Comparison of Pre-Processing and Classification Techniques for Single-Trial and Multi-Trial P300-Based Brain Computer Interfaces

Chanan S. Syan and Randy E.S. Harnarinesingh
Department of Mechanical Engineering,
University of the West Indies, St. Augustine, Trinidad

Abstract: The P300 component of Event Related Brain Potentials (ERP) is commonly used in Brain Computer Interfaces (BCI) to translate the intentions of an individual into commands for external devices. The P300 response, however, resides in a signal environment of high background noise. Consequently, the main problem in developing a P300-based BCI lies in identifying the P300 response in the presence of this noise. Traditionally, attenuating the background activity of P300 data is done by averaging multiple trials of recorded signals. This method, though effective, suffers two drawbacks. First, collecting multiple trials of data is time consuming and delays the BCI response. Second, latency distortions may appear in the averaged result due to variable time-locking of the P300 in the individual trials. **Problem statement:** The use of single-trial P300 data overcomes both these shortcomings. However, single-trial data must be properly denoised to allow for reliable BCI operation. Single-trial P300-based BCIs have been implemented using a variety of signal processing techniques and classification methodologies. However, comparing the accuracies of these systems to other multi-trial systems is likely to include the comparison of more than just the trial format (single-trial/multi-trial) as the data quality and recording circumstances are likely to be dissimilar. **Approach:** This issue was directly addressed by comparing the performance comparison of three different preprocessing agents and three classification methodologies on the same data set over both the single-trial and multi-trial settings. The P300 data set of BCI Competition II was used to facilitate this comparison. **Results:** The LDA classifier exhibited the best performance in classifying unseen P300 spatiotemporal features in both the single-trial (74.19%) and multi-trial format (100%). It is also very efficient in terms of computational and memory requirements. **Conclusion:** This study can serve as a general guide for practitioners developing single-trial and multi-trial P300-based BCI systems, particularly for selecting appropriate pre-processing agents and classification methodologies for inclusion. The possibilities for future study include the investigation of double-trial and triple-trial P300 system based on the LDA classifier. The time savings of such approaches will still be significant. It is very likely that such systems would benefit from accuracies higher than the one obtained in this study for single-trial LDA (74.19%).

Key words: Electroencephalography, P300, principal component analysis, genetic algorithm, linear discriminant analysis

INTRODUCTION

A Brain Computer Interface (BCI) is a system that permits users to control external devices using only their inherent brain activity. Device control is achieved by performing a cognitive or physical task that encodes the command to be executed. For example, the user imagines moving his left hand or attends to a stimulus in order to control the movement of a cursor. BCIs are responsible for recording and pre-processing brain activity, extracting descriptive features from the data and then classifying these features to identify the user's command. The most popular medium of brain-computer communication is Electroencephalography (EEG).

The features of brain activity that are commonly used for Brain-Computer Interfacing are sensorimotor rhythms (Peters *et al.*, 2001; Krusienski *et al.*, 2007) slow cortical potentials (Birbaumber *et al.*, 2003) and visually evoked responses (Lin *et al.*, 2007; Citi *et al.*, 2008). Visually evoked responses can either be oscillatory neuronal responses to repetitively delivered

Corresponding Author: Chanan S. Syan, Department of Mechanical Engineering, University of the West Indies, St. Augustine, Trinidad

stimuli or delayed positive deflections in the EEG following the presentation of target stimuli. The latter is termed the P300 Visual Evoked Potential (P300 VEP). P300-based BCIs have been utilized for cursor control (Trejo *et al*., 2006), spelling systems (Wills and MacKay, 2006) and wheelchair navigation (Pires *et al*., 2008; Rebsamen *et al*., 2007).

The P300 wave was first reported in 1965 (Andrews *et al*., 2008). It appears as a positive deflection in the EEG approximately 300-400 m sec following the presentation of a rare, deviant or target stimulus. It is measured strongly in midline sites (C_Z, F_Z and P_Z) and resides mainly in the 0-8 Hz band (Khosrow-Pour, 2009). Both the latency and amplitude of the P300 wave correlate with the user's level of fatigue and the saliency (brightness and color) of the stimulus. The P300 response can be evoked through the visual, auditory or somatosensory modalities, however, most studies rely on the visually evoked version (Citi *et al*., 2008; Serby *et al*., 2005; Zhang *et al*., 2008; Nijboer *et al*., 2008).

