to decide when to initiate adaptation in a proper manner. Myrden et al. [17] presented an adaptive BCI based on fatigue, frustration and attention level where adaptation was achieved by resampling the training set and retraining the classifier. Their analysis was made offline.

This study presents a novel methodology to adapt CSP based on the fatigue state of the user (abbreviated as ADCSP in this paper). CSP is updated by updating the covariance matrices of each class without estimating the target data's class labels. Mental fatigue level has been used as a metric to activate the adaptation. The growth of mental fatigue was tracked using the Kernel Partial Least Square method [18]. Talukdar et al. [16] have presented a detailed formulation of fatigue analysis. The covariance matrices of CSPs are updated using adaptive weights. The adaptation is made both offline and in near real-time and is activated during high fatigue level. Near real-time processing is when the speed/rate is important, but processing time in minutes can be accepted in place of seconds. It is a processing that is slightly slower than the real time processing. No classification or estimation of the target data class labels is needed in the adaptive scheme. The effectiveness of the ADCSP has been evaluated in comparison to that of the conventional CSP (abbreviated as C-CSP in this study) and ACSP [7] in terms of three separability metrics: Davies Bouldin Index (DBI) [19], Dunn's Index (DI) [20] and

Fisher Score (FS) [21]. ACSP has been chosen for comparison since it is also an unsupervised adaptive scheme.

The rest of the paper is organized as follows: Section 2 describes the materials and methods used in the study. Section 3 presents the experimental findings while Section 4 discusses the results. Finally, Section 5 concludes the paper.

2. Materials and Methods

2.1. Acquisition of Motor Imagery - Mental Fatigue EEG Data

EEG data was collected from 11 individuals at the University of Essex, England. Before the experiment, the subjects gave their informed consent using a form approved by the Ethics Committee of University of Essex. The consent form contains information about the EEG recording procedure and the purpose of the EEG experiment.

The subjects completed two pre-test self-report measures: Visual Analogue Scale - Fatigue (VAS-F) [22] and Chalder Fatigue Scale (CFS) [23]. Thereafter, the subjects performed 4 different MI tasks: a. left hand movement (Class 1) b. right hand movement (Class 2) c. both feet movement (Class 3), and d. tongue movement (Class 4) for one complete session. A session consists of 8 runs. Each run lasts 12 minutes and consists of 80 trials. Each trial begins with a fixation cross that appears on the computer screen along with a short acoustic warning tone. It is followed by a cue that appears either left, right, down or up of the fixation cross indicating the subjects for the imagination of left hand, right hand, both feet, or tongue movement respectively. The subjects accomplished the desired task until the cue and the fixation cross disappeared from the screen at $t = 6$ sec. There was a break for 3 secs between trials. No extra break was there between runs. The experimental paradigm is illustrated in Figure 1.

The fatigued state at the end of each run was rated by using a "fatigue scale" - a subjective scale with a value from 1 to 5 that extends between two extremes (1 = "Least fatigued" and 5 = "Most fatigued"). Subjects chose a number along the scale to express the fatigue they were experiencing. This was taken as the fatigue score of that particular run. The subjects completed the post-test self-report using a. VAS-F and b. CFS. Five out of the 11 subjects quit in the middle of the experiment because of high fatigue level.

EEG data was recorded using Biosemi Active Two System. Sixty four EEG channels were used to record the data following the 10-20 international montage system. The sampling frequency was 256 Hz. EAWICA was used to remove the artefacts

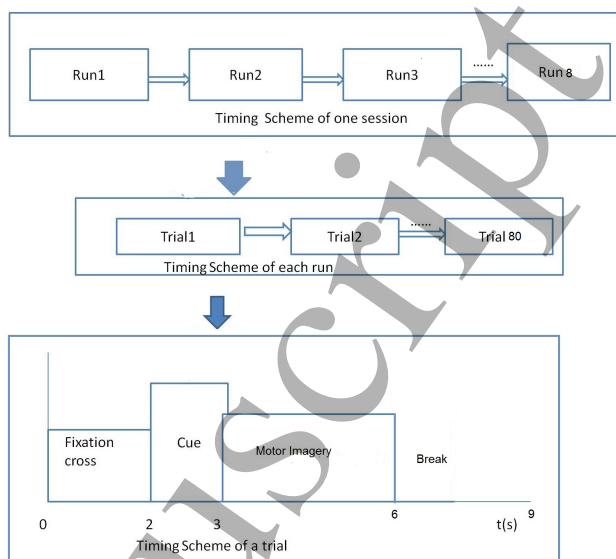


Figure 1: The experimental paradigm

[24]. Thereafter, EEG data were low-pass filtered with 40 Hz as the cut off frequency and then subtracted by common average reference. The EEG data from instant $t=0$ to $t=6$ were then analysed.

2.2. Monitoring the Growth of Mental Fatigue during MI

Monitoring of the growth of mental fatigue through EEG has been carried out on the 6 subjects who completed all the 8 runs of the experiment. Spectral power and spectral entropy were used as features and were computed in four frequency bands, δ (0.1-3.5 Hz), θ (4-7.5 Hz), α (8-12 Hz) and β (13-30 Hz) from five different areas of the scalp: frontal (F1, F3, F5, F7, Fz, F2, F4, F6, AFz, AF3, AF4 and FPz), parietal (P1, P3, P5, P7, Pz, P2, P4, P6, POz, PO3, PO4 and CPz), temporal (T8, T8, TP8, FT7, T7, and TP7), central (C1, C3, C5, Cz, C2, C4, C6) and occipital (O1, O2, Oz). The features that exhibit a significant increase during the last run as compared to that during the first run were identified as optimum features and selected for monitoring the growth of mental fatigue. The Kernel Partial Least Square (KPLS) algorithm [18] was used for monitoring the growth of fatigue which is a non-linear regression method based on the projection of input (explanatory) variables to the latent vectors (components) [18]. The KPLS algorithm took as input the optimal feature vector and the two class labels, low fatigue (-1) and high fatigue (+1) and predicts KPLS scores as output. The KPLS scores are the projection of the observations onto the KPLS regression coefficients. The KPLS score of each trial can be interpreted as "fatigue score" of that trial. The

fatigue score has been referred as f in this study. Since, -1 and +1 are the two classes, -1 for low fatigue state and +1 for high fatigue state, KPLS scores lies in the range of [-1 1]. Negative scores represents low fatigue state and positive scores as high fatigue state. The mean of the KPLS scores for each run was then computed which was interpreted as fatigue score of that particular run. The KPLS model was validated with respect to the subjective scores obtained through fatigue scale.

On the basis of the fatigue score obtained through KPLS, each run was classified as either low or high fatigue level using the K-means algorithm. MI EEG feature separability was estimated in terms of DBI, DI, and FS during each level of fatigue. A detailed formulation of fatigue analysis can be found in Talukdar et al. [16].

2.3. Adaptive Common Spatial Patterns (ADCSP)

CSP is the most widely used method for feature extraction in MI BCI. It uses a spatial filter to discriminate the spatial patterns for two classes. It takes as input the amplitudes of N channel EEG recorded at T consecutive time points and produces as output an ordered list of distinctive spatial patterns. A few discriminative patterns are substantial for differentiating two classes, and are referred to as CSPs. A detailed formulations of CSP can be found in [14].

