Original Article

www.jmss.mui.ac.ir

An Efficient P300-based BCI Using Wavelet Features and IBPSO-based Channel Selection

Bahram Perseh, Ahmad R. Sharafat

Department of Electrical and Computer Engineering, Tarbiat Modares University, Tehran, Iran

Submission: 30-06-2012 Accepted: 01-07-2012

ABSTRACT

We present a novel and efficient scheme that selects a minimal set of effective features and channels for detecting the P300 component of the event-related potential in the brain–computer interface (BCI) paradigm. For obtaining a minimal set of effective features, we take the truncated coefficients of discrete Daubechies 4 wavelet, and for selecting the effective electroencephalogram channels, we utilize an improved binary particle swarm optimization algorithm together with the Bhattacharyya criterion. We tested our proposed scheme on dataset IIb of BCI competition 2005 and achieved 97.5% and 74.5% accuracy in 15 and 5 trials, respectively, using a simple classification algorithm based on Bayesian linear discriminant analysis. We also tested our proposed scheme on Hoffmann's dataset for eight subjects, and achieved similar results.

Key words: *Bayesian linear discriminant analysis, Bhattacharyya distance, brain–computer interface, discrete wavelet, event-related potentials, improved binary particle swarm optimization algorithm*

INTRODUCTION

Brain–computer interface (BCI) provides a direct communication channel between a subject's brain and a computer by using electroencephalogram (EEG) signals.[1] It improves the quality of life for some patients that suffer from a neurological disorder called locked-in syndrome, e.g., the amyotrophic lateral sclerosis (ALS). Some existing implementations of BCI are mainly based on utilizing the P300 wave, which was shown for the first time in[2] to be an event-related potential and was later utilized in[3] as a control signal in BCI systems. The P300 wave is a positive deflection in the EEG around 300 ms after visual or auditory stimuli for normal young adults.

The visual P300-BCI is a synchronous device that enables subjects to spell words or demand an object by focusing their attention on symbols or images in a matrix displayed on a computer screen. In this BCI protocol, the sequence of symbols or images is flashed in a random order and the subject tries to discriminate a desired symbol or image (target) during a random sequence of target and non-target stimuli (oddball paradigm).[1,3] In the oddball paradigm, the subject focus is on detecting target events, and ignores the non-target events. Target events, on the average, produce larger P300 potentials than non-target events.[4] Thus, by detecting the P300-ERP pertaining to a target image, the subject's intention can be recognized, and a sequence of such detections can lead to, for instance, spelling a word that was intended by the subject. Extracting P300-ERPs from background EEG and environmental noise is the main challenge in ERP analysis. The ERP has low signal-to-noise ratio (SNR) and is a transient signal, making ERPs difficult to detect. In spite of this, it is very desirable to correctly detect ERPs by efficiently utilizing a minimal number of EEG channels to reduce calculations.

Typically, a P300-based BCI system has four components, namely preprocessing, feature extraction, channel selection, and classification. Although improving any one of these parts can improve the performance of the system as a whole, in this paper, we focus on feature extraction and channel selection. In many existing P300-BCI systems, due to the large number of electrodes and long durations of recorded EEG signals, one has to deal with extensive data streams that produce a large number of features, which in turn would cause over-fitting in the classifier. Using a minimal set of effective features and channels prevents the over-fitting problem and reduces calculations. As for feature extraction, in existing schemes, either a set of effective features is extracted for a given channel set as in[5-7], or a set of effective channels is selected for given feature set as in.[8-11] However, optimal choices for features and channels

Address for correspondence:
Prof. Ahmad R. Sharafat, Department of Electrical and Computer Engineering, Tarbiat Modares University, Tehran, Iran,
E-mail: sharafat@modares.ac.ir

are subject-dependent, and may depend on the BCI protocol as well. In this regard, we propose a scheme for joint selection of features and channels for each subject.

Feature Extraction

The discriminating features in P300-ERPs may be time-dependent, frequency-dependent, or time–frequency-dependent. In[8,11], pre-processed signal samples, and in[12], frequency-domain features (Fourier transforms of segmented ERPs) are fed to the classification algorithm. However, since the ERP is a transient signal, time–frequency features are more appropriate. Time–frequency features can be obtained by the wavelet transform, which is an efficient tool for multi-resolution analysis of non-stationary and transient signals. In[13], the continuous wavelet transform (CWT) is used for extracting time-frequency features of the EEG, and the T-student algorithm is applied for choosing those features that are more effective and discriminant, resulting in significant improvements. One obvious drawback of the CWT is that it requires excessive calculations.

The discrete wavelet transform (DWT) is used as a powerful denoising and feature extraction tool to detect the P300-ERPs from EEG epochs. In[14,15] a Daubechies 4 wavelet is used for removing noise and unwanted frequency components from the EEG in adults and young people. In[9], the DWT is applied to the dataset IIb of BCI competition 2005. Although the results are relatively accurate, the number of channels and features are excessive. In[16], the discriminating features are the coefficients of the DWT of the signal, and a weighted feature vector is used for further improvements. It was noted that the effective features are in 1–8 Hz frequency band.

In this paper, we take the coefficients in the effective sub-bands of the DWT of EEG signals as their discriminating features, where effective sub-bands are identified via the five-fold cross-validation procedure. The mother wavelet is Daubechies 4 (db4), which is suitable for detecting changes in EEG signals.[17] The beginning part of the impulse response of the decomposition low pass filter and the end part of the impulse response of the decomposition high pass filter for the db4 are near zero in the MATLAB wavelet toolbox. We force such small values to zero by truncating the corresponding DWT coefficients, which causes 12% to 30% reduction in the number of features, yet produces satisfactory results.

Channel Selection

In[12], all EEG electrodes (64 channels) are used for signal classification. Although it involves a significant amount of calculations, the accuracy of BCI results is not very satisfactory. To address such shortcomings, various methods have been proposed in the literature to identify the more effective channels. In[8,9], the training data is divided into several partitions (17 partitions in[8] and 10 partitions in[9]), and for each partition, effective channels are obtained by recursively eliminating the lesser effective channels. Then the classifier algorithm is applied on each partition, and voting is used on the outputs of classifiers to detect P300-ERPs. Although partitioning of the training data and using a separate classifier for each partition reduces calculations, but as we will show later, further improvements are possible.

Another approach is to use the Fisher criterion score (FCS)[18,19] to identify the effective channels, which may result in not selecting a number of highly correlated channels. A channel is effective for signal classification if the sum of FCSs for all features in that channel has a high value. In contrast, the Bhattacharyya criterion is simpler, and is calculated directly from the feature vector of each channel individually. The main drawback of these methods is that correlated channels that may produce better results may not be selected because of their low FCSs. In[20], a binary version of PSO algorithm is used for channel selection among all EEG channels that may include correlated channels. Although they showed that their method outperforms sequential floating forward search algorithm, but selecting from all channels (without first eliminating the lesser effective ones) increases calculations with no apparent benefit.

We present a two stage approach for identifying a minimal subset of effective channels. We begin by sorting channels using the Bhattacharyya distance in decreasing order and eliminate 50% of channels that have smaller distances. We then identify the more effective channels in the remaining channels using the improved binary particle swarm optimization (IBPSO) algorithm. In this way, we limit the search space and processing time of the IBPSO algorithm.

The rest of this paper is organized as follows. The two P300-BCI datasets that we use are described in Section 2. In Section 3, we present our proposed scheme that includes preprocessing, feature extraction and minimal feature selection, classification, and the two-step channel selection. Section 4 contains experimental results. Discussion and conclusions are given in Sections 5 and 6, respectively.

P300-BCI Datasets

In order to benchmark our proposed scheme, we use two different P300-BCI datasets, namely the dataset IIb from the third edition of BCI competition 2005 for two subjects,[21] and data recorded in a P300 environment control paradigm by Hoffmann *et al.*[11] for eight subjects. The protocol of each dataset is briefly explained below.

