I J C T A, 9(18) 2016, pp. 8869-8876

A BCI Binary Classification Technique with Regularized CSP and PCA

B. Bijitha* and Nandakumar Paramparambath*

ABSTRACT

Brain-Computer Interface is a technology in which brain wave signals (mostly EEG signals) are used to control various external devices like computer, wheelchair etc. Studies on BCI technology are more prevalent nowadays since it seems to be useful as a means of communication for people suffering from diseases like ALS, where brain wave is the only option for communication with the external world. In this paper BCI algorithm with feature extraction using regularized CSP and PCA, feature selection, and stacked concept for classification is employed.

Keywords: brain computer interfaces, feature extraction, common spatial pattern, classification.

I. INTRODUCTION

The World Health Organisation says that around 15% of the world population suffers from disability (includes both physical and mental disabilities) [1]. The disability may be caused by various neurological disorders that can be caused by either due to biochemical causes or due to physical injury to the brain, spinal cord or nerves. The disabled people find difficulty in communication with outside world especially those with severe motor disabilities. Brain-computer Interface technology can be a solution for those kind of people especially those suffering from disorders like brainstem stroke, amyotrophic lateral sclerosis (ALS), high-level spinal cord injury, and cerebral palsy[2]. BCI has various definitions, of which one that defined by Wolpaw *et al.* [3] is as follows: "A BCI is a communication system in which messages or commands that an individual sends to the external world do not pass through the brain's normal output pathways of peripheral nerves and muscles". The brain-computer interfaces (BCIs) are otherwise called as the brain-machine interfaces (BMIs) or direct brain interfaces (DBIs) or human-computer interface. Other than its application as a communication alternative, BCI has various other applications in diagnosis, environment control, games and entertainment, security and authentication, etc [4, 5]. Hussanien *et al.* [6] have mentioned different classifications of BCI: exogenous and endogenous, asynchronous and synchronous. Wolpaw *et al.* [3] have mentioned another type of classification as dependent and independent.

BCI system has various stages: brain signal acquisition, feature extraction/selection and last the classification stage [7]. The preprocessing stages will also be there to remove the external noise and the artifacts. The BCI technology uses various brain imaging methods to extract the brain activity: MEG, EEG, fMRI, ECoG, PET [8]. The most commonly used one is the EEG, since it is noninvasive as well as cheaper. The extracted brain signals can be classified into various types as mu and beta rhythms, event-related potentials, visual evoked potentials, event related synchronization/ desynchronization, and slow cortical potentials [9, 10]. The most commonly exploited signals are the sensorimotor rhythms, which are oscillations in alpha (8-12 Hz) and beta (18-26 Hz) recorded over the sensorimotor areas of brain [11, 12]. The information contained in these signals, called as features, need to be extracted for which various methods have been used like CSP, autoregressive model, wavelet transform, ICA, PCA [13, 14]. Then the extracted features

* Department of Electronics and Communication Engineering, NSS College of Engineering, Palakkad, Kerala, India, *E-mails: bijithabalakrishnan@gmail.com; nandakumarpp@hotmail.com*

need to be translated using translational algorithms. The translational algorithms can be either a classifier or a regression function [15].

Despite its significance, BCI technology faces various problems. Non-stationarity is the most important problem that degrades the efficiency of the BCI systems [16]. The non-stationarity of EEG signals is due to the fact that the signals vary with time due to the impact of internal (artifacts) and external noises (due to equipments). Adaptive methods can be employed to overcome the non-stationarity problem [16, 17]. The non-stationarity affects the feature extraction stage. At the classification stage the performance gets affected by problem of curse of dimensionality and bias-variance trade-off. Hence the objective of this experiment is to find an algorithm that could mitigate these problems.

The rest of the paper is organized as follows. The dataset employed for the experiment is explained in section II. The proposed algorithm for BCI binary classification is illustrated in section III. Section IV describes the results obtained in the experiment and section V provides the conclusion and the future works possible.