CSP computation is based on the covariance matrix of each class. This study proposes a new unsupervised adaptive method named ADCSP to update CSP during high fatigue level. It updates the covariance matrices of each class by adding regularisation parameters without estimating the target data's class labels. ADCSP is obtained through Eq 1 and Eq 2 as follows.

$$C_1 = ((1 - \gamma) * R.a) + \frac{\gamma}{N} * trace(R.a) * I \quad (1)$$

$$C_2 = ((1 - \gamma) * R.b) + \frac{\gamma}{N} * trace(R.b) * I \quad (2)$$

where C_1 and C_2 are the updated covariance matrices of class 1 and class 2 respectively, β and γ are two regularization parameters, I is an identity matrix of $N \times N$ and $R.a$, $R.b$ are defined as follows:

$$R.a = \frac{((1 - \beta) * C_1) + ((1 - K_1) * pVect2 * \beta)}{N.a + 1}; \quad (3)$$

$$R.b = \frac{((1 - \beta) * C_2) + ((1 - K_2) * pVect2 * \beta)}{N.b + 1}; \quad (4)$$

where C_1 and C_2 are the current covariance matrices of class 1 and class 2 respectively, $pVect2$ is the covariance matrix of testing data with unknown class labels, K_1 and K_2 are the Kullback-Leibler divergence (KL)

between the testing and training data of class 1 and class 2 respectively. $N.a$ and $N.b$ are the number of training trials of class 1 and class 2 respectively.

KL measures how the probability distribution of target data diverges from the probability distribution of testing data and is computed as follows:

$$K_y(p_{new}, p_{train}) = 0.5(trace(C_y^{-1} C_{y,new}) - \log[\frac{det(C_{y,new})}{det(C_y)}]) - M \quad (5)$$

where, p_{new} and p_{train} are the probability distributions of new target EEG trial and training data respectively, det represents determinant of matrix, $C_{y,new}$ and C_y are the covariance matrices of new EEG data and training data respectively, M is the dimension of covariance matrix and $y \in \{1,2\}$.

Since, $K_y(p_{new}, p_{train}) \neq K_y(p_{train}, p_{new})$, K_y can be symmetrized as follows [7]

$$K_y = 0.5(K_y(p_{new}, p_{train}) + K_y(p_{train}, p_{new})) \quad (6)$$

The idea of using β and γ is derived from the work of Lu et al. [25]. Large β shrinks the current covariance matrix towards the testing covariance matrix while large γ shrinks the current covariance matrix towards the identity matrix to counter the bias due to small training set [25]. The values for β was searched in the range of [0,1] and [1,10] while the values for γ was searched in the range of [0,1] for all subjects. Two different ranges were used for β because in case the high fatigue testing data is quite different from the training data, larger β value works. The value of β and γ within the aforesaid ranges that give the best validation performance (in terms of DBI, DI and FS) is chosen for analysis. However, unlike Lu et al. [25], in addition to the two regularisation parameters β and γ , we also used KL between training and testing data to update the spatial filter. KL has been used to update the covariance matrices because based on the KL we can estimate the difference between the training and testing data and update the covariance matrix of each class accordingly.

2.3.1. Implementation of ADCSP: ADCSP can be implemented for both offline or in near real-time analysis. The adaptation in near real-time is done on trial basis while offline adaptation is done on all the runs that come under high fatigue level. In Fig 2 we show when the adaptation is activated. The EEG data were divided into training and evaluation data. The training data (T) comprises of first two runs while the evaluation data (E) comprises of runs 3-8 for the offline analysis and runs 3-7 runs for near real-time analysis. For offline analysis, evaluation data were clustered into

two levels based on the fatigue score using the K-means clustering: low fatigue (L) and high fatigue (H). No adaptation of the spatial filter was carried out during low fatigue level, but adaptation was performed during high fatigue level. While for near real-time analysis, the trial with positive fatigue score is classified as high fatigue trial and negative fatigue score as low fatigue trials. Adaptation is activated when a high fatigue trial is found.

The detailed implementation of ADCSP for offline analysis (named as ADCSP-I) is shown in Algorithm 1 and implementation of ADCSP in near real-time (named as ADCSP-II) is shown in Algorithm 2.

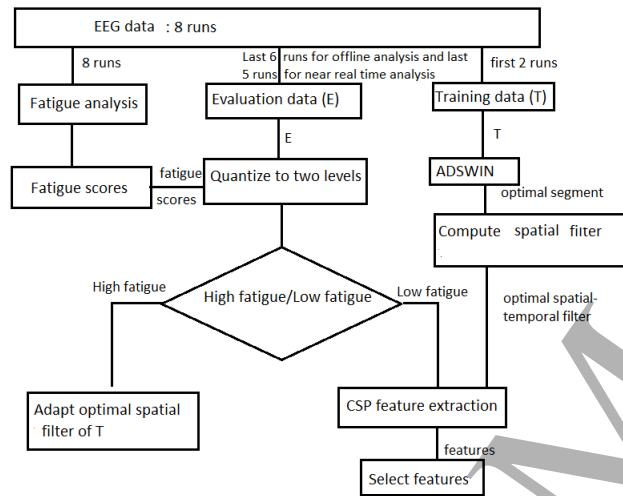


Figure 2: Conceptual overview of when to adapt. The adaptation is activated during high fatigue level, no adaptation is carried out during low fatigue level

- (i) **ADCSP-I**: ADCSP-I relies on fatigue score (f) of a particular run based on which the fatigue level of the user during that run can be estimated. f is estimated as illustrated in Section 2.2. Hence, ADCSP-I starts by performing fatigue analysis on the evaluation data to compute f of each run. ADCSP-I then categorises the runs of the evaluation data into two levels of fatigue: low fatigue and high fatigue based on the fatigue score using the K-means clustering algorithm. The adaptation is done on the runs that come under high fatigue level. The detailed algorithm is shown in Algorithm 1.

The fatigue score f is calculated by the following function namely `Fatigue_Analysis`.

- *Fatigue_Analysis* (i, E): For any run i of evaluation data E , this module performs the fatigue analysis of a subject during the i th run and compute the fatigue score f . Based on f , the fatigue level during i th run is estimated

Algorithm 1: ADCSP-I - ADCSP for offline analysis

Input: Y =EEG data with the 8 runs, T_{csp} =training set to compute CSP projection matrix (first two runs), E =evaluation data (3rd to 8th runs), $N1$ = no. of runs in E
Output: W_o = CSP projection matrix,
 W_{new} =updated CSP projection matrix

```

r=1, i=3;
Low_Fatigue = [], High_Fatigue = [];
Compute  $C_1$  and  $C_2$  on  $T_{csp}$ ;
Compute  $W_o$  on  $T_{csp}$ ;
while  $r \leq N1$  do
    level.fatigue (r) = Fatigue_Analysis (i,E);
    if level.fatigue (r) == L then
        Low_Fatigue = [ Low_Fatigue, i ];
        i ← i + 1
        r ← r + 1
    end
    else if level.fatigue (r) == H then
        High_Fatigue = [ High_Fatigue, i ];
        i ← i + 1
        r ← r + 1
    end
end
for all runs ∈ Low_Fatigue
    Extract features on  $W_o$ ;
    Select features;
end
for all runs ∈ High_Fatigue
    Compute  $K_1$  and  $K_2$ ;
    Update  $C_1$  and  $C_2$  as in Eq. 1 and Eq. 2 respectively;
    Compute  $W_{new}$  on the updated  $C_1$  and  $C_2$ ;
    Extract features with  $W_{new}$  and select features;
end

```

function Fatigue_Analysis(i, E)

```

Perform fatigue analysis on  $i$ th run of  $E$  to compute
fatigue score ( $f$ );
Estimate the level of fatigue during the  $i$ th run of  $E$ 
using K-means algorithm;
if low fatigue level then
    | level.fatigue ← L
end
else if high fatigue level then
    | level.fatigue ← H
end
return(level.fatigue);
end function

```

using the K-means clustering algorithm.