Dataset 1

The P300 speller paradigm[3] of BCI competition 2005 displays a 6×6 matrix of characters [Figure 1a] to each subject. Each row and each column in the display are flashed at random,

and the subject's task is to focus on characters in a given word, one character at a time. Two out of 12 illuminated rows or columns contain the desired letter (in one row and in one column). Thus, one P300-ERP is produced when the row/column of the expected letter is illuminated.[21]

This dataset was recorded for two different subjects A and B. For each subject, 64 channels are sampled at the rate of 240 samples per second for 15 trials per character. Figure 1b shows the position of EEG electrodes. The recorded EEG is band-pass filtered from 0.1 to 60 Hz. As the 60 Hz cut-off is way above the highest frequency components of P300, we will low pass filter the dataset signals to further reduce their additive noise. The training and the testing datasets consist of 85 and 100 characters, respectively. As such, the number of corresponding epochs for each subject are $85\times12\times15=15300$ and $100\times12\times15=18000$, respectively.

Dataset 2

In this dataset, as shown in Figure 2a, six images include a television, a telephone, a lamp, a door, a window, and a radio are shown on a laptop screen to eight subjects (four disabled and four healthy subjects).[11] The disabled subjects were all wheelchair-bound but had varying communication and limb muscle control abilities. The images are flashed in a random sequence, one image at a time, one image being the target one, and the rest are non-targets. A block consists of six images, each flashed once. Similar to the P300 speller paradigm, when the target image is flashed, a P300-ERP is produced. For each subject, the dataset consists of four sessions, each having six runs. The numbers of blocks are randomly chosen between 20 and 25, i.e., on the average, 22.5 blocks of six flashes were displayed in one run. Hence, on the average, each subject generates 540 target trials (4 sessions × 6 runs × 1 target × 22.5 blocks = 540) and 2700 non-target trials (4 sessions × 6 runs × 5 nontargets × 22.5 blocks = 2700). The sampling rate of EEG signals is 2048 samples per second and 32 electrodes are recorded from Figure 2b.

MATERIALS AND METHODS

Figure 3a and b show the block diagrams for training and testing of our proposed scheme, respectively. For training, the preprocessing module includes filtering, artifact reduction, and data segmentation. Features are extracted by discrete wavelet transform, and truncated to remove near-zero coefficients. A five-fold cross-validation procedure[22] is utilized to select the best sub-bands by using BLDA classifier on the first eight channels selected by the Bhattacharyya criterion. As in[8], the extracted features are normalized to zero mean and unit variance. To select the best channels, we disregard 50% of channels whose Bhattacharyya distances are smaller than those of the remaining channels (32 channels

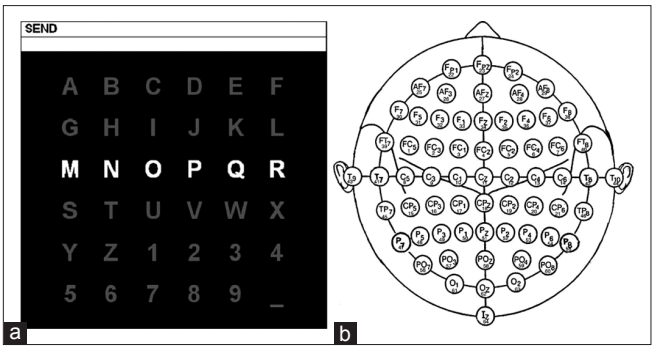

Figure 1: (a) The matrix used in the P300 speller paradigm (b) the position of electrodes

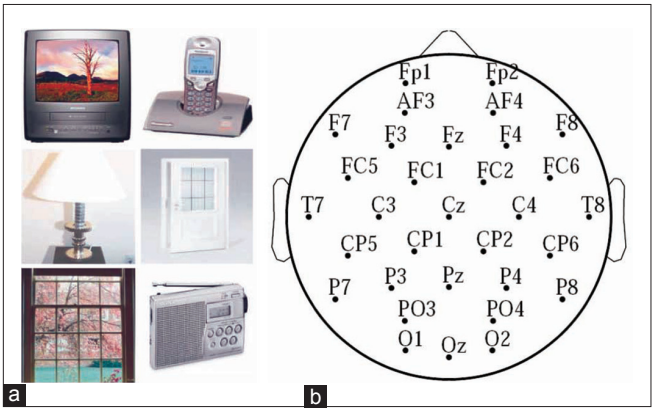

Figure 2: (a) Six images used in[11], (b) the position of electrodes

for Dataset 1 and 16 channels for Dataset 2), and apply the remaining channels together with their selected sub-bands to the IBPSO module. In the sequel, the main modules in each block diagram in Figure 3a and b are described.

Preprocessing

In general, ERP epochs are heavily contaminated by noise, and are difficult to detect in few trials. As in[5,6], signals from each channel are band-pass filtered (0.1–30.0 Hz) using a 6^{th} order forward–backward Butterworth filter. The bandwidth of 0.1–30.0 Hz covers the frequency range of important EEG rhythms (delta (0.5–4.0 Hz), theta (4.0–7.5 Hz), alpha (8.0–13.0 Hz), and beta (14.0–26.0 Hz)). The Windsorizing method described in[11] is used to reduce the effects of large amplitude outliers caused by eye movements, blinking, or subject's movements. In doing so, signal amplitudes above the 90^{th} and below the 10^{th} percentiles are clipped. After each flash, we use the first 700 ms of recorded signals in both datasets. This window is long enough to capture all required time features for an efficient classification, although, the P300 component is expected to occur around 300 ms after the stimulus.[8]

Sorting Channels by Bhattacharyya Distance

The efficiency of each channel can be measured based on

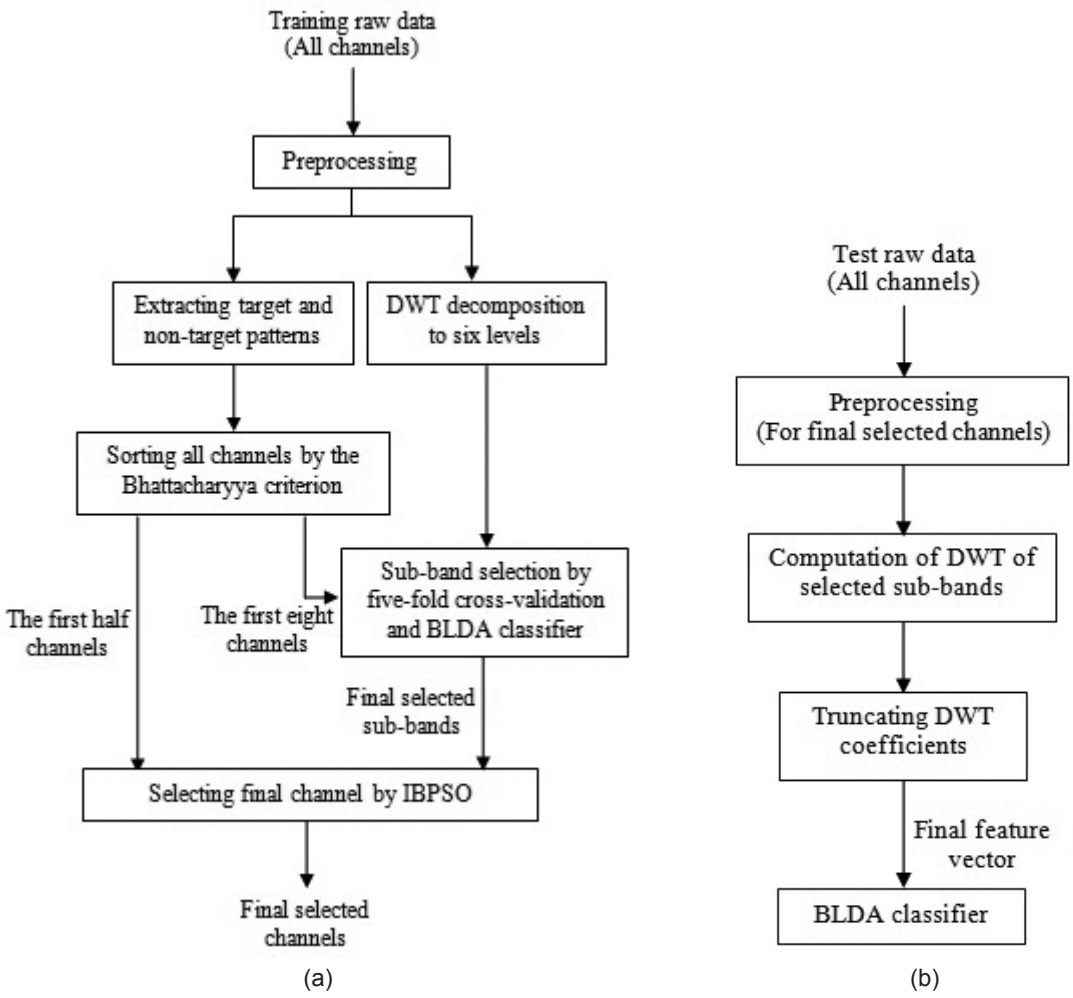

Figure 3: (a) Block diagram of the proposed scheme for training, and (b) for testing

its ability to discriminate signals pertaining to target and non-target patterns in the training dataset. To do so, we use a statistical measure, e.g., the Bhattacharyya distance (BD) that reveals the degree of difference between the two respective patterns via a real valued scalar[23,24] defined by

$$BD = \frac{1}{8}(m_1 - m_2)^T \left(\frac{C_1 + C_2}{2}\right)^{-1} (m_1 - m_2) + \frac{1}{2}\ln\left[\frac{\left|\frac{C_1 + C_2}{2}\right|}{\sqrt{|C_1| \times |C_2|}}\right] \quad (1)$$

where $|\cdot|$ denotes the determinant of a matrix, m_1 is the mean vector of target pattern signals, m_2 is the mean vector of non-target pattern signals, and C_1 and C_2 are the corresponding covariance matrices.