II. DATASET

The dataset used here is the BCI Competition IV dataset 2a, which was provided by Graz University [18]. The dataset consists of EEG signals acquired from 9 healthy subjects. For each subject, data collected consists of four different motor imagery tasks: movement of left hand, movement of right hand, tongue movement and feet movement. There are two sessions one is for training and the other is for evaluation. Each session consists of 72 trilas for each task, hence a total of 288 trials. Twenty-two EEG channels and 3 EOG channels constitute the recording montage. The signals were sampled at 250 Hz. It was also preprocessed. Besides dataset 2a, Graz University provides various other datasets that are useful for BCI research. The evaluation session consists of unknown cues and is used in the experiment to evaluate the algorithm. The evaluation is analyzed using the value of kappa coefficient. The kappa coefficient ranges from 0 (slight or no agreement) to 1 (perfect agreement). The results for the evaluation set are all declared and are available in the web page *http://www.bbci.de/competition/iv/results.*

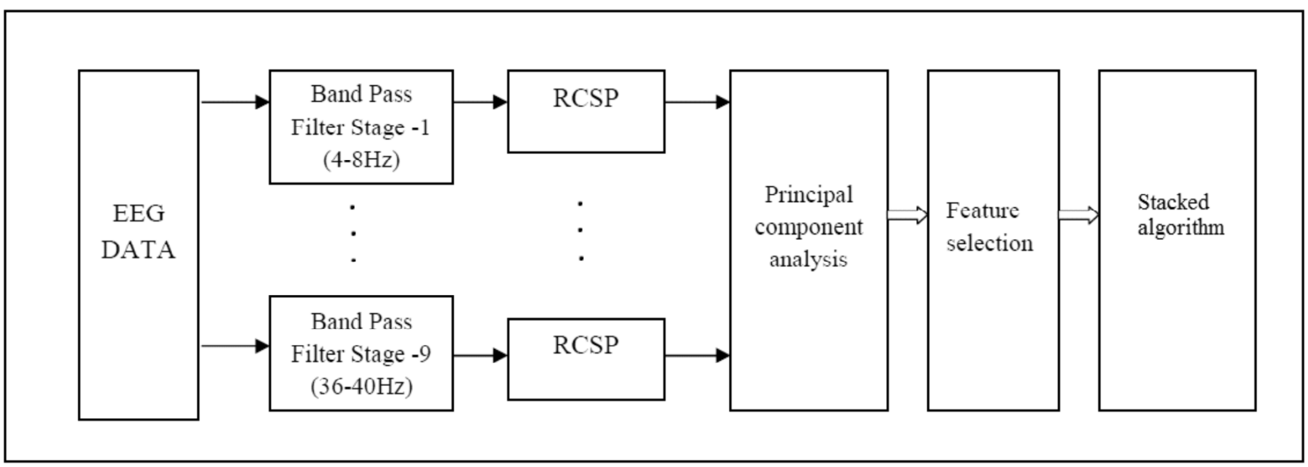

Figure 1: Block Diagram of the Brain Computer Interface Algorithm Implemented in the Paper

III. METHOD

The algorithm that is implemented in this paper is illustrated in Figure.1. The signal processing chain consists of multiple band pass filtering stages, feature extraction using RCSP and PCA, feature selection, and then classification stage using RLDA with stacked concept. Here the regularization is done at two processing stages: the feature extraction stage and the classification stage.

3.1. Feature Extraction

1) *Filtering:* The EEG signal is band-pass filtered such that it is split into 9 different spectral bands: 4-8 Hz, 8-12 Hz… 32-36, 36-40 Hz [19, 20]. FIR filters with Kaiser Window are used and a 1 Hz transition band is set at each of the nine filters. The filter is designed with the help of 'kaiserord' MATLAB function.

2) *Spatial filtering*: To discriminate with different motor imagery tasks, it is necessary to identify the sources of corresponding sensory motor rhythm (SMR) modulations and this can be achieved through spatial filtering [21]. Common spatial pattern (CSP) algorithm is mostly used as a spatial filtering method [22]. The CSP filters maximize the variance for one class and at the same time minimize the variance for the other [22]. The spatial filtering in CSP is done by linearly transforming EEG data using equation (1),