- (ii) **ADCSP-II**: ADCSP-II adapts the CSPs in near real-time on trial basis. To explain how this algorithm works, some other parameters, such as the length of the adaptation window and the initial time when the adaptation should start are used and explained as follows:

- (a) *Fatigue score* (f) : This gives the score of fatigue level of the user during a particular trial. The fatigue score is estimated through EEG as illustrated in Section 2.2. The

Algorithm 2: ADCSP-II - ADCSP in near real-time

Input: Y=EEG data with the 8 runs, T_f =training set to estimate fatigue score (first and last runs), T_{csp} =training set to compute CSP projection matrix (first two runs), E=evaluation data (3rd to 7th runs), $i=1$, $t=0$, β , γ , p = window length, n_t = number of contiguous high fatigue trials

Output: W_{new} =final CSP projection matrix

```

t=p;
Compute  $C_1$  and  $C_2$  on  $T_{csp}$ ;
Compute  $W_o$  on  $T_{csp}$ ;
foreach trial  $i \in E$  do
  level_fatigue = Fatigue_Analysis (i,E);
  if level_fatigue == L then
    Extract features on  $W_o$  of ith trial;
    Select features;
     $i \leftarrow i + 1$ 
  end
  else if level_fatigue == H then
    Extract features on  $W_o$  of ith trial;
     $i \leftarrow i + 1$ 
    ( $i, W_o, t$ )= Adapt_CSP(i, E,  $W_o, t, C_1, C_2$ );
     $i \leftarrow i + 1$ 
     $t \leftarrow t + p$ 
  end
end

```

```

function Fatigue_Analysis(i, E)
Perform fatigue analysis on ith trial of E to compute
fatigue score (f);
Estimate the level of fatigue during the ith trial of E
using K-means algorithm;
if low fatigue level then
  level_fatigue  $\leftarrow$  L
end
else if high fatigue level then
  level_fatigue  $\leftarrow$  H
end
return(level_fatigue);
end function

```

```

function Adapt_CSP(i, E,  $W_o, t, C_1, C_2$ )
Compute  $K_1$  and  $K_2$ ;
Update  $C_1$  and  $C_2$  as in Eq. 1 and Eq. 2 respectively;
Compute  $W_{new}$  on the updated  $C_1$  and  $C_2$ ;
Extract features with  $W_{new}$  of ith trial and select
features;
( $i, W_o, t$ ) = Check ( $W_{new}, W_o, E, t, i, C_1, C_2$ );
return( $i, W_o, t$ );
end function

```

```

function Check ( $W_{new}, W_o, E, j, t, i$ )
Evaluate the criterion as in Eq. 7;
 $i=i+1$ ;
if the criterion is satisfied then
  Adaptation stops;
  ( $i, W_o, t$ ) = After_adapt (i, E, t,  $W_{new}$ );
  return( $i, W_o, t$ );
end
else
   $W_o = W_{new}$ ;
  ( $i, W_o, t$ ) = Adapt_CSP(i, E,  $W_o, t, C_1, C_2$ );
end
end function

```

```

function After_adapt(i, E, t,  $W_{new}$ )
 $W_o = W_{new}$ ;
while  $i \leq t$  do
   $n2 = n_t + i$ ;
  foreach  $i <= n2$  do
    level_fatigue = Fatigue_Analysis (i,E);
    Extract features of ith trial with  $W_o$  and select
    features;
  end
  if level_fatigue of all theses n trials == H then
     $i \leftarrow n2 + 1$ 
    ( $i, W_o, t$ ) = Adapt_CSP(i, E,  $W_o, t, C_1, C_2$ );
     $W_o \leftarrow W_{new}$ ;
  end
  else
     $i \leftarrow n2 + 1$ 
  end
end
return( $i, W_o, t$ )
end function

```

KPLS score of each trial was computed and interpreted as the fatigue score of that particular trial.

- (b) *Length of the adaptation window:* The length of the adaptation window is the time period when the adaptation starts till it reaches the stopping criteria as defined in Eq. 7

$$|f_2(k) - f_2(k-1)| < \phi, \quad (7)$$

where, ϕ is a pre-determined positive constant, k is the k th iteration and

$$f_2(k) = \sum_{i=1}^N \left| \frac{(W_i^{(k)})^T S_L^{(k)} (W_i^{(k)})}{(W_i^{(k)})^T S_R^{(k)} (W_i^{(k)})} \right| \quad (8)$$

where $W_i^{(k)}$ is the i th column of the spatial

filter, $S_L^{(k)}$ and $S_R^{(k)}$ are computed as follows

$$S_L^{(k)} = C_1 - C_2, S_R^{(k)} = C_1 + C_2 \quad (9)$$

with C_1 and C_2 being the covariance matrices of Class 1 and Class 2 respectively.

Eq. 8 computes the sum of Rayleigh coefficients. Since the advantage of CSP feature extraction has been demonstrated in the paradigm of maximization of Rayleigh coefficients, the improvement in the sum of Rayleigh coefficients leads to better class separability of the features [26]. Hence, this study uses this concept as the stopping criterion of the adaptive scheme.

- (c) *Initial time (init)* : Is the initial time when the adaptation starts which is updated based

on the fatigue score estimated through EEG. When fatigue level is high, we hypothesize that adapting the trial with high fatigue level would yield better performance and hence the adaptation starts. The updating of *init* is described in Algorithm 2.

Both ADCSP-I and ADCSP-II rely on fatigue score as well as values of β and γ . ADCSP-II depends on four major modules namely *Fatigue_Analysis*, *Adapt_CSP*, *Check*, and *After_adapt*. The evaluation data E is divided into different windows of length 'p' so as to perform the monitoring of the fatigue level efficiently. Hence, ADCSP-II also relies on the values of 'p' and ' n_t '. n_t is a variable used to denote the number of contiguous high fatigue trials. Different values of p (p=50,100,200,300, 400) and n_t ($n_t \in [1\ 10]$ is an integer) have been tested and the best one (p=200, $n_t=10$) were used for the study. The range of p and n_t is same for all the subjects.

- *Fatigue_Analysis (i, E)*: Unlike *Fatigue_Analysis* of ADCSP-I, this module computes the fatigue score of each trial. For any trial i of evaluation data E , the module performs the fatigue analysis of a subject during the i th trial and compute the fatigue score f . Based on f , the level of fatigue during i th trial is estimated using K-means algorithm.
- *Adapt_CSP (i, E, W_o , t)*: This module updates the existing CSP projection matrix (W_o) to compute new CSP projection matrix (W_{new}) considering the i th trial of the evaluation data E .
- *Check (W_{new} , W_o , E , t , i)*: The stopping criterion of the adaptive process is checked in this module.
- *After_adapt(i, E , t , W_{new})*: This module is activated when adaptation stops. It checks for ' n_t ' contiguous high fatigue trials. If ' n_t ' contiguous high fatigue trials are found, then the module starts adapting the subsequent trials, otherwise no adaptation is performed. This process continues for a specific window length 'p'.

2.4. MI EEG Feature Separability Evaluation

Separability of extracted features can be computed directly by using certain metrics like DBI, FS, etc. or indirectly in terms of classification accuracy [27].

In object classification, one needs to predict the class of an unseen object by identifying the class-distribution pattern. This is only possible when the diction "*similar objects tend to cluster together*" is true [28]. Due to the high dimensionality of the dataset, it cannot be immediately visualized to identify the class-distribution [28]. Hence, using a method that can give information on the separability of

features without using multiple sets of computationally expensive classifiers would be advantageous [28]. And there comes the role of separability indices. Since classification performance depends on the separability of the classes, it can be deduced that the higher separability of features leads to the increase in classification performance [27].