The P300 potential evoked using an oddball paradigm. In an oddball paradigm, a target stimulus which represents the user's command is presented among more frequently occurring non-target stimuli. Attending to the target stimulus causes the P300 to be evoked which allows the BCI to identify the user's message. Stimulus attendance equates to mere visual fixation or keeping a mental count of the amount of times the target is highlighted, as in the case of the P300 speller paradigm (Farwell and Donchin, 1988).

P300-based BCIs benefit both from simplicity and ease of use. First, evoking the P300 requires the subject to focus on the appropriate stimulus, which consumes minimal physical and cognitive resources. Second, since the P300 is an inherent response of the brain, subjects require minimal training before they can operate a P300 based-BCI. This is not always the case with other brain signals such as Slow Cortical Potentials (SCP). Extensive training periods are often required before such brain signals become identifiable and thus become able to facilitate brain-computer communication.

However, the major limitation of the P300 signal is its Signal-to-Noise Ratio (SNR), owing to its corruption by powerful background noise. P300 signal denoising is traditionally carried out using batch averaging of signals recorded in multiple trials. In on-line applications, trails often need to be repeated until the measured P300 value attains statistical significance (Serby *et al*., 2005). However, recording multiple trials of data is time consuming and is manifested as lengthy delays in BCI processing. Additionally, the latency of

the P300 response may vary in each trial, which can lead to latency distortion of the averaged result (Andrews *et al*., 2008).

Single-trial based P300 BCIs suffer neither of these shortcomings. However, single-trial data must be properly preprocessed to allow for reliable BCI operation. Single-trial P300-based BCIs have been developed using a variety of signal processing techniques and classification methodologies. However, no formal catalogue or comparative analysis of these methods exists. This study directly addresses this issue by presenting a comprehensive review of a host of processing and classification techniques which have been used in both the single-trial and multi-trial settings. Additionally, the P300 data set of BCI Competition II (Schalk *et al*., 2004) is used to facilitate a performance comparison of three separate classifiers using various preprocessing agents in both the single-trial and multi-trial format.

MATERIALS AND METHODS

P300 Pre-processing techniques:
Trial averaging: Averaging multiple trials of data is one method by which time-domain features such as the P300 can be pre-processed. According to the central limit theorem (Yuehua *et al*., 2008), the average of n instances of a sample drawn from a population has a variance of σ^2 where:

$$\sigma^2 = \frac{\sigma_{population}^2}{n} \tag{1}$$

Assuming the target P300 signal in each trial is constant, averaging multiple trials can reduce the variance (energy) of signal contaminants and leave the target signal unaltered. Pires *et al.* (2008) compare the effect of changing the amount of averaged trials on the error rate for a P300-based BCI using Bayesian Classification. They registered a monotonic decrease in the false positive, false negative and error rate as the number of averaged channels increase. Their results highlight the efficacy of the trial averaging approach.

However, there are drawbacks to this approach. The collection of multiple trails followed by a computation of their mean is time-consuming. The delay can be reduced by averaging fewer trials however this reduces the factor of noise attenuation. In addition, trial averaging only works well if the signal that undergoes averaging is stationary. For the P300, whose peak value and latency can vary in every trial, averaging can lead to data distortions.

Spatial filtering: Spatially distinct data sources, of different noise and target signal content, can be combined to create a single channel of high SNR. Such functions are called spatial filters. Formally defined, a spatial filter is a function that operates on signals originating at different points in space at the same instant in time.

Examples of spatial filters used in BCI development are the Laplace filter, Local Average Technique (LAT) and the Common Average Reference (CAR) (Peters *et al*., 2001). By definition, bipolar and mastoid referenced EEG data streams are also spatial filters since they produce an output by subtracting channels from a spatially distinct reference.

The Laplace and LAT filter operate only on adjacent channels whereas the CAR filter uses the entire data array. All of the filters, however, implement a form of mean value removal. This represents an effort to reduce the noise content of the data by using noise samples from multiple channels. In (Peters *et al*., 2001), the performance of each filter for Artificial Neural Network classification of a 3-class intention of motion task is compared. The LAT filter performed worse than no filter. Additionally, the Laplace and CAR filters showed equal performance, yielding 98% BCI classification accuracy.