The value of BD provides a quantitative measure for sorting channels based on their pre-processed signal samples in the training datasets. To obtain target and non-target preprocessed signal samples, each segment of the preprocessed signal is down sampled by a factor of 4, which still satisfies the Nyquist rate for the preprocessing band pass filter. For example, Figure 4 shows the BD values for Subject A in the P300 speller dataset IIb, obtained by extracting 42 preprocessed signal samples from a single channel. We use the sorted channels for two purposes, namely, for selecting eight initial channels that will be utilized for finding the best sub-bands of wavelet coefficients, and for identifying those channels that can be used by the IBPSO algorithm.

Feature Extraction

Wavelet transform (WT) has been extensively used in ERP analysis due to its ability to effectively explore both the time-domain and the frequency-domain features of ERP.[22] It is also superior to the short time Fourier transform (STFT). This is because the STFT's window is fixed, resulting in a possible loss of some information on fast changing signals; which is in contrast to WT that estimates the low frequency information of the signal by using expanded windows and

the high frequency information by utilizing short windows. As such, WT can provide an efficient analysis of non-stationary and transient signals.

Wavelet analysis can be performed either in the continuous mode (CWT) or in the discrete mode (DWT). The DWT involves less computation, is simpler than CWT, and can be implemented via digital filtering techniques. The DWT decomposes signal $x[n]$ into different frequency sub-bands with different resolutions using the scaling function ($\phi_{j,k}[n]$) and the wavelet function ($\psi_{j,k}[n]$), where j and k are integers. These functions are the dilated and shifted version of $\phi[n]$ and $\psi[n]$, defined by

$$\phi_{j,k}[n] = 2^{\frac{-j}{2}}\phi\left[2^{-jn}-k\right] \quad (2)$$

$$\psi_{j,k}[n] = 2^{\frac{-j}{2}}\psi\left[2^{-jn}-k\right] \quad (3)$$

The DWT projects the original signal into a set of basis functions built from translations and scaling of the wavelet function (also called the mother wavelet). The DWT coefficients are obtained by convolving $x[n]$ with $\psi_{j,k}[n]$. The DWT employs a discrete-time mother wavelet whose dilation and translation parameters are integers. The contracted and dilated versions of the wavelet function will match the high-frequency and low-frequency components of the original signal, respectively. The DWT can be implemented by multi-resolution analysis (MRA) through the application of digital filter banks.[25] The procedure for MRA via dyadic filter banks for decomposing a signal $x[n]$ is schematically shown in Figure 5. Each stage consists of a high-pass filter ($h_{\text{High}}[n]$) corresponding to $\psi_{j,k}[n]$, a low-pass filter ($h_{\text{Low}}[n]$) corresponding to $\phi_{j,k}[n]$, and two down samplers. The decomposition process of $x[n]$ via dyadic filter banks is described below.

1. A mother wavelet is chosen to obtain the filters' impulse responses $h_{\text{Low}}[n]$ and $h_{\text{High}}[n]$
2. The values of $A_1[n]$ and $D_1[n]$ are obtained by convolving $x[n]$ with $h_{\text{Low}}[n]$ and $h_{\text{High}}[n]$, respectively
3. The values of $A_1[n]$ and $D_1[n]$ are divided by 2 to get the approximation coefficients $CA_1[n]$ (i.e., the low frequency part of the signal), and the detail coefficients $CD_1[n]$ (i.e., the high frequency part of the signal), respectively. This is the first level of wavelet decomposition.
4. The DWT decomposition process continues the same as in 1) above for the low-pass branch in Figure 5. The values of $CA_1[n]$ is further decomposed to $CA_2[n]$ and $CD_2[n]$ by using $h_{\text{Low}}[n]$, $h_{\text{High}}[n]$, and the two down-samplers.
5. By continuing the wavelet decomposition up to level j, the output of the dyadic wavelet transform will be the detail coefficients $CD_1,\cdots,CD_j$ and the approximation coefficients CA_j (i.e., the approximation coefficients of the last decomposition level). Each of these $j+1$ parts of the wavelet coefficients corresponds to the signal information within a specific frequency sub-band.

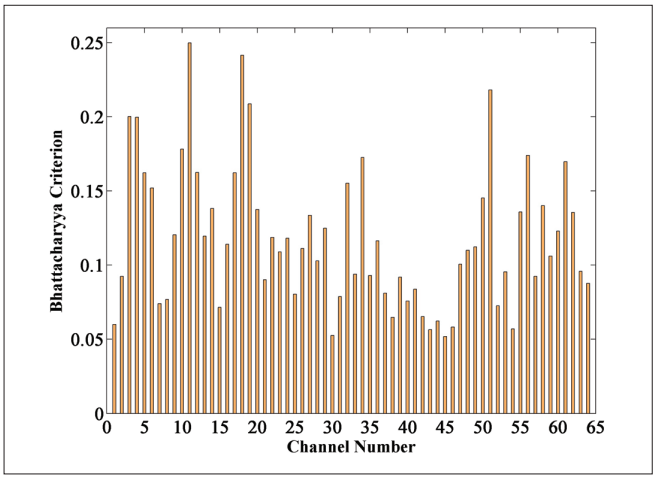

Figure 4: The values of Bhattacharyya distance for each channel for subject A in the P300 speller dataset IIb

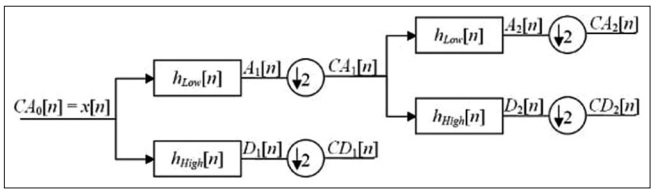

Figure 5: Decomposing of × (*n*) using filter banks

Obtaining wavelet coefficients for the j^{th} level can be summarized by

$$A_j[n] = CA_{j-1}[n] \star h_{\text{Low}}[n] \quad (4)$$

$$D_j[n] = CA_{j-1}[n] \star h_{\text{High}}[n] \quad (5)$$

$$CA_j[n] = A_j[2n] \quad (6)$$

$$CD_j[n] = D_j[2n] \quad (7)$$

Note that because of down-sampling in the dyadic structure in Figure 5, the DWT is a shift-varying transform.[26] In contrast, the stationary wavelet transform (SWT), is shift-invariant.[27] In the SWT, the scales are dyadic but time steps at each level are not. Moreover, the SWT is a non-orthogonal transform with temporal redundancies.[28] In our case, using the shift-invariant SWT that entails more calculations, does not significantly improve the classification accuracy as compared to using the DWT.

Selection of a mother wavelet and a proper decomposition level are very important in the DWT. Choosing the mother wavelet for detecting P300-ERPs can be difficult because many wavelet properties cannot be jointly optimized.[29] The Daubechies family of wavelets are very smooth, orthogonal, and easy to implement. In[4,17,30], the Daubechies order-4 (db4)

wavelet has been employed for decomposing EEG signals. We also choose the db4 mother wavelet, as it resembles the P300 component in ERPs.[17]

Effective frequency components in ERPs specify the number of decomposition levels, which are chosen such that those segments of the signal that are highly correlated with the frequencies required for classification of the signal are retained in the wavelet coefficients[31] To have a sufficient number of low-frequency components, we decompose the signal into six levels. Since the bandwidth of the signal is limited to 0.1–30 Hz, we focus on those subbands and their corresponding coefficients that pertain to 0.1–30 Hz. For selecting the best DWT sub-bands for each subject, we compute all DWT coefficients within 0–30 Hz for the first eight channels selected by the Bhattacharyya criterion in the training dataset. We then truncate the DWT coefficients as explained in Section III-D, and obtain all possible combinations of the truncated DWT coefficients of those sub-bands that do not overlap in frequency.