This study uses three such separability indices, DBI, DI and FS to evaluate the separability of features. Lower value of DBI and higher value of DI and FS indicates higher MI EEG feature separability. Unlike using classifiers, no pre-training is required when using separability indices. They are independent of the number of groupings and the grouping algorithm used [29] and hence are simple, feasible and time saving [30]. Friedman statistical test has been carried out to examine statistical significance. The significant difference between the separability of MI EEG features extracted with C-CSP and ADCSP and that with ACSP and ADCSP is shown by \checkmark .

3. Experimental Results

3.1. Monitoring the Growth of Mental Fatigue during MI

The growth of fatigue was tracked on the 6 subjects who completed the whole experiment using the KPLS algorithm that consists of two key steps: i. KPLS model selection and ii. KPLS model prediction. KPLS model is subject-specific. During the KPLS model selection, the optimal number of KPLS components (KPLS latent vectors) for each subject that provides the maximum classification accuracy with Linear Discriminant Analysis (LDA) was identified. For estimating the optimal number of components, the first and the last runs were used as training as well as testing data, with the first run as active state and the last run as fatigue state. The KPLS components were evaluated in the range of 1 to 10. During KPLS model prediction, the KPLS scores of each trial of all the runs were predicted. In our work [16], the analysis of EEG spectral power and spectral entropy from different frequency bands in different areas of the scalp showed that spectral power increases during the last run as compared to that of the first run. δ , θ and α power from frontal region, α power from parietal lobe, δ , θ and α power from temporal lobe and θ power from occipital lobe has been considered as optimal features since they show significant increase during the last run as compared to that of the first run; while β power from all the lobes shows insignificant change between the first and last runs. The KPLS score of each trial can be interpreted as the fatigue score of that trial. The mean of the KPLS scores of each run was then computed and interpreted as the fatigue score of that

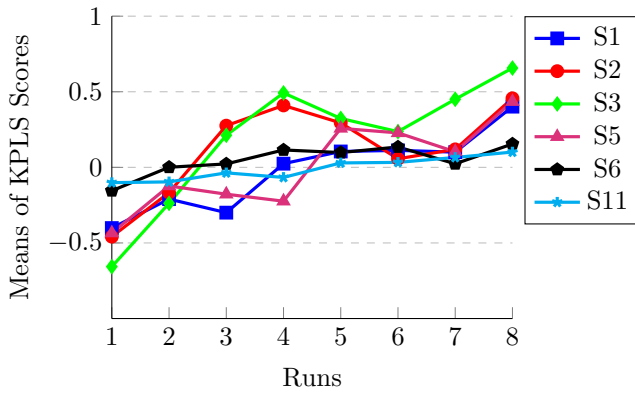


Figure 3: Means of KPLS scores for each run with 6 subjects

Table 1: Correlation between the trends of KPLS scores and subjective scores

ID	Runs 1-8		Runs 2-7	
	Corr. Coeff	p-value	Corr. Coeff	p-value
S1	0.9825	$1.32e^{-5}$ ✓	0.92	0.025 ✓
S2	0.891065	0.001418 ✓	0.66	0.21 ×
S3	0.917	0.0004 ✓	0.88	0.048 ✓
S5	0.960346	$0.29e^{-5}$ ✓	0.95	0.005 ✓
S6	0.89	0.0032 ✓	0.82	0.065 ×
S11	0	1 ×	0	1 ×

particular run. For each of the 6 subjects, the fatigue scores were plotted in a graph as shown in Figure 3.

The KPLS model was validated with respect to the subjective scores obtained through fatigue scale. Table 1 shows the correlation between the KPLS model and the subjective scores. The first column shows the Subject id, 2nd and 3rd column shows the correlation between KPLS model and the subjective scores taking runs 1-8 and 2-7 into account respectively. This is because, runs 1 and 8 were used for training the KPLS model and hence training may find some combination of input features to maximise the separability between runs 1 and 8. Hence, correlation was also found for runs 2-7.

The result shows that five subjects showed a strong correlation (> 0.5) between the KPLS scores and the subjective scores. In case of Subject 11 no correlation was found. Hence the rest of the analysis was carried out on the 5 subjects. Talukdar et. al. [16] has detailed formulation for fatigue analysis.

Based on these computed fatigue scores, each run of the evaluation data was then categorized as either low or high fatigue state by using the K-means algorithm [16]. Table 2 shows the runs categorized as low and high fatigue level based on the computed fatigue score. The first column shows the subject id while the second and the third columns show the runs

Table 2: Runs categorized as low or high fatigue level

Subject id	Low fatigue	High fatigue
S1	3,4	5,6,7,8
S2	6,7	3,4,5,8
S3	3,5,6	4,7,8
S5	3,4,7	5,6,8
S6	3,7	4,5,6,8

categorized as low and high fatigue level respectively.

3.2. Evaluation of the Separability of MI EEG Features

3.2.1. Feature Extraction and Selection To evaluate MI EEG feature separability, this study performs optimal spatio-temporal filtering by employing ADSWIN [31], an adaptive sliding window approach that automatically segments the EEG trials and selects the best segment to give the optimal spatio-temporal patterns.

CSP was employed for extracting features of each segment obtained through ADSWIN. DBI was used as a cost function to identify the optimal segment. The EEG segment with minimum DBI was then selected as optimal EEG segment. CSP projection matrix was computed based on the selected optimal time segment to create a spatio-temporal profile. A detailed formulation of the method can be found in [31]. ADSWIN adapts two parameters, default segment length (ω_d) and overlapping region (δ) to compute the optimal spatio-temporal patterns. In this study, two different ranges of ω_d were analyzed ([3.6 4.5] seconds, i.e., [922 1152] time points and [2.99 3.5] seconds, i.e., [767 896] time points) because ω_d lower than 2.99 seconds would be too small to find the optimal number of time points and ω_d higher than 4.5 seconds may be too large. The best values of both ranges were chosen for ω_d . Two different values of δ (1 sec, i.e., 256 time points and 0.5 sec, i.e., 128 time points) were examined. The values of δ were chosen in such a way that it maintains even distributions along the trial. Also, it was set such that the larger value of δ would keep more historic information rather than new information. Different bandpass filter banks were examined and the best two (FB₁: 4-9 Hz, 9-17 Hz, 15-30 Hz, 30-40 Hz and FB₂: 4-9 Hz, 9-16 Hz, 15-32 Hz, 30-40 Hz) were chosen for the study. Based on the different combinations of ω_d and δ , four segments were then investigated for each filter bank. The spatio-temporal filter was then used to extract the features and KPLS-mRMR [32] selects the features. The number of selected features was set to 25.