Spatial filters are a feasible denoising option when multiple channels of data are present. However, as their transfer functions are constant and insensitive to the input data, they are suboptimal at noise removal.

Principal component analysis: Transformation of the recorded data onto an orthogonal space is one method by which data can be decorrelated. This has the ability to fastrack the identification of noise and target components in the data. Principal Component Analysis (PCA) performs such a transformation (Pearson, 1901). PCA re-references multidimensional data to a new orthogonal basis such that there is no inter-channel covariance. Consequently, the covariance matrix of the transformed data set is diagonal.

PCA can be used in two ways: As a data compression tool or a pre-processing agent. PCA performs data compression if some of the PCs are rejected and the others retained with no transformation back to the original space. Lenhardt *et al*. (2008) use PCA in this manner for P300 data. Alternatively, PCA acts as a pre-processing agent if the original data is reconstructed following the stage of PC rejection. In this case, even though the dimensionality in the PC space has been reduced, the pre-processed data is of the same dimensionality as it was originally. Andrews *et al.* (2008) use PCA in this manner for single-trial P300 data.

The most common PC rejection criteria are the Residual Power (RP) and Kaiser method. The RP method retains the cumulative PCs that account for 95% of the original data variance, whereas the Kaiser (KSR) method retains only those PCs whose variances are greater than 1. BCIs designs that incorporate PCA as a pre-processing tool have reported classification accuracies of 100% (Sellers *et al*., 2006).

P300 classification methodologies:
Statistical classifiers: Statistical classifiers rely explicitly on class probability functions for feature categorization. The three statistical classification methodologies commonly implemented are: Maximum Likelihood (ML), Maximum A Posteriori (MAP) and General Bayes (GB). Each method involves the maximization or minimization of a discriminant function that provides a probabilistic measure of class membership.

For an n-class classification problem, the ML method classifies an observation (feature vector) x according to the rule:

$$\text{class} = \arg\max_{n} P(x \mid C_N) \qquad (2)$$

$P(x|C_N)$, termed likelihood, is the conditional probability that observation x will occur given that a sample is drawn from class C_N. The term "Maximum Likelihood" stems from the fact that the discriminant function of likelihood is maximized in order to determine class membership. The ML algorithm is advantageous since it benefits from both computational and conceptual simplicity. In (Serby *et al*., 2005), ML is used for P300-based BCI.

The major drawback of this approach is derived from the lack of consideration given to the proportion of class exemplars in training data. For example, consider the two-class classification problem where the sample space consists of 100 observations from class 1 and 20 observations from class 2. The probability that a sample is drawn is class 1 is therefore 5 times greater than the probability a sample from class 2 would be drawn. It is intuitive to expect that the classification boundary would be shifted in favor of class 1. However, this is not considered by the ML algorithm.

The probabilities that embody the proportion of class exemplars in the training set are referred to as priors and are denoted as $P(c_N)$, where $P(c_N)$ is the probability that a member of class n is chosen. Unlike the ML classifier, the MAP rule utilizes both class priors and likelihoods for classification. The MAP rule classifies observation x according to the rule:

$$\text{class} = \arg\max_{n} P(C_N \mid x) \qquad (3)$$

$P(C_N|x)$, referred to as posterior probability, is the probability that an observation is drawn from class n given that the observation is x. This is a better measure of class membership than likelihood. Posterior probabilities are determined using Bayes theorem. Bayes theorem states:

$$p(C_N \mid x) = \frac{p(x \mid C_N).p(C_N)}{p(x)} \qquad (4)$$

Since the denominator of the expression is constant for a given observation x, the classification rule is simplified to:

$$\text{class} = \arg\max_{n} p(C_N).p(x \mid C_N) \qquad (5)$$

For equal class priors, the MAP rule is equivalent to the ML rule. The method of k-Nearest Neighbors (kNN) permits the

Linear discriminant analysis: Linear Discriminant Analysis (LDA) is a machine learning method which seeks a linear transformation of features onto a 1-dimentional space that maximizes class separation. The transformation is sought such that inter-class mean distance is maximized and intra-class variance is minimized. This allows the classes to be separated by a point in 1-dimensional space.