For performance evaluation, the training set is randomly partitioned into five subsets using the five-fold cross-validation procedure,[22] where a single subset is reserved for validation and the remaining four are used for training. The cross-validation process is then repeated five times, when each of the five subsets are used exactly once as the validation data. The results are averaged to obtain a single estimation. The performance of each validation set is determined by the channel classification score denoted by C_{cs} in (8) below, taken from,[8] where f_p, t_p and f_n are the numbers of false positives, true positives and false negatives, respectively.

$$C_{cs} = \frac{t_p}{t_p + f_p + f_n}. \quad (8)$$

The reason for using this criterion is that C_{cs} does not include the number of true negatives, which is important for unbalanced datasets. This causes the feature selection to focus on those feature vectors that give positive scores to true positives and false positives, which are fewer in number than true negatives and false negatives. For feature selection, classifier performances are evaluated on target and non-target features (binary classification) and not on character or image recognition performances.

Minimal Feature Selection

By using suitable feature extraction and selection processes, the computation cost decreases and classification performance improves. In general, not all extracted features are useful for classification, as some features are irrelevant or redundant and reduce classification accuracy. We now show that using all wavelet coefficients in each level results in an expanded feature set and may reduce the classification accuracy.

Figure 6 shows the impulse response of the decomposition low-pass and high-pass filters corresponding to db4 mother wavelet in which the first 3 coefficients of $h_{Low}[n]$ and the last 3 coefficients of $h_{High}[n]$ are near zero. We use this property of db4 decomposition filters to reduce the number of features. The values of $A_1[n]$ and $D_1[n]$ in Figure 5 are obtained by convolving $x[n]$ with $h_{Low}[n]$, and $x[n]$ with $h_{High}[n]$, respectively. Hence, the first 3 values of $A_1[n]$ and the last three values of $D_1[n]$ are near zero. The down-sampled values of $A_1[n]$ and $D_1[n]$ provide $CA_1[n]$ and $CD_1[n]$ coefficients, respectively. Thus, the first two values of $CA_1[n]$ and at least the last value of $CD_1[n]$ are near zero. Since $x[n]$ is unknown, we have no information on the number of first near zero values of $CD_1[n]$.

Since the first two values of $CA_1[n]$ and the first three values of $h_{Low}[n]$ are near zero, and $A_2[n] = CA_1[n] \star h_{Low}[n]$, the first five values of $A_2[n]$ are near zero, and so the first three values of $CA_2[n]$ (which is the down sampled $A_2[n]$) are near zero. Besides, since the last three values of $h_{High}[n]$ are near zero and $D_2[n] = CA_1[n] \star h_{High}[n]$, the first two values and the last three values of $D_2[n]$ are near zero. Thus, the first value and at least the last value of $CD_2[n]$ are zero. Similarly, the first three values of $CA_j[n]$, and the first two values and at least the last value of $CD_j[n]$ are near zero. Figure 7 shows the truncated coefficients of a segment of EEG signal for (CA_3 - CA_6) and (CD_3 - CD_6). The eliminated and remaining coefficients are identified by (∘) and (∞), respectively. Not that truncating the DWT coefficients reduces the number of features by 12 to 30%.

Classification Algorithm

Classification accuracy, simplicity, and fast training are three important factors for choosing a classifier. In the literature, different classification methods are used in the P300-BCI applications, among which are the Fisher linear discriminant analysis (FLDA),[13] the support vector machine (SVM),[8,12] and

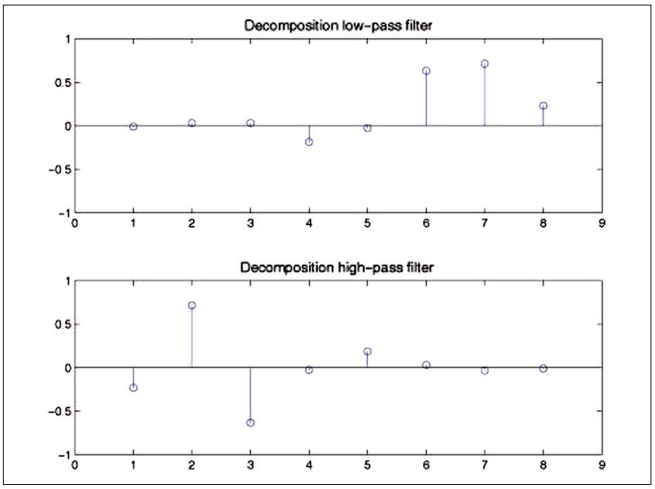

Figure 6: The values of decomposition low-pass and high-pass coefficients for the db4 mother wavelet

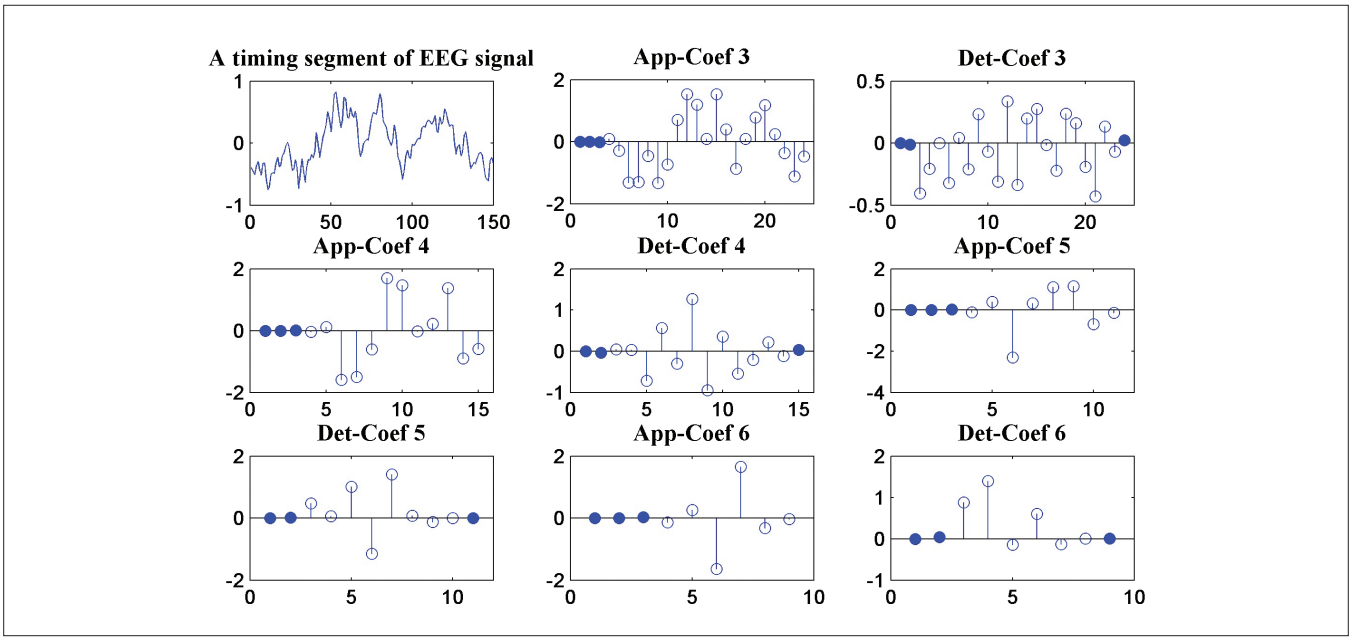

Figure 7: A segment of EEG signal and its truncated approximation and detail coefficients for different decomposition levels of the db4 mother wavelet

the Bayesian linear discriminant analysis (BLDA).[10,11] The FLDA is a simple, fast, and easy to use classifier but its performance deteriorates when many electrodes or features are used. This problem is solved by using BLDA, which uses regularization to prevent over-fitting to high-dimensional and noisy data sets. In the Bayesian analysis, the degree of regularization is estimated quickly, robustly and automatically from the training data without needing the complex cross-validation procedures for tuning its parameters.[11]