Table 3: DBI score from offline analysis

Subjects	FB ₁											
	ω_1, δ_1			ω_1, δ_2			ω_2, δ_1			ω_2, δ_2		
	CSP	ACSP	ADCSP-I	CSP	ACSP	ADCSP-I	CSP	ACSP	ADCSP-I	CSP	ACSP	ADCSP-I
S1	15	13	12	17	19	14	13	14	12	20	20	15
S2	15	12	12	15	13	13	15	15	11	15	15	11
S3	26	21	19	27	23	18	23	21	15	25	21	18
S5	15	15	12	13	11	11	15	15	12	14	13	13
S6	17	19	16	17	19	16	17	20	14	17	20	14
Avg	17.6	16	14.2	17.8	17	14.4	16.6	17	12.8	18.2	17.8	14.2
Subjects	FB ₂											
	ω_1, δ_1			ω_1, δ_2			ω_2, δ_1			ω_2, δ_2		
	CSP	ACSP	ADCSP-I	CSP	ACSP	ADCSP-I	CSP	ACSP	ADCSP-I	CSP	ACSP	ADCSP-I
S1	16	16	13	17	18	14	15	15	12	21	16	15
S2	15	12	12	13	12	11	18	13	11	18	13	11
S3	28	21	20	25	20	18	32	25	18	28	23	17
S5	12	14	10	15	15	12	12	10	10	14	13	13
S6	18	19	15	20	22	14	18	20	14	18	20	14
Avg	17.8	16.4	14.2	18	17.4	13.8	19	16.6	13	19.8	17	14

3.2.2. ADCSP-I: ADCSP for Offline Analysis

- (i) **Training and Testing Sessions:** To carry out the offline analysis, the first two runs were used to compute the optimal spatio-temporal filter. The last 6 runs were used as evaluation data. Evaluation data were categorized as low and high fatigue level by K-means algorithm as described in Section 2.2. Table 2 shows the runs of the experiment that have been categorized as low or high fatigue level. The CSPs during high fatigue level were then adapted using Algorithm 1. ACSP for offline analysis is implemented in the same way as ADCSP-I.
- (ii) **Results:** ADCSP-I has been tried on the runs that have been categorized as high fatigue level. Table 3, Table 4 and Table 5 show the separability of MI EEG features extracted with ADCSP-I in terms of DBI, FS and DI respectively. In these tables, the top half show the results for FB₁ while the bottom half show the results for FB₂. The first column shows the Subject id, 2nd to 13th columns present the DBI/FS/DI values of C-CSP, ACSP and ADCSP-I for the four segments ($\omega_1, \delta_1; \omega_1, \delta_2; \omega_2, \delta_1; \omega_2, \delta_2$) respectively. The results of the Friedman statistical test are shown in Table 6 which demonstrate that ADCSP-I significantly outperformed C-CSP and ACSP in about 80% of the tests. Fig. 4 shows the class distributions of each task for Subject 1. The other subjects have similar distribution pattern. The center of each ellipse represents the class mean under that condi-

tion while the size of the ellipse represents the 95% confidence interval for the class, oriented along the eigenvectors of the covariance matrix. It is seen that all the four classes are more separable in case of ADCSP while in case of C-CSP, the classes are inseparable.

3.2.3. ADCSP-II: ADCSP in Near Real-time

- (i) **Training and Testing Sessions:** To evaluate the performance of ADCSP-II, the first two runs were used for training to compute the optimal spatio-temporal filter. The first and the last run were used for training the KPLS model for fatigue analysis while the 3rd to 7th runs were used for evaluation. CSP was adapted on trial basis. ACSP for near real-time analysis is implemented in the same way as ADCSP-II.
- (ii) **Results:** Table 7, Table 8 and Table 9 show the separability of MI EEG features extracted with ADCSP-II in terms of DBI, FS, and DI respectively. In these tables, the top half show the results of FB₁ while the results of FB₂ are shown in the bottom half. The first column shows the Subject id, the 2nd to 13th columns present the DBI/FS/DI values of C-CSP, ACSP and ADCSP-II for the four segments ($\omega_1, \delta_1; \omega_1, \delta_2; \omega_2, \delta_1; \omega_2, \delta_2$) respectively. The results of the Friedman statistical test are shown in Table 10, which demonstrate that ADCSP-II significantly outperformed C-CSP and ACSP in about 98% of the tests.

Table 4: Fisher score from offline analysis

Subjects	FB ₁											
	ω_{1,δ_1}			ω_{1,δ_2}			ω_{2,δ_1}			ω_{2,δ_2}		
	CSP	ACSP	ADCSP-I	CSP	ACSP	ADCSP-I	CSP	ACSP	ADCSP-I	CSP	ACSP	ADCSP-I
S1	0.04	0.074	0.21	0.05	0.05	0.17	0.007	0.097	0.14	0.02	0.097	0.19
S2	0.58	0.29	1.01	0.41	0.87	1.03	0.3	0.83	1.01	0.3	0.83	1.01
S3	0.004	0.01	0.027	0.003	0.03	0.067	0.009	0.04	0.068	0.004	0.03	0.067
S5	0.27	0.14	0.35	0.23	0.24	0.38	0.27	0.12	0.35	0.21	0.12	0.26
S6	0.06	0.049	0.09	0.07	0.06	0.12	0.06	0.07	0.16	0.06	0.07	0.16
Avg	0.19	0.26	0.47	0.15	0.25	0.35	0.13	0.23	0.35	0.12	0.23	0.34

Subjects	FB ₂											
	ω_{1,δ_1}			ω_{1,δ_2}			ω_{2,δ_1}			ω_{2,δ_2}		
	CSP	ACSP	ADCSP-I	CSP	ACSP	ADCSP-I	CSP	ACSP	ADCSP-I	CSP	ACSP	ADCSP-I
S1	0.07	0.11	0.24	0.04	0.052	0.35	0.02	0.09	0.13	0.01	0.04	0.32
S2	0.49	0.93	0.83	0.79	0.87	1.54	0.3	0.82	0.99	0.3	0.81	0.99
S3	0.005	0.01	0.03	0.001	0.02	0.04	0.003	0.03	0.07	0.007	0.01	0.02
S5	0.22	0.32	1.32	0.16	0.24	0.41	0.22	0.32	1.32	0.15	0.25	0.25
S6	0.06	0.05	0.1	0.06	0.05	0.09	0.06	0.1	0.13	0.06	0.1	0.13
Avg	0.17	0.28	0.51	0.21	0.25	0.49	0.13	0.27	0.53	0.11	0.24	0.34

Table 5: DI score from offline analysis

Subjects	FB ₁											
	ω_{1,δ_1}			ω_{1,δ_2}			ω_{2,δ_1}			ω_{2,δ_2}		
	CSP	ACSP	ADCSP-I	CSP	ACSP	ADCSP-I	CSP	ACSP	ADCSP-I	CSP	ACSP	ADCSP-I
S1	0.07	0.1	0.09	0.08	0.07	0.11	0.1	0.08	0.12	0.07	0.06	0.09
S2	0.13	0.14	0.16	0.11	0.12	0.14	0.1	0.11	0.17	0.1	0.11	0.17
S3	0.08	0.08	0.09	0.07	0.08	0.09	0.07	0.09	0.11	0.07	0.08	0.09
S5	0.1	0.13	0.14	0.11	0.15	0.16	0.1	0.12	0.14	0.1	0.13	0.14
S6	0.1	0.097	0.12	0.1	0.094	0.11	0.1	0.014	0.13	0.1	0.01	0.13
Avg	0.09	0.11	0.12	0.09	0.1	0.12	0.09	0.09	0.13	0.09	0.08	0.12

Subjects	FB ₂											
	ω_{1,δ_1}			ω_{1,δ_2}			ω_{2,δ_1}			ω_{2,δ_2}		
	CSP	ACSP	ADCSP-I	CSP	ACSP	ADCSP-I	CSP	ACSP	ADCSP-I	CSP	ACSP	ADCSP-I
S1	0.07	0.08	0.09	0.1	0.07	0.11	0.07	0.08	0.12	0.07	0.08	0.09
S2	0.11	0.14	0.14	0.07	0.14	0.17	0.08	0.12	0.16	0.08	0.13	0.16
S3	0.06	0.07	0.08	0.09	0.09	0.1	0.05	0.07	0.09	0.05	0.08	0.09
S5	0.16	0.12	0.17	0.1	0.12	0.14	0.16	0.12	0.17	0.1	0.13	0.14
S6	0.09	0.09	0.12	0.08	0.07	0.13	0.09	0.09	0.13	0.09	0.09	0.13
Avg	0.09	0.1	0.12	0.09	0.1	0.13	0.09	0.1	0.14	0.08	0.1	0.13