$$\mathrm{Z} = \mathrm{W}^{\mathrm{T}}\mathrm{E}, \tag{1}$$

where W is CSP projection matrix, E is EEG measurement, and *Z* is EEG measurement after spatial filtering [19]. In this paper, instead of CSP a regularized version is used called RCSP. Regularization was introduced to overcome the small-sample problem of the discriminant analysis [23]. Fabien *et al* [24] have compared different regularization methods and have concluded that RCSP outperforms CSP by 10% in classification accuracy. The regularized RCSP employed here is only a small variation with basic CSP. In our experiment the regularization parameter chosen was 0.1, which is selected by trial and error method. The regularization is introduced in the eigenvalue decomposition problem,

$$\Sigma_1 W = (\Sigma_1 + 0.1 * \Sigma_2)\, WD, \tag{2}$$

where Σ_1 and Σ_2 are covariance matrices of two classes, D is diagonal matrix containing eigen values of Σ_1 [19]. Along with regularization the spatial filtering algorithm proposed by Ang *et al* [19] is used here to enhance the CSP performance for two-class problems. Then PCA is employed. The principal component analysis is a dimension reduction method and helps in identifying the principal components [25].

3.2. Feature Selection

The features extracted in the feature extraction are large in number, hence to select most relevant among them feature selection algorithm is employed. The feature selection method employed here is the mutual information based best individual feature (MIBIF) algorithm [19]. In the algorithm the mutual information of the features are computed. Then the features are arranged in descending order of mutual information and the first *k* features are selected. The value of *k* used here is 15, which is selected by heuristic approach.

3.3. Classification

The basic classifier usually used is the linear discriminant analysis (LDA) classifier [26, 27]. The basic LDA as described by Vidaurre *et al.* [26] is as follows:

$$V(x) = [b \;\; w^T]\begin{bmatrix} 1 \\ x \end{bmatrix} \tag{3}$$

$$w = \Sigma^{-1}\left(\mu_1 + \mu_2\right) \tag{4}$$

$$b = -w^T\left(\frac{\mu_1 + \mu_2}{2}\right) \tag{5}$$

The V(x) describes the distance of feature vector x from the hyperplane. The hyperplane is defined by the normal vector w and the bias b. Here μ_1 and μ_2 indicate the sample mean of two classes and Σ is the covariance of the classes considered to be equal. If V(x) is less than 0 then x is classified as class1, otherwise as class 2 [26]. The LDA can also be used as a dimensionality reduction method [28]. Different from CSP, LDA classifier increases the distance between the classes by minimizing the variance within a class [28]. The classifier employed is the regularized version of LDA. Here, the score obtained from the first RLDA classifier is input to another RLDA classifier. The idea for this type of classification is obtained from [20], wherein the concept of stacked generalization is used in classification algorithm. Nicolas *et al* [20] have used 25 RLDA models per second in level-0 and the results combined and fed to another RLDA classifier in level-1. However in this experiment only a single RLDA classifier is employed at each level.

IV. RESULTS

The experiment is done on the publicly available BCI Competition IV Dataset 2a that are provided by the Graz University. The algorithm is applied for binary class problems. The dataset contains recordings of 9 different subjects with two sets for each subject: one with known cues (training set) and other with unknown cues (evaluation set). The dataset contains 288 trials in each of the two sets. From the dataset one second samples are taken for experiment. A multiple band pass filtering is done wherein the data is discriminated on the basis of frequency. There are 9 band pass filters [(4-8Hz), (8-12Hz)….. (36-40Hz)] each with a transition band of 1Hz. The band passed data is then applied to a regularized version of basic CSP. A regularization parameter of 0.1 is applied to the second component while computing the eigenvalue of the two classes. The parameter is selected by trial and error method. The coefficients obtained are then applied to the spatial filtering algorithm proposed by Ang *et al.* [18], from which the first two and last two columns are selected and hence a total of 36 (9*4) features are obtained. Then PCA is applied. The PCA yields principal components and also helps in reducing noise factors. After the feature extraction stage, 15 most relevant features are selected using MIBIF algorithm [19]. These 15 features then applied to an RLDA classifier. The basic LDA classifiers provide probability scores and the classification is done with these scores and a threshold value. If the probability score is higher than a threshold then it belongs to class 1 otherwise class 2. Here the scores obtained from the RLDA classifier are applied to another classifier. The regularization parameter in the covariance matrix estimation is assigned a value of 0.1.