In[10,32], it is shown that the BLDA outperforms the SVM and some other classifiers for all tested cases, and its complexity is low. Hence, we use the two-class BLDA classifier (which is similar than the one described in[11]) to classify target and non-target EEG signals. Training features include a set of d-dimensional feature vectors for each class $\mathrm{x}_j = [x_{1j}, x_{2j}, \cdots, x_{dj}]$ and the corresponding class-label $y_j \in \{-1, 1\}$, where j is the feature vector number. The basic assumption in the Bayesian regression is that the feature matrix $\mathrm{X} = [\mathrm{x}_1, \mathrm{x}_2, \cdots, \mathrm{x}_{N_i}]$ and its corresponding label vector $\mathrm{y} = [y_1, y_2, \cdots, y_{N_i}]$ are linearly related, i.e.,

$$\mathrm{y} = \mathrm{w}^{\mathrm{T}}\mathrm{X} + \mathrm{n} \tag{9}$$

where $\mathrm{w} = [w_1, w_2, \cdots, w_d]^{\mathrm{T}}$ is a projection vector to be optimized, $\mathrm{n} = [n_1, n_2, \cdots, n_{N_i}]$ is an additive white Gaussian noise vector, and N_i is the number of feature vectors in the i^{th} class. The likelihood function for w in the regression is

$$p(\mathrm{X}, \mathrm{y} \mid \beta, \mathrm{w}) = \left(\frac{\beta}{2\pi}\right)^{\frac{l}{2}} e^{-\frac{\beta}{2}|\,|\mathrm{w}^T\mathrm{x} - \mathrm{y}|\,|^2} \tag{10}$$

where β is the inverse variance of noise, and l is the number of cases in the training set. For the Bayesian setting, the prior distribution of weight vector w is assumed to be Gaussian, defined by

$$p(\mathrm{w} \mid \alpha) = \prod_{i=1}^{d}\left(\frac{\alpha_i}{2\pi}\right)^{\frac{l}{2}} e^{-\frac{l}{2}\left(\mathrm{w}^T \mathrm{I}'(\alpha)\mathrm{w}\right)} \tag{11}$$

where α_i is the inverse variance of the prior distribution for weight w_1, and $I'(\alpha)$ is a $d \times d$ dimensional square matrix, with α_i's along its diagonal. When both prior and likelihood distributions of w are Gaussian, in[11] it is shown that the posterior distribution is also Gaussian with covariance C and mean m

$$\mathrm{C} = \beta\left(\beta\,\mathrm{X}\mathrm{X}^{\mathrm{T}} + \mathrm{I}'(\alpha)\right)^{-1} \tag{12}$$

$$\mathrm{m} = \beta\,\mathrm{C}\mathrm{X}\mathrm{y} \tag{13}$$

The predictive distribution of the target $\hat{\mathrm{y}}$ for an unobserved input vector $\hat{\mathrm{x}}$ is also Gaussian, whose mean and variance are

$$\mu = \mathrm{m}^{\mathrm{T}}\hat{\mathrm{x}} \tag{14}$$

and

$$\sigma^2 = \frac{1}{\beta} + \hat{\mathrm{x}}^{\mathrm{T}}\,\mathrm{C}\,\hat{\mathrm{x}} \tag{15}$$

For both of the P300-BCI datasets, we only use the mean value of the predictive distribution for taking decisions.

Channel Selection Algorithm

Efficiency of our P300-BCI depends on utilizing effective channels. In doing so, we apply the following two-step channel selection algorithm.

Step 1: In Step 1, we reduce to half the number of channels (from 64 to 32, or from 32 to 16) by using the Bhattacharyya distance. We sort BD values in decreasing order, and select the first half of channels with larger BD values.

Step 2: In Step 2, we employ an optimization algorithm to choose the more effective channels from channels selected in Step 1.

In[33], five different optimization approaches, namely, genetic, mimetic, ant-colony optimization, shuffled frog leaping, and particle swarm optimization (PSO) algorithms are compared for solving two benchmark continuous optimization test problems. It is shown that the PSO method outperforms the other methods in terms of convergence speed and accuracy of results, while being the second best in terms of processing time. In[34], statistical analysis and formal hypothesis testing are utilized to show that the PSO algorithm has the same effectiveness (finding the true global optimal solution) as the genetic algorithm (GA), but with significantly less calculations. Moreover, in[35], it is shown that when binary PSO (BPSO) is used for feature selection in the diagnosis of coronary artery disease, it yields better results than the GA. The BPSO is also used for channel selection in the motor imagery-based BCI.[20]

The PSO algorithm is a population-based search scheme based on the movement and flocking of birds that are called particles. Each particle flies in a n-dimensional search space with a certain velocity based on its own previously acquired knowledge and other particles experiences in the swarm. The position and the velocity of the i^{th} particle are denoted by $\mathrm{x}_i = \left(x_{i_1}, \cdots, x_{i_n}\right)$ and $\mathrm{v}_i = \left(v_{i_1}, \cdots, v_{i_n}\right)$, respectively. For each time step Δt, the corresponding velocity is applied to move each particle to its next position by

$$\mathrm{x}_i(t + \Delta t) = \mathrm{x}_i(t) + \mathrm{x}_i(t) \times \Delta t. \tag{16}$$

The step size Δt is usually set to 1, so at each iteration, the velocity and the position of each particle are updated by

$$\mathrm{v}_i^{t+1} = w \times \mathrm{v}_i^t + c_1 \times r_1 \times \left(\mathrm{p}_i^t - \mathrm{x}_i^t\right) + c_2 \times r_2 \times \left(\mathrm{g}^t - \mathrm{x}_i^t\right) \tag{17}$$

and

$$\mathrm{x}_i^{t+1} = \mathrm{x}_i^t + \mathrm{v}_i^t, \tag{18}$$

respectively, where p_i^t is the position of particle i with the highest value of C_{cs} up to iteration t, and g^t is p_i^t with the highest value of C_{cs} among all particles. Also, c_1 and c_2 are positive-valued learning factors, r_1 and r_2 are random numbers in [0, 1], and w is the inertia weight that represents the confidence of the particle to its current position, obtained from

$$w = w_{\max} - \frac{w_{\max} - w_{\min}}{t_{\max}} \times t \tag{19}$$

in which $w_{\min}$ and $w_{\max}$ are the final and the initial weights, respectively, $t_{\max}$ is the last iteration, and t is the current iteration. A large inertia weight facilitates a global search, while a small inertia weight facilitates a local search. From (19), we observe that the inertia weight decreases linearly from a relatively large value to a small value through the course of the PSO run. A linearly decreasing weight provides a better performance as compared to a fixed weight setting.

The velocity and the position of particles are confined to $[-v_{\max}\ v_{\max}]$ and $[-x_{\max}\ x_{\max}]$, respectively. This is to reduce the chances of particles flying out of the search space. Selecting the value of $v_{\max}$ is very important, since for very small values of $v_{\max}$, the step size has to be very small as well, which may cause the algorithm to trap in a local minima, or may take too long to converge. Also, for very large values of $v_{\max}$, a particle may go out of the search space, or its acceleration may exceed its limit.[36]

Assessment of all particles' positions is based on the value of C_{cs} score in (8) on the validation sets by using the BLDA classifier. The value of C_{cs} denotes the particle's position in a 64 or 32 dimensional space according to the five-fold cross-validation procedure that was described in Section 3.3.

In our problem, each particle is defined as a group of channels from the set of 32 or 16 channels selected by the Bhattacharyya criterion. We wish to prune the less effective channels and keep the more effective channels in the set of 32 or 16 selected channels (binary decision). In[37], the BPSO is used to search binary spaces on each dimension, where the position vector of each particle is binary-valued, and the velocity of a particle i was used to obtain the probability that the d^{th} bit of its position vector, i.e., x_{id}, takes on the value of 1 or 0. The velocity updating equation in the BPSO is the same as PSO, but the position of the d^{th} bit is updated by

$$x_{id}^{t+1} = \begin{cases} 1 \text{ if rand} < \text{sigmoid}(v_{id}^t) \\ 0 \qquad\qquad \text{otherwise} \end{cases} \tag{20}$$

where rand is a random number generated at t, and $\text{sigmoid}(v_{id}^t) = \dfrac{1}{1 + e^{-v_{id}^t}}$ maps the velocity to [0,1]. When the value of v_{id} is very large (positive or negative) the probability of a change in the bit value is one or zero, respectively.