The LDA projection of feature vector x can be expressed as a vector dot product:

$$Y = w.x \qquad (6)$$

where, w is the projection vector.

The optimal global value for the pre-multiplying projection vector w can be found using vector calculus. In this regard, LDA is better than Artificial Neural Networks (ANN) as it is common for ANNs to terminated at a local minima.

Genetic algorithms: High-dimensional feature sets based on P300 data trials are likely to contain features that correlate well with the P300 component. However, locating the optimal feature subset for a given classifier is often manifested as an optimization problem riddled with discontinuities and non-linearities. Consequently, the analytical methods of gradient descent/ascent become inapplicable. Genetic Algorithms (GA) however, are ideally suited for this sort of problem.

Evolutionary Algorithms (EA) (Fogel, 2005) are search and optimization techniques inspired by the mechanics of natural selection. Genetic Algorithms (GA) are a type of EAs. A GA is initialized by generating multiple random solutions to an optimization problem. These solutions which are referred to as individuals are evaluated to determine their fitness at solving the problem at hand. The fitter individuals are permitted a greater opportunity to produce new individuals, termed offspring, which populate a new generation of solutions. This process is reiterated until a predefined stopping criterion is met.

In (Citi *et al.*, 2008), a GA is used to locate the optimal subset of joint-domain time-space-frequency features for single-trial P300 data

Comparison of processing and classification techniques: The P300 EEG dataset of the BCI Competition II (Schalk *et al.*, 2004) is used to evaluate and compare the performances of the reviewed processing and classification techniques. A full description of the EEG recording circumstances as well as the visual stimulus presentation paradigm is available online (Schalk, 2002). For each of the following techniques, each of the 64 EEG channels was digitally filtered using a 10th order low-pass Hamming-window filter with 6 dB cutoff at 30Hz.

LDA:
Single-trial: EEG was collected 0-375 m sec following the flashing of each row/column for all 15 trials for all characters. There are therefore 180 (12×15) EEG segments associated with each character; 90 row segments and 90 column segments.

4-dimensional temporal feature vectors were extracted from 16 channels (FC1, FC4, FC6, C6, CP2, FP1, F2, F6, FT8, T7, TP7, PZ, PO7, POZ, PO8 and OZ) by down-sampling the time segment of 200-375 m sec post-stimulus by a factor of 14. Subsequently, the features were concatenated to produce one 64-dimensional spatiotemporal feature vector. 180 feature vectors were extracted from 42 characters (7560 features) and used to train the LDA classifier. However, for the purpose of classifier testing, only a single-trial was used.

The classification of each character was treated as two 6-class classification problems even though the LDA classifier was trained using two classes (P300 present and P300 absent). For each character, 6 row feature vectors and 6 column feature vectors were extracted. The 64-dimensional pre-multiplying LDA projection vector was used to convert the 12 features into 12 1-d values. The target row, that is, the row

which contained the target character was classified as that row which has the largest 1-d value. The columns were classified in a similar manner.

Multi-trial: EEG was collected 0-375 m sec following the flashing of each row/column for 15 trials and then averaged. There are therefore 12 EEG segments associated with each character; 6 row segments and 6 column segments. The feature extraction stage of the single-trial approach was used to extract 12 feature vectors from each character leading to a total number of 504 feature vectors from 42 characters. The rows and columns were classified in the same manner as the single-trial approach.

PCA: PCA was used as a pre-processing tool using both the RP and KSR rejection criteria. PCA was implemented on the entire EEG runs as a whole and not on the individual post-stimulus EEG segments that is sometimes done.

Genetically Optimized Spatial Filtering (GOSF): The techniques of genetic algorithms and spatial filtering were combined to produce one united preprocessing and classification methodology. A GA was used to obtain the optimal spatial filter for a simple classifier.