The algorithm is evaluated using the evaluation data provided for BCI Competition IV Dataset 2a. Here binary-class problems are only evaluated. Since the dataset has samples for four classes (namely: left hand, right hand, tongue movement and feet movement), six binary class problems (left hand v/s right hand, left hand v/s foot, left hand v/s tongue, right hand v/s foot, right hand v/s tongue, and foot v/s tongue) are possible as listed in the Tables. Even though for binary classification 144 trials are available, here only 72 trials (36 from each of two classes) are taken. The results obtained are compared with the SRLDA algorithm by Nicolas *et al.* [20]. The results using CSP and using RCSP are also compared. The algorithm is evaluated using the kappa coefficient. In the tables the results that have high kappa values for the proposed algorithm are shown in bold. The tables also show the results for the 8-fold cross-validation (values within brackets) and that with the training dataset [values within square brackets]. The cross-validation results are all high, but when applied to the evaluation set, the results are little low. The kappa values obtained for proposed algorithm shows that regularization has only a slight impact on the CSP stage in the proposed algorithm. The experiment is implemented in MATLAB and with the help of BIOSIG toolbox. BIOSIG is an open source software and can be downloaded from web page http://biosig.sourceforge.net/download.html.

V. CONCLUSION AND FUTURE WORK

The regularization is done at two signal processing stages of the BCI system: at feature extraction (RCSP) and at the classifier stage (RLDA). The cross validation results are high. 8-fold cross validation is done. Then the algorithm is verified with evaluation data. The results are compared with the SRLDA method proposed by Nicolas *et al.* [20]. In some cases the results are high compared to SRLDA [20]. Here the

Table 1
Summary of kappa values for Binary Class Problem: Left-Right

Method	*Left-Right Subjects*									
	A1	*A2*	*A3*	*A4*	*A5*	*A6*	*A7*	*A8*	*A9*	*Avg.*
SRLD Nicolas *et al* [20]	0.82	0.39	0.92	0.51	0.89	0.49	0.96	0.96	0.81	0.75
Stacked algorithm with RCSP	0.59 (0.97) [0.91]	0.61 (0.85) [0.75]	0.52 (0.94) [0.88]	0.47 (0.94) [0.88]	0.83 (0.94) [0.77]	0.67 (0.96) [0.97]	0.46 (0.93) [0.83]	0.67 (0.97) [0.94]	0.56 (0.85) [0.75]	0.60 (0.93) [0.85]
Stacked algorithm with CSP	0.75 (1) [0.86]	**0.66** (0.94) [0.86]	0.80 (0.94) [0.91]	**0.69** (0.84) [0.71]	0.68 (0.97) [0.83]	**0.84** (0.92) [0.97]	0.51 (0.84) [0.67]	0.88 (0.97) [0.88]	0.40 (0.97) [0.69]	0.69 (0.93) [0.82]

Table 2
Summary of kappa values for Binary Class Problem: Left-Foot

Method	*Left-Foot Subjects*									
	A1	*A2*	*A3*	*A4*	*A5*	*A6*	*A7*	*A8*	*A9*	*Avg.*
SRLD Nicolas *et al* [20]	0.96	0.82	0.96	0.75	0.71	0.61	1	0.88	0.96	0.85
Stacked algorithm with RCSP	0.72 (1) [0.89]	0.57 (1) [0.92]	0.58 (1) [0.83]	0.64 (1) [0.61]	0.60 (0.87) [0.92]	0.33 (0.97) [0.67]	0.64 (0.92) [0.83]	0.83 (0.89) [0.94]	0.56 (0.97) [0.89]	0.61 (0.96) [0.83]
Stacked algorithm with CSP	0.17 (0.97) [0.91]	0.63 (1) [0.94]	0.63 (1) [0.80]	0.73 (0.94) []0.94	**0.94** (0.97) [0.88]	0.43 (0.94) [0.83]	0.77 (0.94) [0.94]	0.76 (0.97) [0.80]	0.32 (0.97) [0.94]	0.69 (0.96) [0.88]