Table 6: Statistical significance test of ADCSP-I vs C-CSP and ACSP for offline analysis

	FB ₁									
	ω_{1,δ_1}		ω_{1,δ_2}				ω_{2,δ_1}		ω_{2,δ_2}	
	CSP vs ADCSP-I	ACSP vs ADCSP-I	CSP vs ADCSP-I	ACSP vs ADCSP-I	ADCSP-I vs ADCSP-I	CSP vs ADCSP-I	ACSP vs ADCSP-I	CSP vs ADCSP-I	ACSP vs ADCSP-I	
DBI	0.045 (✓)	0.05 (×)	0.045 (✓)	0.13 (×)		0.045 (✓)	0.045 (✓)	0.045 (✓)	0.06 (×)	
Fisher Score	0.03 (✓)	0.06 (×)	0.045 (✓)	0.045 (✓)		0.03 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	
DI	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.03 (✓)		0.045 (✓)	0.045 (✓)	0.045 (✓)	0.07 (×)	

	FB ₁									
	ω_{1,δ_1}		ω_{1,δ_2}				ω_{2,δ_1}		ω_{2,δ_2}	
	CSP vs ADCSP-I	ACSP vs ADCSP-I	CSP vs ADCSP-I	ACSP vs ADCSP-I	ADCSP-I vs ADCSP-I	CSP vs ADCSP-I	ACSP vs ADCSP-I	CSP vs ADCSP-I	ACSP vs ADCSP-I	
DBI	0.045 (✓)	0.06 (×)	0.045 (✓)	0.13 (×)		0.045 (✓)	0.045 (✓)	0.03 (✓)	0.06 (×)	
Fisher Score	0.06 (×)	0.045 (✓)	0.045 (✓)	0.045 (✓)		0.045 (✓)	0.045 (✓)	0.03 (✓)	0.045 (✓)	
DI	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.06 (×)		0.045 (✓)	0.045 (✓)	0.045 (✓)	0.03 (✓)	

Table 7: DBI score from near real-time analysis

Subjects	FB ₁											
	ω_{1,δ_1}			ω_{1,δ_2}			ω_{2,δ_1}			ω_{2,δ_2}		
	CSP	ACSP	ADCSP-II	CSP	ACSP	ADCSP-II	CSP	ACSP	ADCSP-II	CSP	ACSP	ADCSP-II
S1	12	20	11	13	21	12	9	25	11	17	23	16
S2	19	31	18	18	25	16	21	20	18	21	41	18
S3	33	27	25	37	22	26	34	26	26	30	26	24
S5	17	30	16	17	21	16	17	30	16	16	27	15
S6	25	30	20	20	29	19	26	29	17	26	27	17
Avg	21.2	27.6	18	21	23.6	17.8	21.4	26	17.6	22	28.8	18
Subjects	FB ₂											
	ω_{1,δ_1}			ω_{1,δ_2}			ω_{2,δ_1}			ω_{2,δ_2}		
	CSP	ACSP	ADCSP-II	CSP	ACSP	ADCSP-II	CSP	ACSP	ADCSP-II	CSP	ACSP	ADCSP-II
S1	16	25	12	12	27	11	16	25	12	21	30	16
S2	19	31	17	20	30	18	19	25	18	19	25	18
S3	35	28	27	37	24	25	28	27	25	35	30	24
S5	19	27	16	19	28	16	19	29	16	17	27	16
S6	24	30	17	24	26	21	22	24	16	22	23	16
Avg	22.6	28.2	17.8	22.4	27	18.2	20	26	17.4	22.8	27	18

Table 8: Fisher score from near real-time analysis

Subjects	FB ₁											
	ω_{1,δ_1}			ω_{1,δ_2}			ω_{2,δ_1}			ω_{2,δ_2}		
	CSP	ACSP	ADCSP-II	CSP	ACSP	ADCSP-II	CSP	ACSP	ADCSP-II	CSP	ACSP	ADCSP-II
S1	0.06	0.024	0.25	0.14	0.002	0.2	0.12	0.003	0.34	0.06	0.01	0.07
S2	0.14	0.01	0.22	0.17	0.007	0.34	0.14	0.005	0.3	0.14	0.05	0.3
S3	0.005	0.003	0.18	0.004	0.008	0.009	0.005	0.01	0.01	0.005	0.003	0.01
S5	0.15	0.007	0.28	0.15	0.006	0.28	0.15	0.002	0.28	0.25	0.04	0.33
S6	0.02	0.001	0.08	0.03	0.005	0.05	0.02	0.003	0.16	0.02	0.004	0.16
Avg	0.07	0.01	0.17	0.09	0.006	0.18	0.09	0.004	0.22	0.09	0.021	0.17
Subjects	FB ₂											
	ω_{1,δ_1}			ω_{1,δ_2}			ω_{2,δ_1}			ω_{2,δ_2}		
	CSP	ACSP	ADCSP-II	CSP	ACSP	ADCSP-II	CSP	ACSP	ADCSP-II	CSP	ACSP	ADCSP-II
S1	0.12	0.03	0.15	0.12	0.02	0.26	0.1	0.01	0.3	0.07	0.002	0.11
S2	0.14	0.08	0.24	0.15	0.09	0.34	0.20	0.09	0.21	0.2	0.07	0.21
S3	0.005	0.005	0.02	0.004	0.006	0.02	0.009	0.003	0.018	0.005	0.006	0.011
S5	0.03	0.006	0.51	0.03	0.002	0.51	0.03	0.006	0.51	0.24	0.007	0.25
S6	0.06	0.002	0.1	0.06	0.06	0.23	0.03	0.01	0.08	0.03	0.01	0.08
Avg	0.07	0.025	0.21	0.09	0.04	0.27	0.08	0.02	0.3	0.11	0.02	0.13

Table 9: DI score from near real-time analysis

Subjects	FB ₁											
	ω_1, δ_1			ω_1, δ_2			ω_2, δ_1			ω_2, δ_2		
	CSP	ACSP	ADCSP-II	CSP	ACSP	ADCSP-II	CSP	ACSP	ADCSP-II	CSP	ACSP	ADCSP-II
S1	0.11	0.06	0.13	0.11	0.04	0.13	0.19	0.05	0.15	0.09	0.05	0.11
S2	0.1	0.03	0.11	0.09	0.08	0.12	0.08	0.03	0.11	0.08	0.03	0.11
S3	0.05	0.07	0.07	0.04	0.08	0.06	0.05	0.07	0.06	0.06	0.06	0.07
S5	0.11	0.06	0.13	0.11	0.06	0.13	0.11	0.07	0.13	0.11	0.06	0.12
S6	0.08	0.05	0.09	0.09	0.05	0.1	0.06	0.002	0.09	0.06	0.06	0.09
Avg	0.09	0.05	0.11	0.09	0.06	0.11	0.09	0.04	0.11	0.08	0.07	0.10

Subjects	FB ₂											
	ω_1, δ_1			ω_1, δ_2			ω_2, δ_1			ω_2, δ_2		
	CSP	ACSP	ADCSP-II	CSP	ACSP	ADCSP-II	CSP	ACSP	ADCSP-II	CSP	ACSP	ADCSP-II
S1	0.08	0.05	0.14	0.13	0.05	0.14	0.08	0.05	0.14	0.07	0.03	0.11
S2	0.1	0.01	0.09	0.11	0.03	0.10	0.09	0.03	0.12	0.09	0.03	0.12
S3	0.04	0.06	0.07	0.04	0.08	0.06	0.05	0.07	0.06	0.05	0.07	0.06
S5	0.10	0.06	0.11	0.1	0.06	0.11	0.1	0.06	0.11	0.11	0.06	0.12
S6	0.07	0.04	0.1	0.007	0.006	0.09	0.08	0.07	0.11	0.08	0.01	0.11
Avg	0.09	0.04	0.1	0.07	0.06	0.1	0.08	0.06	0.1	0.08	0.04	0.11

