Evaluation of Feature Selection Methods for Improved EEG Classification

Akram AlSukker and Ahmed Al-Ani
Faculty of Engineering
University of Technology, Sydney
Sydney, Australia
alsukker@eng.uts.edu.au, ahmed@eng.uts.edu.au.

***Abstract*- this paper compares several methods for feature selection used in EEG classification. Sequential, heuristics and population-based search methods are compared according to their efficiency and computational cost. A support vector machine classifier has been used to compare accuracies. Effect of the size of feature space has been explored by changing the total number of variables between 27 and 168. Experiments have been conducted to select channels as well as to select individual features from different channels.**

I. INTRODUCTION

The analysis of EEG signals is currently playing an important role in a number of research problems. For instance, EEG has the potential to provide fully immobilized people a new way of communication through brain waves, which can be used in controlling wheelchairs, computers and any other equipment. The process of controlling a computer through brain signals is referred to as Brain Computer Interface (BCI). BCI is a challenging problem due to the complexity of extracting information from EEG signals. The reliability of building these systems depends on the accuracy and speed. Achieving more accurate and faster classification system gives a better chance to integrate BCI in wider range of applications.

Most of EEG based BCI systems depends on imagining performing specific tasks, such as left and right hand movements as implemented in [1, 2]. Another BCI system was developed to move a cursor toward a defined target on a computer monitor [3, 4], which involved imagining left, right, up, and down movements. A different application called "speller" explored in [5, 6] detects imagined word according to concentrating in a matrix of letters on screen. In this work, we will concentrate on the first type of EEG system due to availability of data, however, the work can be applied to the rest of BCI systems.

The practicality of EEG-based BCI depends upon a number of factors. Two of the most important factors that should be considered are accuracy and speed. Consequently, it is crucial to find appropriate feature extraction and classification methods to the problem at hand.

Researchers used different methods to extract features form EEG signals. Some of these methods are: Principal Component analysis (PCA) [7], wavelet transform [8], adaptive Auto Regressive Model (AAR) [9], and adaptive Gaussian representation [1]. According to [10] wavelet transform was found to provide good way to visualize and decompose EEG signals. Therefore, wavelet transform is used in this paper to extract features.

Another essential issue in BCI is choosing a proper classifier that can discriminate between available classes given the extracted features. Various classification methods used in [11] which include linear discrimination analysis, Neural Networks (NN) and Support Vector Machines (SVM). We have decided to use SVM to classify EEG trials as it has a solid foundation in statistical learning theory and was found to be a powerful method for data classification.

Processing EEG signals requires dealing with multi-dimensional data (56 channels in our case). As a result there is a need to minimize the dimension of the problem to achieve acceptable performance with less computational cost. For instance, in [12] 8 channels where able to achieve similar performance as 39 channels. More research needs to be conducted to find the optimal combination of channels/features that would achieve appropriate results with less computational cost. Such subset combination should include all the necessary information for the classification task. Because in most cases the optimal subset combination is not known, variable selection becomes a necessity. For example, if we would like to find the most important subset of 5 channels out of the available 56 channels, then an exhaustive search would lead to evaluating 3.8198×10^6 subsets.

$$\binom{56}{5} = \frac{56!}{(56-5)!\times 5!} \tag{1}$$

If the evaluation of each subset takes 0.2 seconds, then we will need 8.8 days to test all subsets. Therefore, many procedures that aim at reducing the search space have been developed. These methods differ in their computational cost and the optimality of the solutions they find.

In the next section, a number of search methodologies are described. Section III presents a description of data. Results are discussed in Section IV, and conclusions are given in Section V.

II. SEARCH METHODOLOGIES

Exhaustive Search for the best available solution is only viable when dealing with small number of variables, which is

not the case for many real-life problems. As a result, several search methodologies have been developed to overcome this need. Theses methods aim at finding solutions that are close to the global one, by partially searching the problem space. Below are brief descriptions of some of the famous search methods.