Table 3
Summary of kappa values for Binary Class Problem: Left-Tongue

Method	*Left-Tongue Subjects*									
	A1	*A2*	*A3*	*A4*	*A5*	*A6*	*A7*	*A8*	*A9*	*Avg.*
SRLD Nicolas *et al* [20]	0.93	0.63	0.90	0.81	0.68	0.33	0.96	0.92	0.94	0.79
Stacked algorithm with RCSP	0.41 (0.97) [0.80]	0.50 (0.94) [0.69]	0.82 (0.83) [0.69]	0.57 (1) [0.89]	**0.81** (0.91) [0.77]	**0.72** (0.97) [0.86]	0.68 (0.80) [0.77]	0.65 (0.97) [0.92]	0.59 (0.86) [0.86]	0.64 (0.92) [0.72]
Stacked algorithm with CSP	0.63 (0.97) [0.88]	**0.69** (0.92) [0.91]	0.77 (0.92) [0.77]	0.18 (0.97) [0.77]	0.76 (0.88) [0.83]	0.61 (0.93) [0.77]	0.77 (0.97) [0.86]	0.86 (0.96) [0.74]	0.75 (0.97) [0.75]	0.66 (0.94) [0.80]

Table 4
Summary of kappa values for Binary Class Problem: Right-Foot

	Right-Foot									
Method	*Subjects*									
	A1	*A2*	*A3*	*A4*	*A5*	*A6*	*A7*	*A8*	*A9*	*Avg.*
SRLD Nicolas *et al* [20]	0.97	0.92	0.94	0.90	0.76	0.56	0.99	0.88	0.64	0.84
Stacked algorithm with RCSP	0.72 (0.90) [0.88]	0.75 (1) [0.91]	0.61 (0.97) [0.77]	0.77 (0.97) [0.88]	0.77 (0.93) [0.83]	0.48 (1) [0.75]	0.62 (1) [0.92]	0.54 (1) [0.74]	**0.70** (0.88) [0.86]	0.66 (0.96) [0.84]
Stacked algorithm with CSP	0.17 (0.93) [0.62]	0.58 (0.93) [0.76]	0.74 (0.94) [0.86]	**0.93** (1) [0.91]	**0.88** (0.96) [0.75]	**0.87** (0.93) [0.80]	0.61 (0.97) [0.75]	0.74 (0.91) [0.75]	0.55 (0.93) [0.82]	0.67 (0.94) [0.78]

Table 5
Summary of kappa values for Binary Class Problem: Right-Tongue

	Right-Tongue									
Method	*Subjects*									
	A1	*A2*	*A3*	*A4*	*A5*	*A6*	*A7*	*A8*	*A9*	*Avg.*
SRLD Nicolas *et al* [20]	0.99	0.53	0.94	0.86	0.86	0.36	1	0.75	0.83	0.79
Stacked algorithm with RCSP	0.42 (0.93) [0.85]	**0.97** (1) [0.86]	0.80 (0.97) [0.92]	0.34 (0.90) [0.86]	**0.91** (0.88) [0.94]	0.18 (0.94) [0.86]	0.60 (0.94) [0.80]	0.78 (1) [0.80]	0.69 (1) [0.89]	0.63 (0.95) [0.86]
Stacked algorithm with CSP	0.62 (0.91) [0.97]	0.94 (0.94) [0.94]	0.91 (0.96) [1]	0.58 (0.96) [0.85]	0.70 (0.93) [0.80]	**0.57** (0.97) [0.79]	0.68 (1) [0.83]	**0.86** (0.97) [0.72]	0.49 (0.91) [0.88]	0.70 (0.95) [0.86]

Table 6
Summary of kappa values for Binary Class Problem: Foot-Tongue

	Foot-Tongue									
Method	*Subjects*									
	A1	*A2*	*A3*	*A4*	*A5*	*A6*	*A7*	*A8*	*A9*	*Avg.*
SRLD Nicolas *et al* [20]	0.72	0.81	0.86	0.61	0.43	0.68	0.75	0.81	0.83	0.72
Stacked algorithm with RCSP	0.62 (0.96) [0.75]	0.57 (1) [0.77]	0.83 (0.97) [0.79]	**0.75** (1) [0.70]	**0.85** (0.90) [0.85]	**0.95** (0.90) [0.95]	0.75 (0.86) [0.74]	**0.83** (0.94) [0.83]	0.56 (1) [0.56]	**0.75** (0.95) [0.77]
Stacked algorithm with CSP	0.62 (0.91) [0.80]	0.59 (0.97) [0.82]	0.56 (0.96) [0.83]	0.62 (1) [0.83]	0.52 (0.85) [0.75]	0.82 (0.94) [0.75]	0.72 (0.97) [0.83]	0.85 (0.94) [0.86]	0.88 (0.97) [0.77]	0.68 (0.94) [0.80]

