An Efficient P300 Speller System for Brain- Computer Interface

Conference Paper · September 2015

DOI: 10.1109/ISPCC.2015.7374998

3 authors:

Rahul Kumar Chaurasiya
National Institute of Technology Raipur
12 PUBLICATIONS 51 CITATIONS

SEE PROFILE

Narendra Londhe
National Institute of Technology Raipur
81 PUBLICATIONS 268 CITATIONS

SEE PROFILE

S. Ghosh
National Institute of Technology Raipur
71 PUBLICATIONS 261 CITATIONS

SEE PROFILE

An Efficient P300 Speller System for Brain-Computer Interface

Rahul Kumar Chaurasiya[1]*, Narendra D. Londhe[2], Subhojit Ghosh[2]
[1]Department of Electronics and Telecommunication, [2]Department of Electrical Engineering,
National Institute of Technology-Raipur, Raipur-India-492010
*(Corresponding Author), [1]rkchaurasiya@nitrr.ac.in

***Abstract*— P300 speller for Brain-Computer Interface systems aim to provide a direct communication between computer - machine and human brain, without any muscular activity. The communication is provided by detecting the presence of P300 Event Related Potentials (ERPs) in the electroencophelogram (EEG) signals, recorded from scalp. The major problem associated with P300 spellers is the stratification of EEGs recorded out of visual stimulation given to a subject. In this paper, we present a method to analyze the EEG data for P300 speller system using Support Vector Machines (SVM) classification technique. Using the proposed method, we are able to find a correct and faster solution for the "target character detection" associated with the P300 speller system. The method requires minimal preprocessing and provides a high transfer rate, which makes it suitable for online analysis also.**

***Index Terms*—P300 spellers, Brain-Computer Interface, SVM, Event Related Potentials.**

I. Introduction

Brain-Computer Interfacing (BCI) systems aim to develop a direct communication medium between a user and a computer machine [1]. Using BCI, we can send commands to a computer without any muscular activity. The BCI systems are the only way of communication for people suffering from various motor disabilities but still have cognitive abilities [2-5]. The BCI system involves recording conscious neural electrical activates via electroencephalograms (EEGs) signals, and performing signal processing and pattern recognition activities to detect what user wants to communicate.

Event-Related Potentials (ERPs) are the components of EEG that occur in response to some stimuli. A speller system based on the P300 ERP is commonly known as P300 speller. Here, P300 means a positive going ERP produced after 300 ms of stimulation. The first P300 ERP was recorded by Sutton [6] in 1965. If a user is asked to distinguish between two stimuli (presented in a random order), where one of them rarely occurs (the "odd-ball"), then a P300 wave is produced in response, and this phenomenon is known as "odd-ball paradigm" [7]. The task of P300 speller is to detect the presence of P300 ERP in the recorded EEG signal. The very first P300 speller for a visual matrix based input stimulus was proposed by Farewell and Donchin in 1988 [8]. Unfortunately, the P300 ERP has relatively very small amplitude (2-10 μV) in an EEG signal (10-100 μV). A signal averaging process over multi-trial EEG is in common practice to detect the presence of P300 ERP in order to detect the target character (the one, which user wants to communicate) from the speller matrix [9].

The paper is organized as follows- in next section we describe about the P300 speller system for visual input paradigms, section III describes the dataset used for the experimental purpose. In section IV we discuss the signal processing and classification methods used, the results of experimentation are presented in section V. We discuss and conclude the success and significance of our research work in section VI and VII respectively.

II. The Visual Paradigm Based P300 Speller System

A visual paradigm based P300 speller system consists of several stages: stimulating a subject by presenting a P300 display paradigm matrix; recording the EEG; signal pre-processing; feature extraction and classification [10]. A block diagram representation of the P300 speller system is given in Fig. 1.

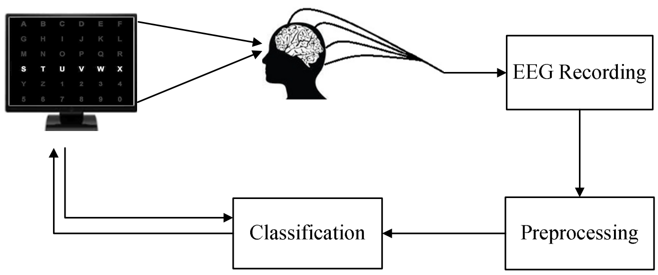


Fig. 1: A block diagram representation of P300 speller system.