A. Sequential Search (SS)

Sequential forward search starts with choosing the feature that gives the highest classification accuracy. Other features are then added to improve the accuracy one at a time. When a feature is added it can not be removed (this is known as the nesting effect). Another refined approach is introduced to partially overcome the nesting effect and is based on applying a specific number of additions/removals, where a feature that is not working well with other selected features can be removed. This step is repeated until the required number of features is achieved. This method is known as *plus l take away r* method ($l > r$). Search might start by adding features to empty subset, which is known as the Sequential Forward Search (SFS) or by removing features from the original set, which is known as the Sequential Backward Search (SBS).

B. Genetic Algorithms (GA)

The Genetic Algorithm (GA) is a population based search methodology inspired by evolution. It initially generates a random population, where each member of this population represents a solution. The best members in the current population (parents) are combined through crossover to generate another population (children). There is also a small chance of obtaining a new member through the mutation operation, by introducing small modification to one of the parents.

A binary version of GA can be used to implement feature selection. A gene (possible solution) is expressed as a binary code where presence of a feature expressed by '1' and absence of it expressed by '0'. Crossover is executed by swapping two parents' strings at a specific point. On the other hand, mutation is done by randomly complementing a small number of bits. A fitness function is needed to evaluate the solutions of the current population. The classification accuracy obtained by a certain classifier can be used for this purpose.

The GA can be implemented as follows:

- Initialize a random population of N members as a current population.
- Evaluate fitness function (accuracy) of each member and rank them according to their performance.
- Copy a subset of best members to the next population (elite children).
- Randomly Crossover two members at a time according to their fitness function to produce a number of members for next population.
- Randomly mutate a number of members in current population one at a time to produce a number of members for next population.
- Evaluate the fitness of the new population.
- Repeat creating new populations until a stopping criterion is met.

C. Simulated Annealing (SA)

Simulated Annealing (SA) is a search methodology inspired by annealing in metals. Annealing in metals is the process of heating a metal to liquid structure, then gradually cooling it to a certain temperature. This process would gradually produce a different structure with reduced internal energy. However, at certain instances of the cooling process, the internal energy may increase. Accepting the structure with such energy will depend upon the following probability equation (adapted from [13]).

$$P(A) = \exp(\frac{E_i - E_j}{k_B T}) \quad (2)$$

Where:

E_i = Previous Energy.
E_j = Current Energy.
T = Temperature.
k_B = Constant.

A SA search implementation of feature selection starts with a random binary string. A neighborhood solution can then be expressed by adding/dropping one feature at a time. For simplicity only a small number of neighbors are evaluated to reduce the computational cost. The energy, initial temperature and cooling process are calculated as follows:

1) Energy: The energy of a solution is represented by its accuracy.

2) Initial Temperature: The initial temperature is represented by a high number which would gradually be reduced. A simple way to estimate the temperature is presented in [13], which measures the maximum difference between possible achieved accuracies. The minimum accuracy can be achieved is 0% (classifier failure) and the maximum accuracy is 100%, therefore the maximum possible difference in accuracies is 100 (Δ Accuracy $_{max}$ = 100% - 0%).

3) Cooling process: Temperature can be reduced by multiplying current temperature (Tc) with a factor (α), which is usually less than one (0.8-0.99), every specific number of iteration as shown in (3).

$$T_{new} = \alpha T_c \quad (3)$$

The SA algorithm can be implemented as follows:

- Initialize current solution and temperature.
- Assign value for the cooling factor α.
- Calculate current Accuracy.
- Find a small list of neighborhood solution and find the best accuracy achieved.
- If the new accuracy is greater than previous one, replace it with current solution.
- If the new accuracy is less than previous accuracy accept it with probability of $P(\Delta\text{Accuracy}/T)$.
- Update Temperature.
- Repeat until a stopping criterion is met.

D. Tabu Search (TS)

A search methodology that uses a short term memory to forbid certain moves in search space. A list of previous visited solutions is stored as a Tabu List (TL). The search starts with a random solution then a neighborhood solution is explored and stored in the list, which would replace the current solution, noting that the move should not be done if it has been previously explored. Tabu list can store all visited solutions but its length is set to specific number to reduce the computational cost. On the other hand, this approach might prohibit good moves to be taken; therefore, aspiration criterion can be used to force certain good moves. [13] presents a simple aspiration criterion, which is implemented by allowing the move when its objective value is better than the previous one.