EEG was collected 300-400 m sec following the delivery of row/column stimuli. 12 EEG segments were therefore extracted from each character. Each segment was spatially filtered, resulting in one discrete-time series. The spatial filtering is represented by the matrix-vector multiplication:

$$Y = X.s \tag{7}$$

Where:
$X = 24 \times 64$ matrix
$s = 64 \times 1$ vector

The P300 feature is taken as the most positive value in the resulting discrete-time series (Y). Therefore, for each character, there are 12 associated features: 6 row features and 6 column features. The row with the largest P300 feature is classified as the target row, i.e., the row that contains the character to which the attention is paid. The columns are similarly classified. Identification of the target row and target column uniquely identifies the character to which attention was paid.

Statistical classifiers: Class probability functions are required to implement the statistical classifiers reviewed earlier in the text. However, the posterior probability can be directly estimated using the technique of K-Nearest Neighbors (KNN) thus making it possible to implement the MAP classifier. In order to classify a given feature vector, the KNN algorithm searches for the K closest feature vectors from the training set in Euclidean space. Features were extracted in the same manner as they were for the LDA classifier. K was chosen to be 10 after some preliminary simulations.

The posterior probabilities for class K is then given as:

$$P(C_K \mid x) = \frac{n_K}{K} \tag{8}$$

RESULTS

In addition to frequency filtering, three pre-processing instances were employed for the performance comparison. They are RP-PCA, KSR-PCA and no further preprocessing. They were implemented for the three classification methodologies presented earlier in the text in both the Single-Trial (ST) and Multi-Trial (MT) settings. Percentage accuracy for both the training data and the unseen test data for all possible classification/preprocessing combinations are provided in Table 1. As aforementioned, the training data consists of 42 alphanumeric characters (84 6-class classification problems) whereas the test data consists of 31 alphanumeric characters (62 6-class classification problems). The test data was not seen by the classifier and as such provides an unbiased measure of classifier generalization.

Table 1: P300 classification accuracy for LFA, GOSF and KNN classifiers over a range of preprocessing agents in both the single-trial and multi-trial settings

Method	Format	Pre-processing	Accuracy	
			Training data (%)	Test data (%)
LDA	ST	NONE	63.10	75.81
LDA	MT	NONE	98.81	100.00
LDA	ST	RP-PCA	52.38	54.84
LDA	MT	RP-PCA	98.81	100.00
LDA	ST	KSR-PCA	63.10	75.81
LDA	MT	KSR-PCA	98.81	100.00
GOSF	ST	NONE	41.67	35.48
GOSF	MT	NONE	95.23	80.65
GOSF	ST	RP-PCA	36.90	30.62
GOSF	MT	RP-PCA	70.24	64.52
GOSF	ST	KSR-PCA	41.67	35.48
GOSF	MT	KSR-PCA	95.23	80.65
KNN	ST	NONE	40.48	35.48
KNN	MT	NONE	84.52	80.65
KNN	ST	RP-PCA	48.81	37.10
KNN	MT	RP-PCA	88.10	79.03
KNN	ST	KSR-PCA	40.48	35.48
KNN	MT	KSR-PCA	84.52	80.65

ST: Single-Trial; MT: Multi-Trial

DISCUSSION

Predictably, single-trial approaches perform worse than multi-trial approaches. The mean classification accuracy of the single-trail approaches across all classification and pre-processing techniques for unseen test data was 47.62% compared to 85.13% for the multi-trial format. The majority of single-trial approaches performed poorly (<50%) except LDA which exhibited a classification performance of 75.81% for unseen test data with no additional preprocessing besides frequency filtering.

For the majority of cases, RP-PCA attained worse classification performance than no preprocessing agent. This was so for all instances except that of the KNN classifier on single-trial data where the classification accuracy for RP-PCA (37.10%) was slightly better than no preprocessing agent. The general poor performance of the PCA algorithm using the RP PC rejection criterion is consistent with the findings of (Andrews *et al.*, 2008). KSR-PCA performed the same as no preprocessing agent for every possible preprocessing/classification instance. KSR-PCA entails the rejection of PCs whose variances are less than 1. However, PCA was performed on the entire data set and not just the individual extracted segments. As such, no PC had a variance which was less than 1. Therefore, KSR-PCA resulted in the rejection of no PCs which is equivalent to no additional preprocessing.