We apply the BPSO algorithm to the set of Bhattacharyya pre-selected channels to choose the more effective channels, where each channel is an element of the vector that represents a particle. The value of each element can be either 1 or 0, where 1 means selection and 0 means rejection of the channel. As an example, for binary values of x_1 and x_2,

at iteration t in Figure 8, the corresponding two particles are $x_1^t = \{C_Z, FC_1, \cdots, PO_Z, AF_Z\}$ and $x_2^t = \{CP_Z, FC_1, \cdots, F_{PZ}, PO_8\}$.

The PSO algorithm suffers from the possibility of convergence to a local minima. In[38], a modified PSO is proposed that solves this problem by utilizing chaotic sequences for the weights in order to find a global solution that is better than the solution obtained by the PSO algorithm. The chaotic sequences are obtained by

$$f_t = \mu \times f_{t-1} \times (1 - f_{t-1}) \quad (21)$$

where μ is a control parameter that determines whether f tends to a fixed value, oscillates between a limited sequence of values, or behaves chaotically in an unpredictable manner. Also, the behavior of the system is influenced by the initial value of f. By choosing $\propto = 4$ and $f_0 \notin \{0, 0.25, 0.5, 0.75, 1\}$, the value of f corresponds to a chaotic sequence. Now, the new inertia weight is obtained by multiplying (19) by (21).

$$w_{new} = w \times f \quad (22)$$

Unlike the PSO algorithm in which the weight decreases monotonically from w_{max} to w_{min}, in the improved PSO, the new weight decreases and oscillates simultaneously as shown in Figure 9. We were inspired by the work in[38] to use the improved weights in BPSO algorithm and utilize the improved BPSO (IBPSO) to identify the more effective channels.

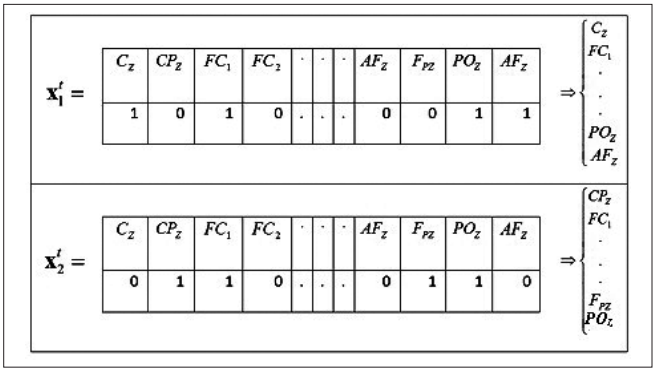

Figure 8: Binary particles in the IBPSO algorithm, where one means selection and zero means rejection of the channel

RESULTS

Experimental Result of Dataset 1

We now present the results of applying our proposed scheme to dataset IIb of BCI competition III in[21]. First, we compute the Bhattacharyya distance of each channel for subjects A and B by using target and non-target preprocessed signal samples. We sort the BD values in decreasing order, and select the first half of channels with larger BD values. The selected 32 channels for subjects A and B are listed in Table 1, respectively. We use the first eight channels of each subject, i.e., $[C_Z, CP_Z, P_Z, CP_2, FC_1, C_1, PO_7, F_Z]$ for subject A and $[PO_8, C_Z, CP_Z, C_2, O_1, PO_7, CP_2, FC_2]$ for subject B, to select the best truncated DWT coefficients as explained in Section 3.3. We begin by eliminating the near-zero coefficients from the beginning and the end parts of the DWT of single trial training data, as per Section 3.4; and obtain all possible combinations of the truncated DWT coefficients within

Table 1: The 32 channels sorted by BD criteria for subjects A and B

Subject	Sorted channels
A	$Cz, CP_Z, P_Z, CP_2, FC_1, C_1, PO_7, F_Z, FC_Z, O_1$
	$C_2, FC_2, CP_1, F_3, FC_4, P_1, PO_Z, C_6, CP_4, P_8, O_Z$
	$AF_Z, AF_8, PO_8, C_3, C_4, F_{P1}, F_{P2}, F_4, CP_3, P_3, AF_3$
B	$PO_8, C_Z, CP_Z, C_2, O_1, PO_7, CP_2, FC_2, PO_3, CP_Z$
	$C_1, I_Z, P_Z, CP_1, FC_1, F_4, AF_Z, PO_4, C_4, F_Z, PO_Z$
	$O_2, P_2, FC_6, P_1, O_Z, CP_4, AF_3, C_3, P_6, P_8, CP_6, P$

BD – Bhattacharyya distance

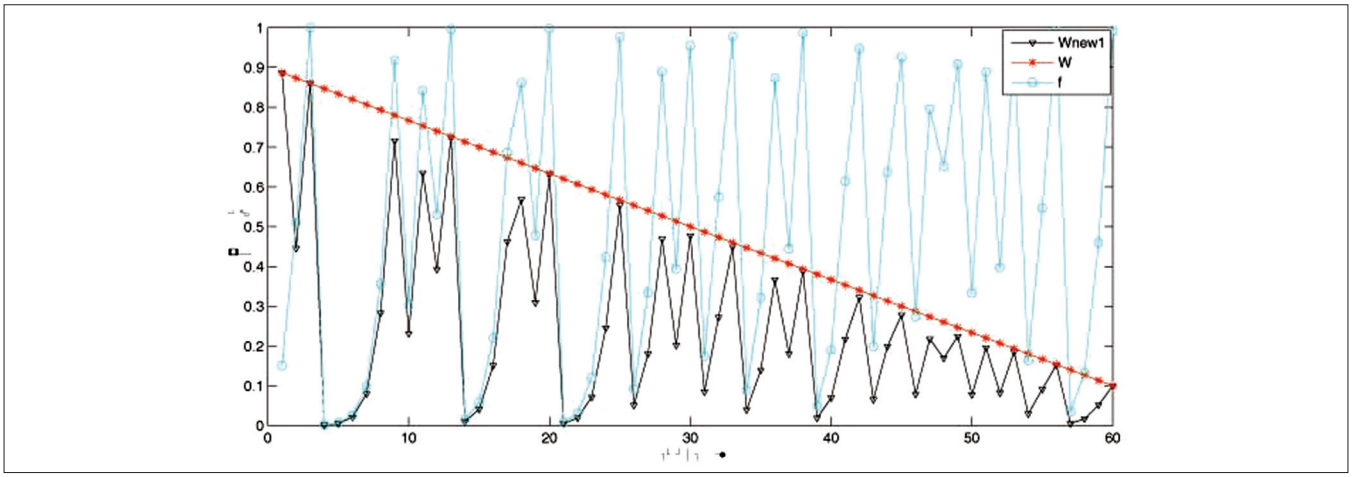

Figure 9: Variations in the conventional weight and in the proposed new weight[38]

0-30 Hz that do not overlap in frequency. The value of C_{cs} score in (8) for each combination set is obtained by the five-fold cross-validation procedure and the BLDA classifier. To compare the impact of using these coefficients vis-a-vis using all DWT and SWT coefficients, the mean classification accuracy for Subjects A and B are shown in Figure 10 for different trials by using the first 8 Bhattacharyya-selected channels. As can be seen, the classification accuracy for the SWT coefficients or for the selected sub-bands is not significantly better than those of the DWT coefficients. This also indicates that our results are not sensitive to varying shifts in the DWT. Our proposed scheme reduces the number of effective features about 20% for all DWT coefficients while maintaining accuracy.

As features, we apply the truncated coefficients of the 32 channels that were selected via the BD criteria [Table 1] for Subjects A and B, respectively, to the IBPSO algorithm in order to reduce the number of channels even further. We run the algorithm for 6, 8, 10, 12, and 15 particles (a particle is a subset of the 32 channels selected via the BD criteria) and 200 repetitions using the parameter values in Table 2, and observed that the highest C_{cs} is obtained when the number of particles in the IBPSO algorithm is 10. Figure 11 shows that C_{cs} for $\mathbf{g}^t$, i.e., $C_{cs}\left(\mathbf{g}^t\right)$, reaches its final value in less than 200 iterations for both subjects. The mean values of C_{cs} for $\mathbf{p}_i^t\big|_{i=1,2,\cdots,10}$, i.e., $\bar{C}_{cs}\left(\mathbf{p}_i^t\big|_{i=1,2,\cdots,10}\right)$, are also shown in Figure 11. Note that the value of $\bar{C}_{cs}\left(\mathbf{p}_i^t\big|_{i=1,2,\cdots,10}\right)$ does not change after 150 iterations for both subjects and its final value is the same as the final value of $C_{cs}\left(\mathbf{g}^t\right)$. This means that 200 iterations are sufficient and all ten position vectors $\mathbf{p}_i$ are able to follow g.