Table 10: Statistical significance test of ADCSP-II vs C-CSP and ACSP for near real-time analysis

	FB ₁									
	ω_1, δ_1		ω_1, δ_2		ω_2, δ_1		ω_2, δ_2			
	CSP vs ADCSP-II	ACSP vs ADCSP-II	CSP vs ADCSP-II	ACSP vs ADCSP-II	CSP vs ADCSP-II	ACSP vs ADCSP-II	CSP vs ADCSP-II	ACSP vs ADCSP-II	CSP vs ADCSP-II	ACSP vs ADCSP-II
DBI	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.06 (×)
Fisher Score	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)
DI	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)

	FB ₂									
	ω_1, δ_1		ω_1, δ_2		ω_2, δ_1		ω_2, δ_2			
	CSP vs ADCSP-II	ACSP vs ADCSP-II	CSP vs ADCSP-II	ACSP vs ADCSP-II	CSP vs ADCSP-II	ACSP vs ADCSP-II	CSP vs ADCSP-II	ACSP vs ADCSP-II	CSP vs ADCSP-II	ACSP vs ADCSP-II
DBI	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)
Fisher Score	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)
DI	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)	0.045 (✓)

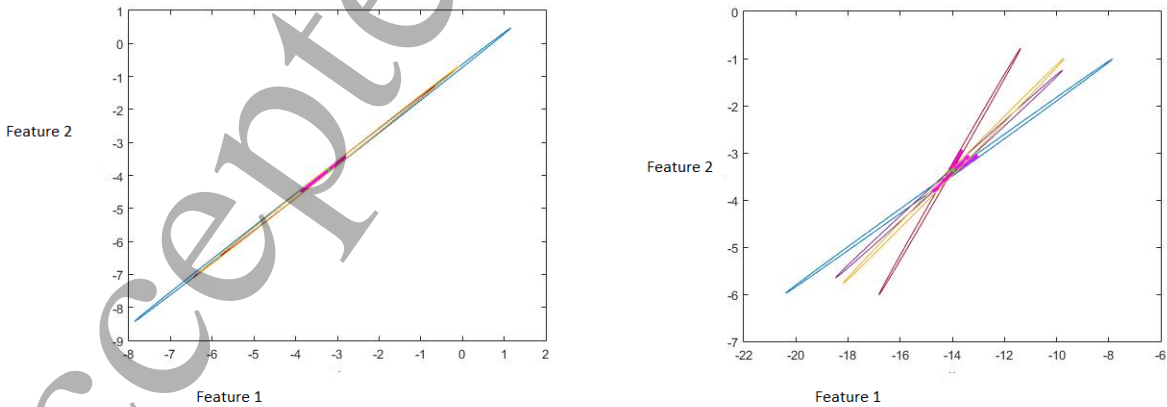


Figure 4: Class distributions for Subject 1 (FB₁, Segment: ω_1, δ_2). (a) shows the separability of features in extracted by C-CSP while (b) shows separability of features extracted by ADCSP. Classes are inseparable when C-CSP is used and more separable when ADCSP is used. Blue colour represents Class 1, Orange Class 2, Magenta Class 3 and Violet Class 4

4. Discussion

4.1. ADCSP-I and ADCSP-II vs C-CSP

Tables 3-5 show that the average separability of MI EEG features extracted with ADCSP-I across all the 5 subjects is higher in terms of DBI, FS, and DI than that with C-CSP. Similarly, Tables 7-9 portray that the average separability of MI EEG features extracted with ADCSP-II across all the 5 subjects is higher in terms of DBI, FS, and DI than that with C-CSP. The Friedman statistical test as shown in Table 6 and Table 10 reveals that the separability of MI EEG features extracted with ADCSP-I and ADCSP-II increases significantly in terms of DBI, FS, and DI when compared to that with C-CSP. The improved separability of MI EEG features with ADCSP over C-CSP demonstrates clear benefits of adapting CSP during high fatigue level. Therefore, the adaptation based on cognitive state provides a means to improve the separability of MI EEG features.

4.2. ADCSP-I and ADCSP-II vs ACSP

Tables 3-5 show that the average separability of MI EEG features extracted with ADCSP-I across all the 5 subjects is higher in terms of DBI, FS and DI than that with ACSP in almost all the cases except in the case of S2 with ω_1, δ_1, FB_1 and ω_1, δ_1, FB_2 where ACSP shows better performance in terms of FS, while in the case of S2 with $\omega_1, \delta_1, FB_1, \omega_1, \delta_2, FB_1$ and $\omega_1, \delta_1, FB_2, S5$ with $\omega_1, \delta_2, FB_1, \omega_2, \delta_2, FB_1, \omega_1, \delta_2, FB_2$ and ω_2, δ_2, FB_2 , ACSP shows equal performance to that with ADCSP-I in terms of DBI. The Friedman statistical test as shown in Table 6 shows the significantly improved separability of MI EEG features with ADCSP-I as compared to that with ACSP except for the aforesaid cases.

However, in the case of near real-time analysis, as shown in Tables 7-9, ADCSP-II outperformed ACSP in all the cases except in the case of S3 with ω_2, δ_2, FB_1 . Friedman statistical test as shown in Table 10 shows significant improvement of the separability of MI EEG features extracted with ADCSP-II as compared to that with ACSP except in the case of ω_2, δ_2, FB_1 . The performance of ACSP is even worse than that of C-CSP in the near real-time analysis.

The poor performance of ACSP as compared to that of ADCSP-I and ADCSP-II may be because of two reasons: First, ADCSP uses the regularisation parameter γ to update the covariance matrices of each class. The use of γ counters the bias due to the small training set. Second, ADCSP uses β to counter the dissimilarity between the training and testing data during high fatigue level which is not used by ACSP.

4.3. Limitations and Future Work

It can be concluded that the separability of MI EEG features can be significantly improved through ADCSP. However, there are some limitations of the proposed method. First, the method needs to estimate an optimal value for β and γ to update the covariance matrices causing some computational drawbacks. The study does not include any automatic way to estimate optimal β and γ for a particular problem. Future work would include an automatic method of estimating β and γ using gradient descent algorithm.

Second, the effect of mental fatigue on MI EEG feature separability is subject-specific and is highly individualized. The term "one size fits all" may not be applicable in implementing an adaptive approach by tracking such user's cognitive state. Therefore, parameters such as the size of the training set, range of β and γ , the range of 'p' and 'n_t' need to be set individually, causing some computational drawbacks.

Third, the 'init' in case of ADCSP-II is based on high fatigue level instead of the fatigue score. It may happen that a slight increase in fatigue score will activate adaptation, but in reality, the subject may not be experiencing much fatigue and adaptation is actually not required. Hence accurate and robust estimation of the point where fatigue enters high fatigue level and the adaptation process is activated is substantial.

Fourth, although, the proposed methodology works well on the 5 samples, applying this model to a larger number of subjects of all ages, gender etc. would reinforce the study.

Fifth, this study used class separability metrics like DBI, DI, and FS for evaluating the separability of MI EEG features. In practice, the adaptive methods need various evaluation metrics specially because of the issue of unreliable labels. Information based methods, classification accuracy, true-false difference can also be used for such evaluation. However, these were not explored as the thrust of the study was to enhance class separability. Future work would include validation of all the aforesaid approaches using different evaluation metrics.