regularized CSP outperforms the standard in only few cases and the average result is higher for only one binary case (Foot-tongue). In all other binary class problems the CSP outperforms the regularized version. However, with only 15 features and 72 trials, the obtained results are remarkable. In SRLDA [20] they have taken 144 trials for binary class problems. The algorithm has scope for improvement. It can be extended for four-class classification, and be tested with various other datasets as well.

REFERENCES

[1] World Report on Disability 2011 from World Health Organization and The World Bank available at http://www.who.int/disabilities/world_ report/2011/report. pdf

[2] Jonathan R. Wolpaw, "Brain-Computer Interface Research Comes of Age: Traditional Assumptions Meet Emerging Realities", Journal of Motor Behavior, Vol. 42, No. 6, pp. 351-353, 2010.

[3] Jonathan R. Wolpaw, Niels Birbaumer, Dennis J. McFarland, Gert Pfurtscheller, Theresa M. Vaughan, "Brain-computer interfaces for communication and control", Elsevier Science Ireland Ltd, Clinical Neurophysiology 113, pp. 767–791, 2002.

[4] Cincotti. F, Mattia. D, Aloise. F, Bufalari. S, Schalk. G, Oriolo. G, Cherubini. A, Marciani. M.G, Babiloni. F, "Non-invasive brain-computer interface system: Towards its application as assistive technology", Robotics and Neuroscience, Vol. 75, pp. 796-803, April 2008.

[5] Roman. K, Benjamin. B, Gabriel. C, Klaus-Robert. M, "The Berlin Brain-Computer Interface (BBCI)-Towards a new communication channel for online control in gaming applications", Multimedia Tools and Applications 33, pp. 73–90, 2007.

[6] Aboul Ella Hussanien, and Ahmed Taher Azar," Brain Computer Interface Current Trends and Applications", Intelligent Systems Reference Library, Vol. 74, Springer Interational Publishing, 2015.

[7] Obed Carrera-Leon, Juan Manuel Ramirez, Vicente Alarcon-Aquino, Mary Baker, David DCroz-Baron, Pilar Gomez-Gil, "A Motor Imagery BCI Experiment using Wavelet Analysis and Spatial Patterns Feature Extraction", IEEE Conference on Engineering Applications (WEA), 2012 Workshop on, 2012.

[8] Katarzyna Blinowska, Piotr Durka, "Electroencephalography (EEG)", Wiley Encyclopedia of Biomedical Engineering, John Wiley & Sons, Inc., 2006.

[9] Yijun Wang, Ruiping Wang, Xiaorong Gao, Bo Hong, and Shangkai Gao, "A Practical VEP-Based Brain–Computer Interface", IEEE Transactions on Neural Systems and Rehabilitation Engineering, Vol. 14, No. 2, pp.234-239, June 2006.

[10] Yijun Wang, Xiaorong Gao, Bo Hong, Chuan Jia, and Shangkai Gao, "Brain-Computer Interfaces Based on Visual Evoked Potentials Feasibility of Practical System Designs", IEEE Engineering in Medicine and Biology Magazine, pp.64-71, September/October, 2008.

[11] J. R. Wolpaw, D. J. Mcfarland, and T. M. Vaughan, "Brain–Computer Interface Research at The Wadsworth Center", IEEE Transactions on Rehabilitation Engineering, Vol. 8, No.2, pp.222-226, June 2000.

[12] H. Yuan and B. He, "Brain-Computer Interfaces using Sensorimotor Rhythms: Current State and Future Perspectives", IEEE Transaction on Biomedical Engineering, Vol. 61, No. 5, pp. 1425–1435, May 2014.