The TS algorithm can be implemented as follows:

- Randomly initialize Current solution.
- Calculate Current Performance
- Select a neighborhood solution and find its performance.
- If the new accuracy is better than previous one and the move does not exist in TL, then replace current solution with the previous one.
- Check aspiration criterion, if exist.
- Record the move in TL (delete oldest move if required).
- Repeat until a stopping criterion is met.

III. Description of Data

Data was taken from the Department of Medical Informatics, University of Technology, Graz, Austria[1]. EEG signals were recorded for three right handed females with 56 Ag/AgCl Electrodes, with reference electrode on the right ear. The Subject were placed in an armchair and asked to imagine right and left finger movements according to stimuli on screen. A total of 8 seconds of data were recorded at 128 Hz sampling rate, 2 seconds before the stimulus and 6 after it. A total of 406 trials were used, 208 for the left movement and 198 for right movement. More details on experiment set-up can be found in [14].

IV. Experimental Results

The search methods presented in Section II are applied to find the best channel/feature subsets. Firstly, channels are selected from the original 56 channels. Secondly, channel selection is performed by considering only the 27 channels around the motor area to observe the changes due to smaller search space. Idividual features are then selected from all channels and the motor area channels. In all the experiments described in this section, the accuracy was calculated using the average classification accuracy of a seven-fold cross-validation. Trials from all three subjects were combined to provide subject independent experiment.

[1] The authors would like to thank the Department of Medical Informatics, University of Technology, Graz, Austria for providing the data.

A. Feature Extraction

For each trail, a discrete wavelet transform, whose tree is shown in Fig. 1, has been used to extract features from the EEG signal of each channel. Three features have been used that represent energy of the frequency bands 4-8, 8-16 and 16-24 Hz. It has been found that these three features represent a good compromise between computational cost and classification accuracy.

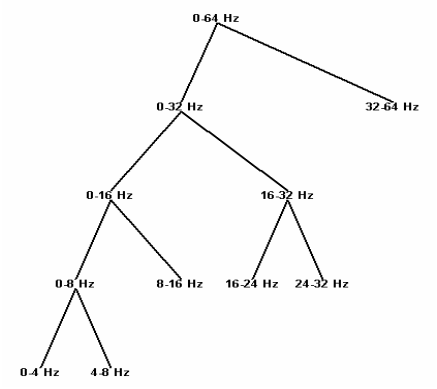

Figure 1. Wavelet tree decomposition

B. Evaluation of Search Procedure Methods

1) Sequential Search:

Sequential search has been implemented using *plus l take away r* search (*l=3, r=2*) to select channels as well as to select individual features from different channels. Both Sequential Forward Search (SFS) and Sequential Backward Search (SBS) have been tested using all channels and motor area channels, as shown in Figs. 2 and 3 respectively. The two figures indicate that the SBS achieves better results than the SFS and that selecting individual features from different channels would generally give better results than selecting channels. The figures also show that the performance starts to degrade when selecting large number of channels (or features).

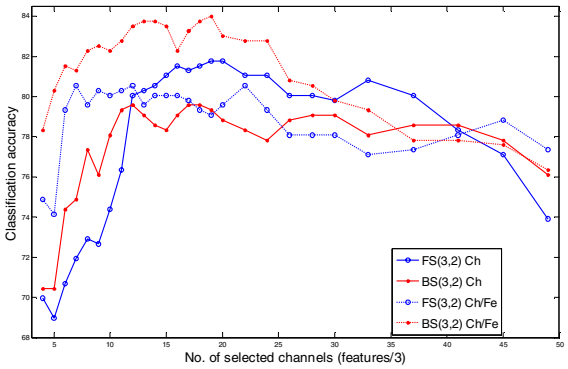

Figure 2. Sequential search classification accuracy of selected Channels/Features using all channels.