After the first P300 speller system of Farewell and Donchin [8], various improvements have been witnessed in the speller systems in last couple of decades [11]. The variations in the display paradigm [12, 13], EEG signal recording devices, feature extraction methodologies, signal classification techniques [14], and channel selection algorithms [15] have made the P300 spellers more accurate and faster. BCI competitions II (2003) [16] and BCI competitions III (2004) [17] provide standard benchmark P300 speller dataset for analysis and to improve the P300 speller performance in general. Large number of research articles has been published analyzing P300 speller based on these competition dataset [18-24].

Inspite of the large volume of reported works mentioned above, the practice implementation of P300 speller is still quite challenging. This is because of the following facts:

1. EEG recording is still a very inconvenient task (especially for lengthy multi-trial acquisition).
2. Less reliability in the system
3. Slow transfer rate
4. Complexity in stimulation
5. Poor accuracy
6. Increasing classification challenges

In this paper, we have tried to address the aforementioned challenging issues related to P300 spellers. We have used a Gaussian kernel based Support Vector Machine (SVM) classification algorithm that can accurately detect the target character in relatively less number of trials. Necessity of less number of trials results in better accuracy with short length training session (and hence a better user convenience). The decimated and filtered amplitude values of EEG signals have been taken as feature coefficients for classification. The decimation process converts the lengthy signals into effective but compact and lesser dimensional feature vector. This results in faster classification and requires lesser size of training data. As a result, we are able to provide comparatively higher communication bit rate and classification accuracy. We have applied the proposed methodologies on the BCI Competition II (2003) dataset [16].

III. The Data Set for Experimental Analysis

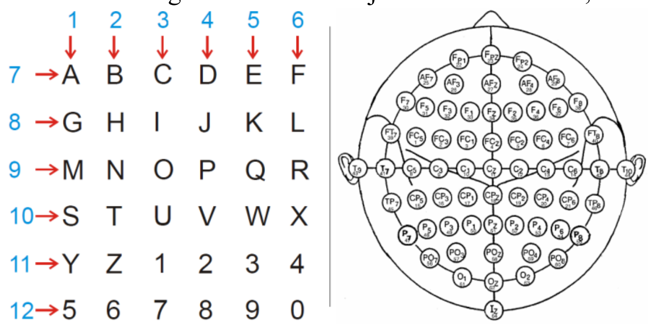
The 2nd Wadsworth BCI dataset [16] provides the record of P300 evoked potentials recorded with BCI2000 using the 6×6 sized matrix display paradigm of Farewell and Donchin [8]. The EEG signals were acquired using a standard 10-20 electrode placement system with 64 electrodes (see Fig. 2). The EEG signals were recorded from one subject in three sessions (viz. Session 10, 11 and 12), where each session was divided into number of runs. During the data recording, the user's task was to focus on a particular character (known as target character) out of 36 different characters. All the rows and columns of this matrix were randomly and successively intensified for 100 ms (followed by a 75 ms blank period). The intensification was block randomized in a set of 12 i.e. total 12 intensifications (6 rows and 6 columns) were provided in a random fashion such that each row and column get intensified exactly ones in each block. 2 out of 12 such intensification contain the target character. This block randomization process was repeated for 15 times for each target character (15 trials of same character).

For each run of session 10 and 11, the information of actual EEG signals (sampled at 240 Hz), *StimulusCode* (the row/ column number intensified) and *StimulusType* (the class label of signals containing target character) were provided. For session 12, *StimulusType* information was not provided. The user's task was to predict the target word for each run of session 12. For more details please go to [16].

IV. Method

For each block of intensification of 12 rows/columns, we have to identify one row and column, which has the stimulating character (i.e. target character). For this binary classification problem (of detecting the presence of P300 ERP component in the signal sequence), we have trained a SVM classifier. As the recorded evoked potentials are very noisy and include the background brain activities, it is not possible to detect the target character from just one trial. Hence, we have applied a multi-trial classification approach for detecting the target character. Based on the majority voting of the rows/columns we have decided the row and column number which is supposed to have the desired character. We have divided the whole process into following 3 stages:

- Data preprocessing and feature vector generation
- Training the SVM classifier.
- Detection of target characters in the test data.