[13] Sun-Yuge, Ye-Ning, Zhao-Lihong, Xu-Xinh, "Research on Feature Extraction Algorithms in BCI", IEEE, Chinese Control and Decision Conference (CCDC), pp.5874-587, 2009.

[14] Zhisong Wang, Alexander Maier, Nikos K. Logothetis, and Hualou Liang, "Extraction of Bistable-Percept-Related Features from Local Field Potential by Integration of Local Regression and Common Spatial Patterns", IEEE Transactions on Biomedical Engineering, Vol. 56, No. 8, pp. 2095-2103, August 2009.

[15] Dennis J. McFarland, Dean J. Krusienski and Jonathan R. Wolpaw, "Brain computer interface signal processing at the Wadsworth Center: mu and sensorimotor beta rhythms", Progress in Brain Research, Vol. 159, pp.411-419, February 2006.

[16] P. Shenoy, M.Krauledat, B.Blankertz, R.P.N.Rao, and K.R.Müller, "Towards Adaptive Classiûcation for BCI," Journal of Neural Engineering, Vol. 3, pp. R13, March 2006.

[17] L.F.Nicolas-AlonsoandJ.Gomez-Gil, "Brain Computer Interfaces, a Review," Sensors, Vol.12, pp.1211–1279, January 2012.

[18] M. Tangermann *et al*., "Review of the BCI competition IV," Frontal Neuroscience, Vol. 6, pp. 55, July 2012.

[19] K.K.Ang, Z.Y.Chin, C.Wang, C.Guan, and H.Zhang, "Filter Bank Common Spatial Pattern Algorithm on BCI Competition IV Datasets 2a and 2b," Frontal Neuroscience, Vol. 6, March 2012.

[20] Luis F.Nicolas-Alonso, Rebeca Corralejo, Javier Gomez-Pilar, Daniel Álvarez , and Roberto Hornero, "Adaptive Stacked Generalization for Multiclass Motor Imagery-Based Brain Computer Interfaces", IEEE Transactions on Neural Systems and Rehabilitation Engineering,Vol.23, No.4, pp. 702-712, July 2015.

[21] Wojciech Samek, Motoaki Kawanabe, and Klaus-Robert Muller , "Divergence-Based Framework for Common Spatial Patterns Algorithms", IEEE Reviews in Biomedical Engineering, Vol. 7, pp.50-72, 2014.

[22] B. Blankertz, R.Tomioka, S.L.M.Kawanabe, and K.R.Muller, "Optimizing Spatial Filters for Robust EEG Single-Trial Analysis", IEEE Signal Processing Magazine, Vol. 25, pp. 41–56, January 2008.

[23] Haiping Lu, How-Lung Eng, Cuntai Guan, Konstantinos N. Plataniotis, and Anastasios N. Venetsanopoulos, "Regularized Common Spatial Pattern with Aggregation for EEG Classification in Small-Sample Setting", IEEE Transactions on Biomedical Engineering, Vol. 57, No. 12, pp.2936-2946, December 2010.

[24] Fabien Lotte, and Cuntai Guan, "Regularizing Common Spatial Patterns to Improve BCI Designs: Unified Theory and New Algorithms", IEEE Transactions on Biomedical Engineering, Vol. 58, No. 2, pp.355-362, February 2011.

[25] Li Ke, Rui Li, "Classification of EEG Signals by Multi-Scale Filtering and PCA", IEEE, pp. 362-366, 2009.

[26] C.Vidaurre, M.Kawanabe, P.vonBunau, B.Blankertz, and K.Muller, "Toward Unsupervised Adaptation of LDA for brain–computer interfaces", IEEE Transaction on Biomedical Engineering, Vol.58, pp.587–597, March 2011.

[27] Z. Jinyin, G. Sudre, L. Xin, W. Wei, D. J. Weber, and A. Bagic, "Clustering Linear Discriminant Analysis for MEG-Based Brain Computer Interfaces", IEEE Transaction on Neural System and Rehabilitation Engineering, Vol.19, pp. 221–231, March 2011.

[28] D. Cai, X. He, and J. Han, "SRDA: An efficient Algorithm for Large Scale Discriminant Analysis," IEEE Transaction on Knowledge Data Engineering, Vol. 20, pp. 1–12, January 2008.