With regard to different classification methodologies, LDA outperformed both GOSF and KNN with a mean classification accuracy of 84.41% on unseen test data. LDA was the only classifier to attain 100% classification accuracy with no additional preprocessing on the multi-trial setting. In contrast, the GOSF and KNN classifiers achieved mean accuracies of 54.57 and 58.07% respectively. Of these three classifiers, GOSF is the most computationally and memory intensive and takes hours on average to execute. The KNN classifier is also memory-intensive as all training examples need to be stored in order to implement the classifier. For this reason, it is considered to be a computationally greedy algorithm. However, the LDA classifier executes in a matter of seconds and has minimal memory requirements.

In one rare instance, the single-trial LDA classifier outperformed a multi-trial approach. This is significant given the powerful preprocessing efficacy of trial averaging. As the main attraction of single-trial P300 BCI operation is it's time saving ability, the LDA classifier is ideal for single-trial operation if a fractional drop (25%) in system accuracy is tolerable to the specific application. It must be noted however, that the classification accuracy and communication delay are application-specific requirements. For cursor control, time is not as important as accuracy as wrong moves can be easily compensated for in subsequent actions. However, for wheelchair navigation, the communicated commands are likely to be success-critical.

Additionally, adaptive stimulus presentation schemes were observed in some multi-trial P300-based BCIs. These schemes limit the amount of presentations/trials based on the quality of the collected signal (Serby *et al.*, 2005; Pearson, 1901). They do not present the time savings of single-trial operation or the classification accuracy of multi-trial designs, but they are a reasonable middle ground for performance/speed tradeoff. It will be worthwhile to examine the performance of LDA in the double-trial and triple-trial P300 setting as this mode will likely offer high accuracies along with significant time savings.

CONCLUSION

In this study, a number of P300 processing techniques and classification methodologies were compared using the P300 data set of BCI Competition II in both the single-trial and multi-trial settings. Single-trial P300 operation presents significant time savings to BCIs compared to the conventional multi-trial averaging approach. Furthermore, the single-trial mode of operation averts the problem of latency distortion associated with trial averaging. Predictably, the single-trial approaches performed worse in general than the multi-trial approaches. However, the LDA classifier exhibited a classification accuracy of 75.17% in the single-trail setting with no pre-processing besides forward frequency filtering. It is also relevant to note that this accuracy surpassed the results of some multi-trial setups. This is significant considering that the multi-trial setting entails the averaging of 15 trials. It may be worthwhile to investigate P300 double-trial and triple-trial operation in the future as it likely to produce significant time savings at reasonably high classification accuracies (>80%).

REFERENCES

Andrews, S., R. Palaniappan, A. Teoh and L.C. Kiong, 2008. Enhancing P300 component by spectral power ratio principal components for a single trial brain-computer interface. Am. J. Applied Sci., 5: 639-644. http://www.scipub.org/fulltext/ajas/ajas56639-644.pdf

Birbaumber, N., T. Hinterberger, A. Kubler and N. Neumann, 2003. The Thought-Translation Device (TTD): Neurobehavioral mechanisms and clinical outcome. IEEE Trans. Neural Syst. Rehab. Eng., 11: 120-123. DOI: 10.1109/TNSRE.2003.814439

Citi, L., R. Poli, C. Cinel and F. Supulveda, 2008. P300-based BCI mouse with genetically-optimized analogue control. IEEE Trans. Neural Syst. Rehab. Eng., 16: 51-61. DOI: 10.1109/TNSRE.2007.913184

Farwell, L.A. and E. Donchin, 1988. Taking off the top of your head: Toward a mental prosthesis utilizing event-related brain potentials. Electroencephalogr. Clin. Neurophysiol., 70: 510-523. DOI: 10.1016/0013-4694(88)90149-6

Fogel, D.B., 2005. Evolutionary Computation: Toward a New Philosophy of Machine Intelligence. 1st Edn., Wiley-IEEE Press, USA., ISBN: 13: 978-0471669517, pp: 296.

Khosrow-Pour, M., 2009. Encyclopedia of Information Science and Technology. 2nd Edn., Information Science Reference, Hershey, USA., ISBN: 1605660264, pp: 4154.