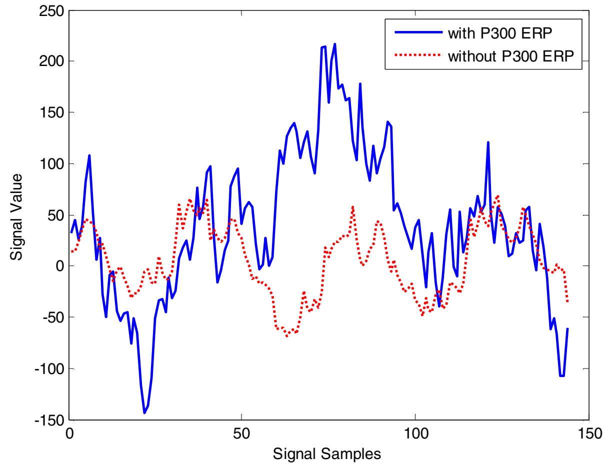
(a) (b)

Fig. 2: (a) The 36 character sized 6×6 matrix display paradigm used for data collection. The numbers 1-12 are the *StimulusCode* value for intensification of corresponding row/column. (b) Standard 10-20 electrode placement system for 64 channel EEG data acquisition.

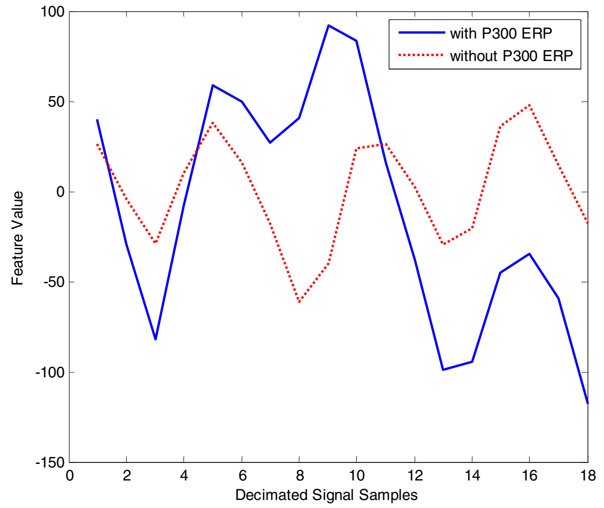
A. Data Preprocessing and Feature Vector Generation

As we are interested in the response occurred after 300 ms of the stimuli, for each epoch, we have taken the time windowed signal (from 0 ms to 600 ms) after the each stimulation. This window size is large enough to properly capture the response but short to contain the irrelevant information. As the data was recorded at the sampling rate of 240 Hz, a 600 ms duration signal gives 144 samples. In preprocessing, each signal has been filtered with an 8 order Chebyshev Type I filter with cut-off frequencies 0.1 and 30 Hz. Then the signals were decimated with 30 Hz frequency. After decimation, a 144 sample sized signal has been converted to 18 sample sized signal. The signal was then normalized to the interval [-1, 1]. Fig. 3. gives an example of sample variations of signals containing P300 ERP (containing the target character and hence class label +1) and without P300 ERP (not containing the target character and hence class label -1). We can notice that there is a positive going peak in the class +1 (blue) signals at around 300 ms after the stimulation (for Fig. 3(a). 300 ms corresponds to sample number 72, and for Fig. 3(b). 300 ms corresponds to sample number 9). This shows the presence of P300 ERP in the class +1 signals.

Afterwards, a signal has been converted into a feature vector, where each feature vector is a concatenation the preprocessed signals from all 64 channels for a given intensification of row/column (the normalized 18 samples, which we get after the decimation of filtered signals, are taken as feature coefficients for each channel). Hence, the size of a feature vector is 64×18=1152. Thus for a single character, the training set is composed of 180 (12 intensifications of rows/columns × 15 trials) feature vectors. In these 180 feature vectors, 30 are from class +1 (2 from each block of 12 × 15 trials) and remaining 150 are from class -1.