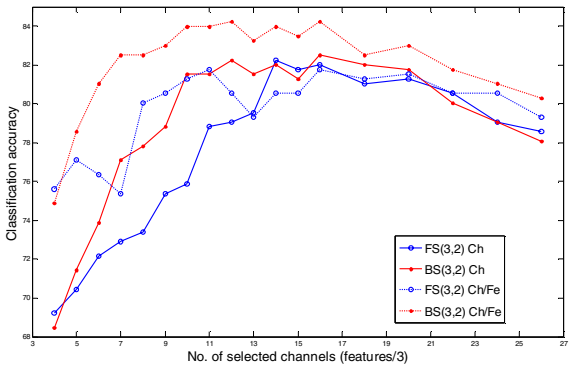

Figure 3. Sequential search classification accuracy of selected Channels/Features in motor channels only.

For the all channel experiment, the maximum obtained accuracy was found to be 83.99% obtained using 57 features (the equivalent of 19 channels each represented with three features), as shown in Fig. 2. On the other hand, the search in motor channels only gave a maximum accuracy of 84.24 % using 36 features (the equivalent of 12 channels), as shown in Fig. 3. Results obtained from the two figures indicate that motor channels provide important information for the classification of motor imagery data, but in certain instances, better results was achieved by considering other channels. In general, the analysis of motor channels gave better results with reduced computational cost.

2) Genetic Algorithms:

GA is used to select channels that maximize the classification accuracy. Two experiments have been conducted; one using all of the 56 channels, while in the other one we used the motor channels only. Searching for the best channel combination as well as the best individual features from different channels is explored.

The GA based search is performed using the following parameters: population size = 30, number of generation = 50, probability of crossover = 0.8, probability of mutation = 0.05 and number of elite children =2.

The results of all channels and motor area channels are shown in Figs. 4 and 5 respectively. In both cases, the selection of individual features proved to be better than selecting channels. The highest classification accuracy when considering all channels was 83.25% (achieved using 16 channels). On the other hand, searching the motor channels gave better result as shown in Fig. 5 where the highest accuracy (83.99%) was achieved using 51 features (the equivalent of 17 channels with three features each). It is worth mentioning that in certain instances searching the channels gave slightly better results than searching the individual features. This is due to the increased search complexity, which made it hard for the GA to find the global minimum.

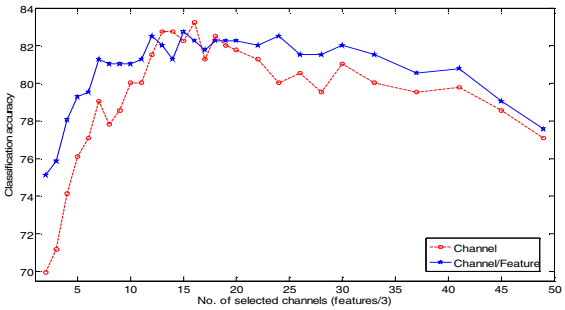

Figure 4. GA classification accuracy of selected Channels/Features using all channels.

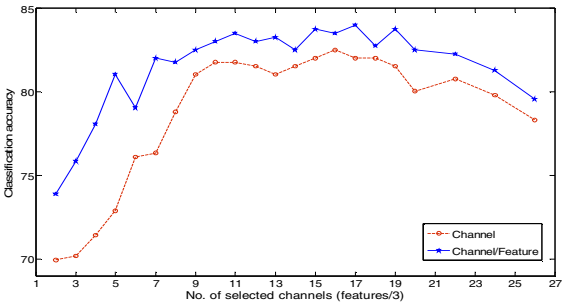

Figure 5. GA classification accuracy of selected Channels/Features in motor channels only.

3) Simulated Annealing:

Simulated annealing algorithm is implemented using the following parameters: initial temperature = 50, cooling rate = 0.95, and the final temperature = 0.023. The value of final temperature is calculated to provide the same number of iterations as genetic algorithm to provide a consistence comparison between these two methods. The highest classification accuracies that SA was able to produce were 83.25% and 85.22% using all channels and motor channels respectively (see Figs. 6 and 7). Similar to sequential search and GA, the obtained results indicate that selecting individual features outperforms channels selection. Also, there is a high degree of fluctuation shown in Fig. 6, which is caused by the big search space.