Krusienski, D.J., G. Schalk, D.J. McFarland and J.R. Wolpaw, 2007. A µ-rhythm matched filter for continuous control of a brain-computer interface. IEEE Trans. Biomed. Eng., 54: 273-280. DOI: 10.1109/TBME.2006.886661

Lenhardt, A., M. Kaper and H.J. Ritter, 2008. An adaptive P300-based online brain-computer interface. IEEE Trans. Neural Syst. Rehab. Eng., 16: 121-130. DOI: 10.1109/TNSRE.2007.912816

Lin, Z., C. Zhang, W. Wei and X. Gao, 2007. Frequency recognition based on canonical correlation analysis for SSVEP-based BCIs. IEEE Trans. Biomed. Eng., 54: 1172-1176. DOI: 10.1109/TBME.2006.889197

Nijboer, F., E.W. Sellers, J. Mellinger, M.A. Jordan and T. Matuz *et al.*, 2008. A P300-based brain-computer interface for people with amyotrophic lateral sclerosis. Clin. Neurophysiol., 119: 1909-1916. DOI: 10.1016/j.clinph.2008.03.034

Pearson, K., 1901. On lines and planes of closest fit to systems of points in space. Philos. Mag., 2: 559-572. DOI: 10.1080/14786440109462720

Peters, B.O., G. Pfurtscheller and H. Flyvbjerg, 2001. Automatic differentiation of multichannel EEG signals. IEEE Trans. Biomed. Eng., 48: 111-116. DOI: 10.1109/10.900270

Pires, G., M. Castelo-Branco and U. Nunes, 2008. Visual P300-based BCI to steer a wheelchair: A Bayesian approach. Proceedings of the 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Aug. 20-25, IEEE Xplore Press, Vancouver, BC., pp: 658-661. DOI: 10.1109/IEMBS.2008.4649238

Rebsamen, B., E. Burdet, C. Guan, H. Zhang and C.L. Teo *et al.*, 2007. Controlling a wheelchair in a building using thought. IEEE Intell. Syst., 22: 18-24. DOI: 10.1109/MIS.2007.26

Schalk, G., 2002. BCI classification contest. Wadsworth Center. http://www.bbci.de/competition/ii/albany_desc/albany_desc_ii.pdf

Schalk, G., D.J. McFarland, T. Hinterberger, N. Birbaumber and J.R. Wolpaw, 2004. BCI2000: A general-purpose Brain-Computer Interface (BCI) system. IEEE Trans. Biomed. Eng., 51: 1034-1043. DOI: 10.1109/TBME.2004.827072

Sellers, E.W., A. Kubler and E. Donchin, 2006. Brain-computer interface research at the University of South Florida cognitive psychophysiology laboratory: The P300 speller. IEEE Trans. Neural Syst. Rehab. Eng., 14: 221-224. DOI: 10.1109/TNSRE.2006.875580

Serby, H., E. Yom-Tov and G.F. Inbar, 2005. An improved P300-based brain-computer interface. IEEE Trans. Neural Syst. Rehab. Eng., 13: 89-98. DOI: 10.1109/TNSRE.2004.841878

Trejo, L.J., R. Rosipal and B. Matthews, 2006. Brain-computer interfaces for 1-D and 2-D cursor control: Designs using volitional control of the EEG spectrum or steady-state visual evoked potentials. IEEE Trans. Neural Syst. Rehab. Eng., 14: 225-229. DOI: 10.1109/TNSRE.2006.875578

Wills, S.A. and D.J.C. MacKay, 2006. DASHER-an efficient writing system for brain-computer interfaces. IEEE Trans. Neural Syst. Rehab. Eng., 14: 244-246. DOI: 10.1109/TNSRE.2006.875573

Yuehua, Y., A.T. Khalaf, Z. Xiaoxiang and W. Xinggang, 2008. Narrow-band ultraviolet B and conventional UVB phototherapy in psoriasis: A randomized controlled trial. Am. J. Applied Sci., 5: 905-908. http://www.scipub.org/fulltext/ajas/ajas58905-908.pdf

Zhang, H., C. Guan and C. Wang, 2008. Asynchronous P300-based brain-computer interfaces: A computational approach with statistical models. IEEE Trans. Biomed. Eng., 55: 1754-1763. DOI: 10.1109/TBME.2008.919128