(a)



(b)

Fig. 3. (a) The sample variation of the raw signals (144 samples) with P300 ERP in blue color (class +1) and without P300 ERP in dotted-red (class label -1). (b) Corresponding sample variation of the preprocessed normalized signals. The plot is provided for the average of all the class +1 and class -1 signals of channel number 13 of run 1 of session 10. In both the images (a) and (b), we can notice that there is a positive going peak in the class +1 (blue) signals at around 300 ms after the stimulation. This shows the presence of P300 ERP.

We have used all the characters from all the runs of session 10 and 11 to prepare the training set. There are total 42 characters in session 10 and 11, so the training set consists of 7560 feature vectors (42×180), where each feature vector is of size 1152. Out of these 7560 feature vectors, 1260 vectors are from class +1 and remaining 6300 from class -1.

B. *Training the SVM Classifier*

The SVM classifier is based on the concept of providing the better *generalizing capabilities* of machines for classification [25]. A better generalizing capabilities of a classification algorithm indicates that a classifier has higher chances to perform equally well for the samples outside the training dataset. For this reason, SVM learns a hyper-plane that maximizes the separation *margin* between the dataset of two classes [26, 27].

Consider a hyper-plane of Eq. (1).

$$\boldsymbol{w}.\boldsymbol{x} + b = 0 \tag{1}$$

This hyper-plane is a separating hyper-plane between the two parallel hyper-planes of Eq. (2).

$$\boldsymbol{w}.\boldsymbol{x} + b = 1, \text{ and } \boldsymbol{w}.\boldsymbol{x} + b = -1 \tag{2}$$

The separation *margin* here is defined as is the Euclidian distance ${}^{1}/_{||\boldsymbol{w}||}$ between the hyper-planes of Eq. (2). Thus, the learning task in SVM for linearly separable data case is formalized as the following constrained optimization problem:

$$Min \frac{||\boldsymbol{w}||^2}{2}$$

$$\text{Subjected to } y_i(\boldsymbol{w}.\boldsymbol{x_i} + b) \geq 1, i = 1,2, \dots N \tag{3}$$

Here $(\boldsymbol{x_i}, y_i)$ $i = 1,2, \dots N$ is the training data to learn SVM. In order to generalizing the SVM for linearly non-separable data, we have to modify the formulation of Eq. (3) to learn a decision boundary that is tolerable to small training errors. This modification is performed by introducing positive valued slack variables (ξ) into the objective function and the constraints of the optimization problem of Eq. (3) to get Eq. (4).

$$\text{Minimize } L(\boldsymbol{w}, b, \xi_i) = \frac{1}{2}||\boldsymbol{w}||^2 + C\sum_{i=1}^{N}\xi_i$$

$$\text{Subjected to} \quad y_i(\boldsymbol{w}.\boldsymbol{x_i} + b) \geq 1 - \xi_i;\ i = 1,2, \dots N$$
$$\text{and} \quad \xi_i \geq 0 \tag{4}$$

The optimization problem of Eq. (4) will result in a hyper-plane providing the maximum separation margin between the data-set of two classes. Here, parameter C is a user defined parameter to penalize the number of training errors we are allowed to make. The problem of Eq. (4) is a convex optimization problem and is solved using the Lagrange's theorem [26, 28]. By solving the Eq. (4) we learn the weights as in Eq. (5) and a linear discriminant function as in Eq. (6).

$$\boldsymbol{w} = \sum_{i=1}^{N} y_i\lambda_i\boldsymbol{x_i} + b \tag{5}$$

$$f(x) = \sum_{i=1}^{N} y_i \lambda_i (\boldsymbol{x}.\boldsymbol{x_i}) + b \quad (6)$$

It is shown [27] that replacing the dot product $\boldsymbol{x}.\boldsymbol{x_i}$ of Eq. (6) by a Kernel function $K(\boldsymbol{x}.\boldsymbol{x_i})$, we can learn a non-linear discriminant function. In our two class classification problem with $y_i \in \{-1, 1\}$, we have applied both linear-discriminant function and a Gaussian Kernel (given in Eq. (7)) based non-linear discriminant function to classify the data.

$$K(\boldsymbol{x}.\boldsymbol{x_i}) = \boldsymbol{exp}(-\frac{||x-x_i||^2}{2\sigma^2}) \quad (7)$$

In our classification problem, we have trained the SVM classifier with full training dataset (i.e. the 1260 feature vectors of class +1 and 6300 feature vectors with class -1) using 5-fold *cross validation* method. The values of parameter C and σ (for learning Gaussian Kernel SVM) were varied systematically to get the best possible accuracy.