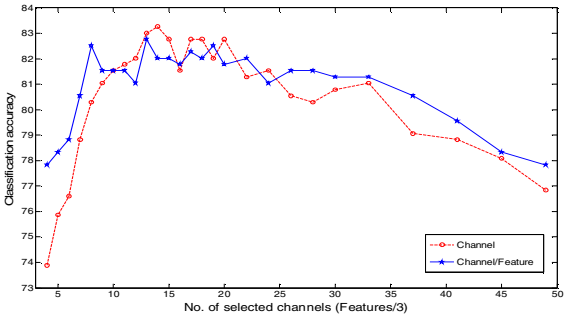

Figure 6. SA classification accuracy of selected Channels/Features using all channels.

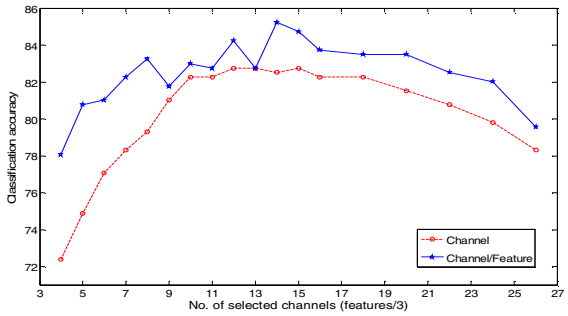

Figure 7. SA classification accuracy of selected Channels/Features in motor channels only.

3) Tabu Search:

As explained in section IId, if the search space is big, only a moderate size Tabu list would be used and the number of neighborhood solutions would be limited. Accordingly, we set the Tabu list size to 10, and used one neighborhood solution. In addition, we set the number of iterations to 2500 to provide a consistent comparison with other methods.

Figs. 8 and 9 show the classification accuracy of subsets selected from all channels and motor area channels respectively. Due to the increased complexity of the search when considering all channels, the selection of individual features could not produce good results. Better results were obtained by the individual feature selection for the case of motor channels due to the reduced search space.

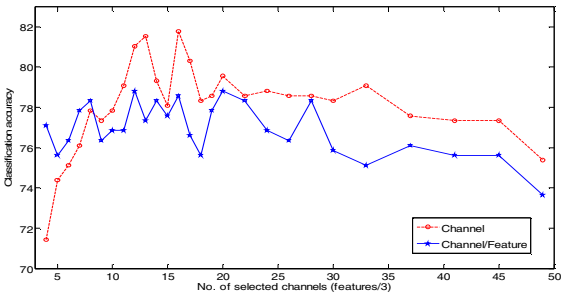

Figure 8. TS classification accuracy of selected Channels/Features using all channels.

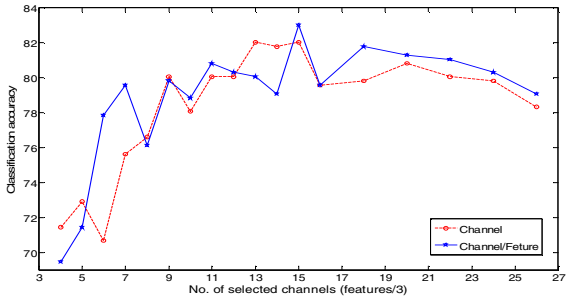

Figure 9. TS classification accuracy of selected Channels/Features in motor channels only.

C. Time Versus Accuracy Evaluation

A high classification accuracy is not always good enough, expect if it can be achieved in acceptable time. Both the classification accuracy and computational time for the methods described in section II are calculated. Figs. 10 and 11 provide a mean of comparison between the classification accuracy of these methods. The SBS proved to be slightly better than other methods when considering all channels. However, its performance degraded noticeably when the desired number of selected channels/features increased. The GA gave slightly better results when selecting higher number of features/channels (see Fig. 10). On ther other hand, the performance of SA, SBS and GA gave comparison results for the case of motor channels, with slightly better results achieved by SA. However, speed should be considered as well to evaluate these findings.