The IBPSO algorithm is executed 7 times separately to verify the consistency of channel selection. In each run, a different channel set is obtained, which shows the existence of local minima in the IBPSO. Note that $\{FC_1, C_3, C_1, C_Z, C_6, P_3, P_1, P_Z, PO_7, PO_Z, PO_8, O_1, O_Z\}$ channels for Subject A, and $\{C_3, C_Z, CP_Z, CP_6, P_6, P_8, PO_4, PO_3, PO_8, O_Z, I_Z\}$ channels for Subject B, are common among the six or seven sets. It shows that they are more important than the other channels. Note also that only $\{C_3, C_Z, PO_8, O_Z\}$ channels are common in both sets, and the rest are subject-dependent, meaning that channel selection should be performed on each subject separately.

Table 2: Parameter values for IBPSO

No of particles	10
Max. Iteration no	200
Weight parameters	w_{max}=0.9, w_{min}=0.1
Learning factors	c_1=c_2=2
Velocity constrains	v_{max}=5, v_{min}=-5
Control parameter	μ=4
Initial value of f	A random number between [0,1]>, except for 0, 0.25, 0.5, 0.75, 1.

IBPSO – Improved binary particle swarm optimization

Table 3 contains the classification accuracy of each channel set for Subjects A and B in 1, 5, and 15 trials. To show that our proposed scheme extracts effective features, we compare the classification accuracies for down-sampled signal, the DWT features, and the truncated DWT features in Table 4 by using the first channel set of each subject in Table 3. As can be seen, the classification accuracy of

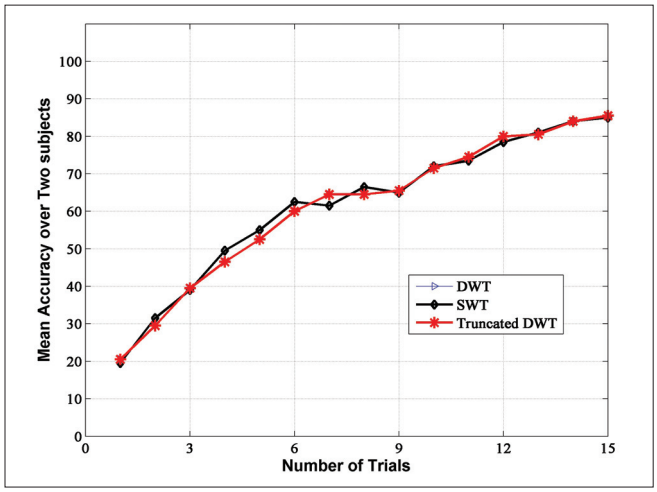

Figure 10: The mean classification accuracy over Subjects A and B for all DWT, truncated DWT, and SWT coefficients

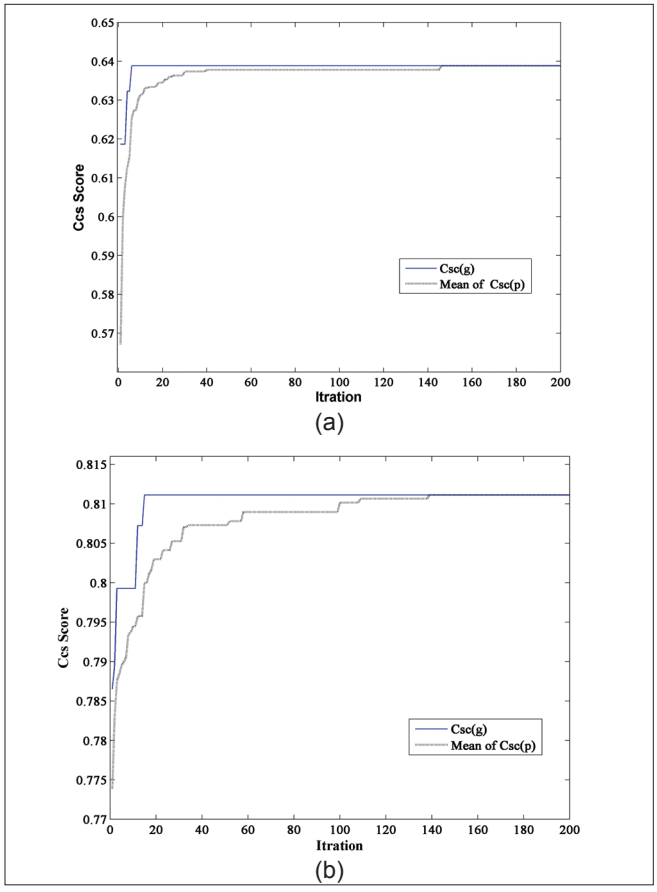

Figure 11: Variations of the Ccs score and the mean values of Ccs over ten particles for (a) subject A and (b) subject B

using the truncated DWT features in all trials except one item is equal to or higher than that of using the down-sampled signal. Moreover, the results of using the DWT and the truncated DWT features are exactly the same for all trials, meaning that by truncating those coefficients whose values are near zero, the classification accuracy is not deteriorated.

Classification results for both Subjects A and B in different trials are shown in Table 5. Using BCI 2005 evaluation criteria, we achieved a correct classification rate of 29%, 74.5%, and 97.5% in 1, 5, and 15 trials, respectively, as compared to the three best results of the BCI competition[9,10,21] shown in Table 5. As can be seen, in almost all trials, our results are better than those in[9,10,21], where the aim is accurate classification with less calculations.

In Table 6, we compare the number of channels in our approach with those of the three best results in the BCI competition. Note that we use fewer channels than the first ranked competitor.[9,10] Besides, we use the BLDA classifier that needs less calculations as compared to the SVM.

Table 3: Classification accuracy in % for selected channels by IBPSO in 1, 5, and 15 trials for subjects A and B

Subject	Channel set	Trials		
		1	5	15
A	$\{FC_2, FC_4, C_3, C_1, C_z, C_2, C_6, CP_3, CP_z, F_{P1}, F_{P2}, AF_z, F_z, P_3, P_1, P_z, P_8, PO_7, PO_z, PO_8, O_1, O_z\}$	21	68	97
	$\{FC_1, FC_z, FC_2, C_3, C_1, C_6, CP_3, CP_z, F_{P2}, AF_3, F_3, F_z, F_4, P_3, P_1, P_z, PO_7, PO_z, PO_8, O_1, O_z\}$	20	66	98
	$\{FC_1, FC_z, C_3, C_1, C_z, C_6, CP_3, CP_z, CP_2, AF_3, F_4, P_3, P_1, P_z, PO_7, PO_z, PO_8, O_1, O_z\}$	20	67	97
	$\{FC_1, C_3, C_1, C_z, C_6, CP_3, CP_1, F_{P1}, AF_3, AF_z, F_4, P_3, P_1, P_z, P_8, PO_7, PO_z, PO_8, O_1, O_z\}$	19	66	98
	$\{FC_1, FC_2, FC_4, C_3, C_z, C_2, C_6, CP_z, AF_3, F_z, F_4, P_1, P_z, P_8, PO_7, PO_z, PO_8, O_1, O_z\}$	21	67	97
	$\{FC_1, FC_2, C_3, C_1, C_z, C_6, CP_3, CP_1, F_{P1}, AF_3, AF_z, F_4, P_3, P_1, P_z, P_8, PO_7, PO_z, O_1, O_z\}$	20	67	97
	$\{FC_1, C_3, C_1, C_z, C_4, CP_3, CP_z, CP_4, AF_3, F_z, F_4, P_3, P_z, PO_7, PO_z, PO_8, O_1, O_z\}$	20	65	97
B	$\{FC_1, FC_6, C_3, C_1, C_z, C_2, CP_z, CP_4, CP_6, AF_3, F_4, P_2, P_6, P_8, PO_7, PO_z, PO_4, PO_8, O_1, O_z, I_z\}$	37	81	98
	$\{FC_6, C_3, C_1, C_z, C_2, CP_z, CP_6, AF_z, P_z, P_2, P_6, P_8, PO_7, PO_3, PO_z, PO_4, PO_8, O_z, O_2, I_z\}$	36	79	99
	$\{C_3, C_1, C_z, CP_z, CP_2, CP_6, AF_z, P_1, P_4, P_6, P_8, PO_7, PO_3, PO_z, PO_4, PO_8, O_z, O_2, I_z\}$	36	79	99
	$\{FC_2, C_3, C_z, C_2, CP_1, CP_z, CP_4, CP_6, AF_z, P_6, P_8, PO_7, PO_3, PO_4, PO_8, O_1, O_z, I_z\}$	37	78	99
	$\{FC_1, FC_6, C_3, C_1, C_z, C_2, CP_z, CP_4, F_z, P_4, P_6, P_8, PO_3, PO_z, PO_4, PO_8, O_1, O_z, I_z\}$	36	81	97
	$\{FC_2, C_3, C_z, C_2, CP_z, CP_2, CP_4, CP_6, AF_z, F_z, P_4, P_6, P_8, PO_7, PO_3, PO_4, PO_8, O_z, I_z\}$	36	78	99
	$\{FC_1, C_3, C_z, C_2, C_4, CP_1, CP_z, CP_2, CP_6, F_4, P_z, P_2, P_8, PO_3, PO_z, PO_4, PO_8, O_z, I_z\}$	36	79	98