C. *Detection of Target Character in The Test Data*

After learning the SVM classifier with the dataset of Session 10 and 11, we have applied it to classify the dataset of Session 12. Ideally, we should get two features vectors with class +1 and 10 with class -1 (in a block of 12 intensifications) as a result of classification. But, the data is too noisy to obtain the correct symbol from only a single trial. So deal with this problem we applied a multi-trial approach. After each trial, we added +1 to the score of each row/column detected in class +1. (e.g. after 5 trials if the score of column 1 is 4, means in 5 trials the row 1 is 4 times classified in class +1). Thus after n number of trials, one out of size rows and one out of six columns (with the highest score in all rows and columns respectively) was classified to contain the desired character. The desired character was then inferred from the matrix of Fig. 2(a) for the given row and column number.

V. RESULTS

The first step of the classification problem was to learn a classifier to decide whether the given set of feature correspond to P300 ERP or not. Sensitivity and specificity are used as performance measuring parameter for this binary classification problem. In order to analyze the performances of linear SVM and Gaussian Kernel SVM, sensitivity (true positive ratio) and specificity (true negative ratio) are calculated by using confusion matrix. Equation (8) and (9) describe the formula used for calculating sensitivity and specificity using True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN).

$$Sensetivity = TPR = \frac{TP}{TP+FN} \times 100\% \quad (8)$$

$$Specificity = TNR = \frac{TN}{TN+FP} \times 100\% \quad (9)$$

The accuracy then was calculated using the Eq. (10), given below:

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \times 100\% \quad (10)$$

Using a 5-fold cross validation on training set, we are able to obtain 89.79 % accuracy with Gaussian Kernel SVM and 84.72% accuracy with linear SVM. These are the highest values of accuracy obtained with optimal set of the parameters C and σ. The detailed results of training-phase are summarized in Table-1.

TABLE I. THE TRAINING ACCURACY OF SVM CLASSIFICATION ALGORITHM WITH 5-FOLD CROSS VALIDATION

The Classifier used	**Optimal Parameter Values**		**Sensitivity**	**Specificity**	**Accuracy**
	C	σ			
Linear SVM	23.50	Not Applicable	82.12%	85.24%	84.72%
Gaussian Kernel SVM	25.00	29.00	87.50%	90.25%	89.79%

TABLE II. THE WORDS PREDICTED (WITH 1, 3 AND 5 TRIALS) AND ACTUAL WORDS FOR ALL THE RUNS OF SESSION 12

Run Number (Session 12)	**Word Predicted After**			**Actual Word**
	2 trials	**3 trials**	**5 trials**	
1	FOND	FOOD	FOOD	FOOD
2	MONT	MOOT	MOOT	MOOT
3	NGM	HGM	HAM	HAM
4	2IF	PIE	PIE	PIE
5	CGLQ	CGKE	CAKE	CAKE
6	TONA	TONA	TUNA	TUNA
7	ZYNOZ	ZYNOZ	ZYGOT	ZYGOT
8	X567	X567	4567	4567
Accuracy Obtained=	**54.83%**	**80.64%**	**100%**	**--**

After learning the SVM with training data, it was applied to detect the target words of each of the 12 runs of session 12. We have applied Gaussian Kernel SVM classification approach for detecting the target character, as it produced better results in the training phase. We tried to predict the target word with different number trials. We were able to obtain 100% accuracy (in predicting the target characters) just after 5 trials, and hence, we did not try to predict the words with more than 5 trials. The results of the words obtained after first, third and fifth trial for different runs of Session 12 are presented in the Table II.

VI. DISCUSSIONS

The aim of our research work was to provide an efficient P300 speller system that addresses the issues of user-inconvenience, poor accuracy of classification and slow transfer rate. In this regards, a novel technique based on supervised SVM classifier has been proposed.