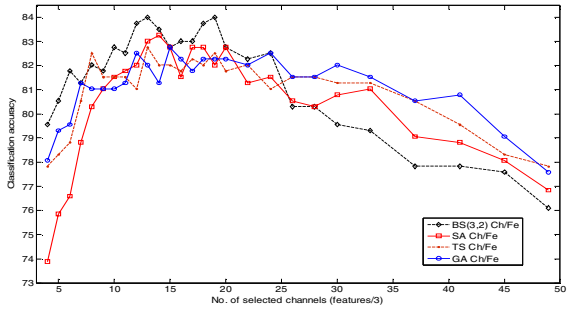

Figure 10. Comparison between different methods in channel motor area.

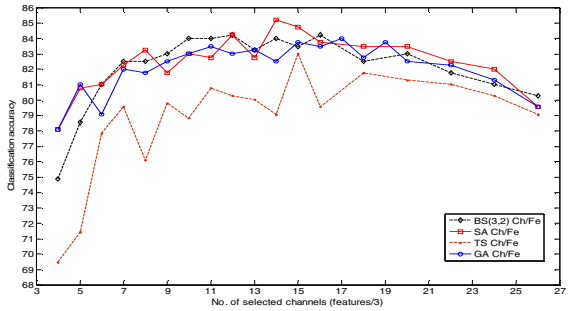

Figure 11. Comparison between different methods in all channels.

A number of selected channels are evaluated according to their speed[2] as shown in Tables I and II. The desired number of channels was set to 10, 28, and 46 for the case of all 56 channels experiment, while for the motor area experiment; we used 5, 13, and 22 as the desired number of selected channels. Instead of recording the highest classification accuracy and the time required to achieve it, we used a formula that finds a compromise between classification accuracy and computational time. For instance, if the second best classification accuracy is lower than the best one by 0.25%, but was achieved before 300 seconds or less, then the second best

[2] All experiments have been implemented in MATLAB using Intel Centrino 1.7GHz laptop.

accuracy is considered to represent a better compromise between time and accuracy.

Both tables indicate that in almost all cases SA shows superiority over other methods. It gave better results in less time. Although Tabu search is considered a fast search but it lacks the accuracy. On the other hand, SBS gave a relatively high result but in more time (due to the *plus l take away r* concept).

TABLE I
COMPROMISED CLASSIFICATION ACCURACY VS TIME FOR CHANNEL SEARCH

No of Channels	FS(3,2) Acc (%) time (sec)	BS(3,2) Acc (%) time (sec)	GA Acc (%) time (sec)	SA Acc (%) time (sec)	TS Acc (%) time (sec)
All					
10	74.38% 630	78.08% 7254	79.31% 9278	**80.30% 811**	78.82% 51
28	80.04% 3969	79.06% 5778	80.54% 2231	**80.54% 206**	78.57% 154
46	76.84% 7575	77.34% 2713	76.85% 527	**78.57% 528 Sec**	75.86% 173
Motor					
5	70.44% 81	71.43% 1698	74.63% 768	**74.88% 137**	73.40% 17
13	79.56% 580	81.53% 1413	82.02% 579	**82.51% 366**	80.54% 71
22	80.54% 1317	80.05% 667	80.79% 259	**81.28% 402**	80.30% 22

TABLE II
COMPROMISED CLASSIFICATION ACCURACY VS TIME FOR FEATURE SEARCH

No of Features/3	FS(3,2) Acc (%) time (sec)	BS(3,2) Acc (%) time (sec)	GA Acc (%) time (sec)	SA Acc (%) time (sec)	TS Acc (%) time (sec)
All					
10	80.05% 5899	82.76% 64338	81.53% 642	**82.02% 1731**	81.03% 554
28	78.08% 37320	80.54% 51160	**81.53% 993**	**81.28% 637**	79.56% 744
46	77.59% 81943	77.09% 23064	**78.08% 991**	**78.33% 2346**	77.59% 2210
Motor					
5	77.09% 579	78.57% 12530	77.09% 395	**80.79% 667**	78.08% 170
13	79.31% 4633	83.25% 10029	82.51% 559	**83.00% 1549**	**82.76% 512**
22	80.54% 11106	81.77% 4775	82.27% 1178	**83.01% 1627**	81.28% 1049