5. Conclusion

This study proposes an unsupervised adaptive scheme called ADCSP to update CSP by tracking the fatigue state of the user. The proposed method improves the class separability of MI EEG features during high fatigue level. Based on the two regularization parameters β and γ and KL between the training and testing data, the method updates the CSP both in offline mode and in near real-time as well. Experimental results demonstrate that the class

REFERENCES

14

separability of MI EEG features increases significantly with the adaptive approach as compared to that with C-CSP and ACSP. The results collectively portray the benefit of adapting CSP by tracking the fatigue state of the user.

Acknowledgment

Financial support from MHRD as a Centre of Excellence on Machine Learning Research and Big Data Analysis is acknowledged. We would also like to acknowledge the DST-UKEIRI Project: DST/INT/UK/P-91/2014.

References

- [1] Curran, E. A. and Stokes, M. J. Learning to control brain activity: a review of the production and control of EEG components for driving brain-computer interface (BCI) systems. *Brain and Cognition*, 51(3):326–336, 2003.
- [2] Myrden, A. and Chau, T. Effects of user mental state on EEG-BCI performance. *Frontiers in Human Neuroscience*, 9:308, 2015.
- [3] Sun, S. and Zhang, C. Adaptive feature extraction for EEG signal classification. *Medical and Biological Engineering and Computing*, 44(10):931–935, 2006.
- [4] Arvaneh, M., Guan, C., Ang, K. K., and Quek, C. EEG data space adaptation to reduce intersession nonstationarity in brain-computer interface. *Neural Computation*, 25(8):2146–2171, 2013.
- [5] Arvaneh, M., Guan, C., Ang, K. K., and Quek, C. Optimizing spatial filters by minimizing within-class dissimilarities in electroencephalogram-based brain-computer interface. *IEEE Transactions on Neural Networks and Learning Systems*, 24(4):610–619, 2013.
- [6] Bamdadian, A., Guan, C., Ang, K. K., and Xu, J. Online semi-supervised learning with KL distance weighting for motor imagery-based BCI. In *IEEE International Conference on Engineering in Medicine and Biology Society (EMBC), 2012*, pages 2732–2735. IEEE, 2012.
- [7] Song, X. and Yoon, S.-C. Improving brain-computer interface classification using adaptive common spatial patterns. *Computers in Biology and Medicine*, 61:150–160, 2015.
- [8] Song, X. and Yoon, S.-C. Incremental versus non-incremental learning in adaptive common spatial patterns. In *Signal Processing in Medicine and Biology Symposium (SPMB)*, pages 1–4. IEEE, 2015.
- [9] Jin, Y., Mousavi, M., and de Sa, V. R. Adaptive csp with subspace alignment for subject-to-subject transfer in motor imagery brain-computer interfaces. In *IEEE 6th International Conference on Brain-Computer Interface (BCI), 2018*, pages 1–4. IEEE, 2018.
- [10] Raza, H., Cecotti, H., Li, Y., and Prasad, G. Adaptive learning with covariate shift-detection for motor imagery-based brain-computer interface. *Soft Computing*, 20(8):3085–3096, 2016.
- [11] Chowdhury, A., Raza, H., Meena, Y. K., Dutta, A., and Prasad, G. Online Covariate Shift detection based Adaptive Brain-Computer Interface to Trigger Hand Exoskeleton Feedback for Neuro-Rehabilitation. *IEEE Transactions on Cognitive and Developmental Systems*, 10(4):1070–1080, 2018.
- [12] Nicolas-Alonso, L. F., Corralejo, R., Gomez-Pilar, J., Álvarez, D., and Hornero, R. Adaptive semi-supervised classification to reduce intersession non-stationarity in multiclass motor imagery-based brain-computer interfaces. *Neurocomputing*, 159:186–196, 2015.
- [13] Tsui, C. S. L., Gan, J. Q., and Roberts, S. J. A self-paced brain-computer interface for controlling a robot simulator: an online event labelling paradigm and an extended Kalman filter based algorithm for online training. *Medical & Biological Engineering & Computing*, 47(3):257–265, 2009.
- [14] Müller-Gerking, J., Pfurtscheller, G., and Flyvbjerg, H. Designing optimal spatial filters for single-trial EEG classification in a movement task. *Clinical Neurophysiology*, 110(5):787–798, 1999.
- [15] Talukdar, U., Hazarika, S. M., and Gan, J. Q. Adaptation of Common Spatial Patterns based on Mental Fatigue in EEG MI-BCI. *Biomedical Signal Processing and Control (Submitted)*.
- [16] Talukdar, U., Hazarika, S. M., and Gan, J. Q. Motor imagery and mental fatigue : inter-relationship and EEG based estimation. *Journal of Computational Neuroscience*, 46(1):55–76, 2019.
- [17] Myrden, A. and Chau, T. Towards psychologically adaptive brain-computer interfaces. *Journal of Neural Engineering*, 13(6):066022, 2016.
- [18] Rosipal, R. and Trejo, L. J. Kernel partial least squares regression in reproducing kernel hilbert space. *Journal of Machine Learning Research*, 2 (Dec):97–123, 2001.
- [19] Davies, D. L. and Bouldin, D. W. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(2):224–227, 1979.

REFERENCES

- [20] Dunn, J. C. A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. *Journal of Cybernetics*, 3(3): 32–57, 1973.
- [21] Duda, R. O., Hart, P. E., and Stork, D. G. *Pattern Classification*. Wiley, New York, 1973.
- [22] Lee, K. A., Hicks, G., and Nino-Murcia, G. Validity and reliability of a scale to assess fatigue. *Psychiatry Research*, 36(3):291–298, 1991.
- [23] Cella, M. and Chalder, T. Measuring fatigue in clinical and community settings. *Journal of Psychosomatic Research*, 69(1):17–22, 2010.
- [24] Mammone, N. and Morabito, F. C. Enhanced automatic wavelet independent component analysis for electroencephalographic artifact removal. *Entropy*, 16(12):6553–6572, 2014.
- [25] Lu, H., Plataniotis, K. N., and Venetsanopoulos, A. N. Regularized common spatial patterns with generic learning for EEG signal classification. In *Annual International Conference of Engineering in Medicine and Biology Society (EMBC 2009)*., pages 6599–6602. IEEE, 2009.
- [26] Li, Y. and Guan, C. A semi-supervised SVM learning algorithm for joint feature extraction and classification in brain computer interfaces. In *In 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS'06)*, pages 2570–2573, 2006.
- [27] Hasan, B. A. S. *Adaptive Methods Exploiting the Time Structure in EEG for Self-paced Brain-Computer Interfaces*. PhD thesis, University of Essex, 2010.
- [28] Mthembu, L. and Marwala, T. A note on the separability index. *arXiv preprint arXiv:0812.1107*, 2008.
- [29] Adel, T., Wong, A., and Stashuk, D. A weakly supervised learning approach based on spectral graph-theoretic grouping. *arXiv preprint arXiv:1508.00507*, 2015.
- [30] Löster, T. Determining the optimal number of clusters in cluster analysis. In *The 10th International Days of Statistics and Economics, Prague*, pages 1078–1090, 2016.
- [31] Talukdar, U. and Hazarika, S. M. Designing optimal spatio-temporal filter for single trial EEG based BCI. In *3rd International Conference on Advances in Robotics (AIR)*, page 4. ACM, 2017.
- [32] Talukdar, U., Hazarika, S. M., and Gan, J. Q. A kernel partial least square based feature selection method. *Pattern Recognition*, 83:91–106, 2018.