Firstly, using the proposed classification scheme on the described set of features and using the data captured from all 64 channels, we are able to correctly predict the target character in just 5 trials. The requirement of only 5 trials (compared to 15, which are provided for classification in) ensures lesser execution time for recording the data set during test run. Thus, we are successful in increasing the user convenience by decreasing the recording time requirement.

Secondly, we have applied an SVM classification scheme to detect the presence of P300 ERP in the EEG signals and achieved 100% accuracy. The results ensure that we would be able to achieve a high accuracy with new test samples as well.

Thirdly, we have extracted the features based on the amplitude of the EEG signals, which turns out to be very simple and affective set of features for classification. Another advantage of our method is low computation complexity, which makes the proposed method suitable for online applications. Finally, using the combination of easily extractable feature set and low processing requirement, we are able to achieve high communication rate, thus successfully addressing the issue of low transfer rate.

VII. CONCLUSION

The challenges related to user-convenience, accuracy and transfer rates make the task of designing a reliable P300 speller quite challenging. Motivated by this, in this paper, we have proposed a SVM based P300 speller system. A proper choice of features and systematically learned Gaussian Kernel SVM classification technique provide high accuracy and transfer rates with better user convenience. The efficiency of the algorithm has been proven by applying it on widely used BCI competition dataset.

At the cost of increase computation (and hence with decreased transfer rate), an improvement in the classification performance may be achieved by using more complex set features. Selecting the set of most relevant channels (for classification) from all 64 channels may also lead to better performance and can be considered as the future work for the presented method.

REFERENCES

[1] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain-computer interfaces for communication and control," *Clin Neurophysiol,* vol. 113, no. 6, pp. 767-91, Jun, 2002.

[2] J. R. Wolpaw, D. J. McFarland, and T. M. Vaughan, "Brain-computer interface research at the Wadsworth Center," *Rehabilitation Engineering, IEEE Transactions on,* vol. 8, no. 2, pp. 222-226, 2000.

[3] E. A. Curran, and M. J. Stokes, "Learning to control brain activity: a review of the production and control of EEG components for driving brain-computer interface (BCI) systems," *Brain Cogn,* vol. 51, no. 3, pp. 326-36, Apr, 2003.

[4] T. M. Vaughan, W. J. Heetderks, L. J. Trejo, W. Z. Rymer, M. Weinrich, M. M. Moore, A. Kubler, B. H. Dobkin, N. Birbaumer, E. Donchin, E. W. Wolpaw, and J. R. Wolpaw, "Brain-computer interface technology: a review of the Second International Meeting," *IEEE Trans Neural Syst Rehabil Eng,* vol. 11, no. 2, pp. 94-109, Jun, 2003.

[5] N. Birbaumer, "Brain-computer-interface research: coming of age," *Clin Neurophysiol,* vol. 117, no. 3, pp. 479-83, Mar, 2006.

[6] S. Sutton, M. Braren, J. Zubin, and E. R. John, "Evoked-potential correlates of stimulus uncertainty," *Science,* vol. 150, no. 3700, pp. 1187-8, Nov 26, 1965.

[7] E. Donchin, "Presidential address, 1980. Surprise!...Surprise?," *Psychophysiology,* vol. 18, no. 5, pp. 493-513, Sep, 1981.

[8] L. A. Farwell, and E. Donchin, "Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials," *Electroencephalogr Clin Neurophysiol,* vol. 70, no. 6, pp. 510-23, Dec, 1988.

[9] S. Sanei, and J. A. Chambers, *EEG signal processing*: John Wiley & Sons, 2013.

[10] M. Akcakaya, B. Peters, M. Moghadamfalahi, A. R. Mooney, U. Orhan, B. Oken, D. Erdogmus, and M. Fried-Oken, "Noninvasive brain-computer interfaces for augmentative and alternative communication," *IEEE Rev Biomed Eng,* vol. 7, pp. 31-49, 2014.

[11] R. Fazel-Rezai, B. Z. Allison, C. Guger, E. W. Sellers, S. C. Kleih, and A. Kubler, "P300 brain computer interface: current challenges and emerging trends," *Front Neuroeng,* vol. 5, pp. 14, 2012.