V. CONCLUSIONS

Different search methodologies have been explored for evaluation purpose. The classification accuracy obtained by these methods for selecting channels as well as individual features from different channels, when considering two different sizes of the search space, were calculated. Firstly, the methods were compared with respect to their classification accuracy, and the results show that in most cases the selection of individual features provides better accuracy and that the sequential backward selection, genetic algorithm and simulated annealing were noticeably better than Tabu search. When considering both the classification accuracy and computational time aspects, results indicate that simulated annealing was able to achieve better compromise than other methods.

REFERENCES

[1] E. J. X. Costa and E. F. CabralJr, "EEG-based discrimination between imagination of left and right hand movements using adaptive gaussian representation," *Medical Engineering & Physics*, vol. 22, pp. 345-348, 2000.

[2] L. Boqiang, W. Mingshi, Y. Hongqiang, Y. Lanlan, and L. Zhongguo, "Study of Feature Classification Methods in BCI Based on Neural Networks," 2005.

[3] D. J. McFarland, W. A. Sarnacki, T. M. Vaughan, and J. R. Wolpaw, "Brain-computer interface (BCI) operation: signal and noise during early training sessions," *Clinical Neurophysiology*, vol. 116, pp. 56-62, 2005.

[4] F. Piccione, F. Giorgi, P. Tonin, K. Priftis, S. Giove, S. Silvoni, G. Palmas, and F. Beverina, "P300-based brain computer interface: reliability and performance in healthy and paralysed participants," *Clinical Neurophysiology: Official Journal Of The International Federation Of Clinical Neurophysiology*, vol. 117, pp. 531-537, 2006.

[5] G. Cuntai, M. Thulasidas, and W. Jiankang, "High performance P300 speller for brain-computer interface," 2004.

[6] M. Thulasidas, C. Guan, and J. Wu, "Robust Classification of EEG Signal for Brain–Computer Interface," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on [see also IEEE Trans. on Rehabilitation Engineering]*, vol. 14, pp. 24-29, 2006.

[7] L. Yong, G. Xiaorong, L. Hesheng, and G. Shangkai, "Classification of single-trial electroencephalogram during finger movement," *Biomedical Engineering, IEEE Transactions on*, vol. 51, pp. 1019-1025, 2004.

[8] 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," *Biomedical Engineering, IEEE Transactions on*, vol. 51, pp. 1057-1061, 2004.

[9] G. S. Dharwarkar and O. Basir, "Enhancing Temporal Classification of AAR Parameters in EEG single-trial analysis for Brain-Computer Interfacing," 2005.

[10] V. J. Samar, "Wavelet Analysis of Neuroelectric Waveforms," *Brain and Language*, vol. 66, pp. 1-6, 1999.

[11] D. Garrett, D. A. Peterson, C. W. Anderson, and M. H. Thaut, "Comparison of linear, nonlinear, and feature selection methods for EEG signal classification," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on [see also IEEE Trans. on Rehabilitation Engineering]*, vol. 11, pp. 141-144, 2003.

[12] T. N. Lal, M. Schroder, T. Hinterberger, J. Weston, M. Bogdan, N. Birbaumer, and B. Scholkopf, "Support vector channel selection in BCI," *Biomedical Engineering, IEEE Transactions on*, vol. 51, pp. 1003-1010, 2004.

[13] E. Burke and G. Kendall, *Search methodologies : introductory tutorials in optimization and decision support techniques*. New York: Springer, 2005.

[14] H. Ramoser, J. Muller-Gerking, and G. Pfurtscheller, "Optimal spatial filtering of single trial EEG during imagined hand movement," *Rehabilitation Engineering, IEEE Transactions on [see also IEEE Trans. on Neural Systems and Rehabilitation]*, vol. 8, pp. 441-446, 2000.