[12] R. Fazel-Rezai, and W. Ahmad, *P300-based Brain-Computer Interface paradigm design*: INTECH Open Access Publisher, 2011.

[13] R. Fazel-Rezai, and K. Abhari, "A comparison between a matrix-based and a region-based P300 speller paradigms for brain-computer interface." pp. 1147-1150.

[14] F. Lotte, M. Congedo, A. Lecuyer, F. Lamarche, and B. Arnaldi, "A review of classification algorithms for EEG-based brain-computer interfaces," *J Neural Eng,* vol. 4, no. 2, pp. R1-R13, Jun, 2007.

[15] D. Feess, M. M. Krell, and J. H. Metzen, "Comparison of sensor selection mechanisms for an ERP-based brain-computer interface," *PLoS One,* vol. 8, no. 7, pp. e67543, 2013.

[16] B. Blankertz, K. R. Muller, G. Curio, T. M. Vaughan, G. Schalk, J. R. Wolpaw, A. Schlogl, C. Neuper, G. Pfurtscheller, T. Hinterberger, M. Schroder, and N. Birbaumer, "The BCI Competition 2003: progress and perspectives in detection and discrimination of EEG single trials," *IEEE Trans Biomed Eng,* vol. 51, no. 6, pp. 1044-51, Jun, 2004.

[17] B. Blankertz, K. R. Muller, D. J. Krusienski, G. Schalk, J. R. Wolpaw, A. Schlogl, G. Pfurtscheller, R. Millan Jdel, M. Schroder, and N. Birbaumer, "The BCI competition. III: Validating alternative approaches to actual BCI problems," *IEEE Trans Neural Syst Rehabil Eng,* vol. 14, no. 2, pp. 153-9, Jun, 2006.

[18] V. Bostanov, "BCI Competition 2003--Data sets Ib and IIb: feature extraction from event-related brain potentials with the continuous wavelet transform and the t-value scalogram," *IEEE Trans Biomed Eng,* vol. 51, no. 6, pp. 1057-61, Jun, 2004.

[19] A. Rakotomamonjy, and V. Guigue, "BCI competition III: dataset II- ensemble of SVMs for BCI P300 speller," *IEEE Trans Biomed Eng,* vol. 55, no. 3, pp. 1147-54, Mar, 2008.

[20] N. Xu, X. Gao, B. Hong, X. Miao, S. Gao, and F. Yang, "BCI competition 2003-data set IIb: enhancing P 300 wave detection using ICA-based subspace projections for BCI applications," *IEEE transactions on biomedical engineering,* vol. 51, no. 6, pp. 1067-1072, 2004.

[21] M. Kaper, P. Meinicke, U. Grossekathoefer, T. Lingner, and H. Ritter, "BCI Competition 2003--Data set IIb: support vector machines for the P300 speller paradigm," *IEEE Trans Biomed Eng,* vol. 51, no. 6, pp. 1073-6, Jun, 2004.

[22] H. Cecotti, and A. Graser, "Convolutional neural networks for P300 detection with application to brain-computer interfaces," *IEEE Trans Pattern Anal Mach Intell,* vol. 33, no. 3, pp. 433-45, Mar, 2011.

[23] S. Lemm, C. Schafer, and G. Curio, "BCI Competition 2003--Data set III: probabilistic modeling of sensorimotor mu rhythms for classification of imaginary hand movements," *IEEE Trans Biomed Eng,* vol. 51, no. 6, pp. 1077-80, Jun, 2004.

[24] J. Long, Z. Gu, Y. Li, T. Yu, F. Li, and M. Fu, "Semi-supervised joint spatio-temporal feature selection for P300-based BCI speller," *Cogn Neurodyn,* vol. 5, no. 4, pp. 387-98, Nov, 2011.

[25] A. K. Jain, R. P. W. Duin, and J. Mao, "Statistical pattern recognition: A review," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 22, no. 1, pp. 4-37, 2000.

[26] S. Theodoridis, and K. Koutroumbas, *Pattern Recognition, Fourth Edition*: Academic Press, 2008.

[27] C. J. Burges, "A tutorial on support vector machines for pattern recognition," *Data mining and knowledge discovery,* vol. 2, no. 2, pp. 121-167, 1998.

[28] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification*: John Wiley & Sons, 2012.