IBPSO – Improved binary particle swarm optimization

Table 4: Classification accuracy in % for the down-sampled signal, the DWT coefficients and the truncated DWT coefficients

Subject	Type of features	Number of trials							
		1	2	3	4	5	10	13	15
A	Signal samples	19	33	45	56	63	82	92	93
	CA_3	21	37	49	59	68	83	94	97
	Trun. CA_3	21	37	49	59	68	83	94	97
B	Signal samples	37	54	65	75	78	94	93	96
	CA_4	37	55	64	75	81	95	97	98
	Trun. CA_4	37	55	64	75	81	95	97	98

DWT – Discrete wavelet transform

Table 5: Mean classification accuracy of our scheme in % and the first ranked competitor in BCI competition 2005, dataset IIb, and[9,10] for subjects A and B

	Number of trials							
	1	2	3	4	5	10	13	15
Our scheme	29	46	56.5	67	74.5	89	95.5	97.5
First ranked	25.5	42.5	57	64	73.5	87	95	96.5
[9]	31	46	56	65	71.5	87.5	90	95
[10]	28	–	53	–	71	–	94	97.5

BCI – Brain-computer interface

Experimental Result of Dataset 2

We use the data recorded in the first three sessions and the last session as the training and the test data, respectively, for disabled subjects (Subject 1-Subject 4) and able-bodied subjects (Subject 6-Subject 9). Data for Subject 5 is not considered in this paper for reasons stated in[11] The EEG signals was down sampled from 2048 to 256 samples per second by selecting every 8^{th} sample from the bandpass-filtered data as described in Section 3.1. For each session, the single trials corresponding to first 20 blocks of flashes were extracted via preprocessing. Hence, a single trial includes 180 samples per trial, as compared to 168 samples per trial for dataset 1. Each block consists of six flashing images, and so the training data is comprised of 360 target trials and 1800 non-target trials. The test data consists of 120 target and 600 non-target trials. For each subject, we reduce the number of channels from 32 to 16 by using the sorted BD values in decreasing order. The first eight channels were used to select the best truncated sub-bands as described in Sections 3.3 and 3.4. Table 7 shows the best truncated DWT coefficients and their length for each subject by using the five-fold cross-validation procedure with cost function C_{cs} and BLDA classifier. Note that in Figure 12, the mean classification accuracy for eight subjects, corresponding to the truncated DWT coefficients in Table 7, are exactly the same as those of utilizing all DWT coefficients (no truncation). Besides, note that using a higher number of SWT features is not very beneficial.

In order to select the final channel sets, we run the IBPSO algorithm by using the selected truncated DWT coefficients for 16 remaining channels that were identified via the BD criteria. Since the number of input channels to IBPSO algorithm in this dataset is half of the input channels in the previous dataset, we used 100 iterations instead of 200 iterations. The other parameters of the IBPSO algorithm are stated in Table 2. For each subject, we run the IBPSO algorithm seven times by using C_{cs}, the BLDA classifier and five fold cross-validation procedure. In each run, we observed that the values of $C_{cs}\left(g^t\right)$ and $\bar{C}_{cs}\left(p_i^t\big|_{i=1,2,\cdots,10}\right)$ do not change after 80 iterations for all subjects, which indicates that 100 iterations are sufficient. Table 8 shows the best selected channel set in 7 runs of the IBPSO for each subject. For each subject, some channel sets were similar in 7 runs, which shows better convergence of the IBPSO algorithm as compared to dataset 1 due to fewer input channels.

For each subject, feature vectors are the truncated DWT coefficients in Table 7, and the channel sets are obtained by the IBPSO algorithm. Hence, we obtained seven different feature vectors corresponding to seven output channel sets

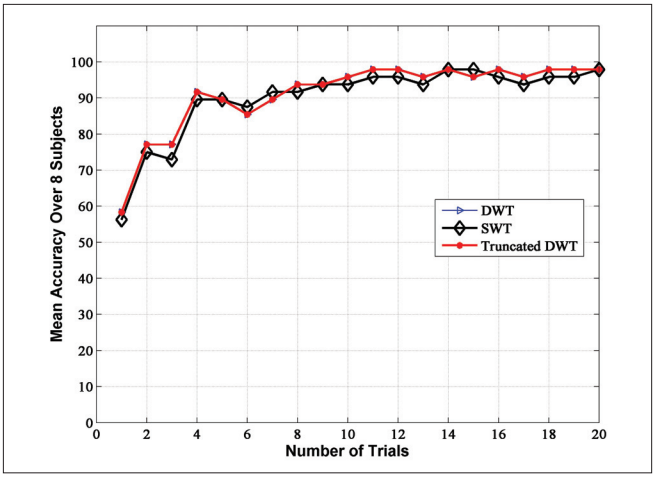

Figure 12: The mean classification accuracy for 8 subjects for all DWT, truncated DWT, and SWT coefficients

Table 6: No. of channels and classifiers' types in our scheme and the three best competitors in BCI competition 2005, dataset IIb

Algorithms	Number of channels		Classifiers
	Subject A	Subject B	
Our scheme	22	21	BLDA
First ranked	Almost all 64	Almost all 64	Ensemble SVM
[9]	Almost all 64	Almost all 64	Ensemble FLD
[10]	32	32	BLDA

BCI – Brain-computer interface; BLDA – Bayesian linear discriminant analysis; SVM – Support vector machine; FLD – Fisher linear discriminant

Table 7: The best selected features (truncated DWT coefficients) and length of the feature vector using the five-fold cross-validation procedure and BLDA classifier for 8 subjects

Subject	The best feature vector	Feat. length
S1	Truncated CA_3	25
S2	Truncated CA_4	14
S3	Truncated CA_3	25
S4	Truncated CA_4	14
S6	Truncated CA_4	14
S7	Truncated CA_3	25
S8	Truncated CA_4	14
S9	Truncated [CA_6, CD_5, CD_4]	29

DWT – Discrete wavelet transform; BLDA – Bayesian linear discriminant analysis

Table 8: The best selected channel-sets by IBPSO for all 8 subjects

Subject	The best channel set
1	$F_{P1}, P_7, PO_3, O_1, O_2, P_8, F_4, FP_2, F_Z, C_Z$
2	$F_{P1}, AF_3, F_7, P_7, P_Z, O_1, O_2, F_Z$
3	$P_7, P_3, P_Z, PO_3, O_1, O_Z, PO_4, FC_2, F_Z$
4	$F_{P1}, C_3, P_7, P_Z, CP_2, FC_2$
6	$F_{P1}, AF_3, P_7, P_8, C_4, FC_2, FP_2, C_Z$
7	P_7, P_Z, O_2, F_Z
8	$F_{P1}, CP_5, P_7, P_3, O_1, O_Z, PO_4, P_8, C_4, FC_2, FP_2$
9	$C_3, P_7, P_Z, O_1, O_2, P_8$

IBPSO – Improved binary particle swarm optimization

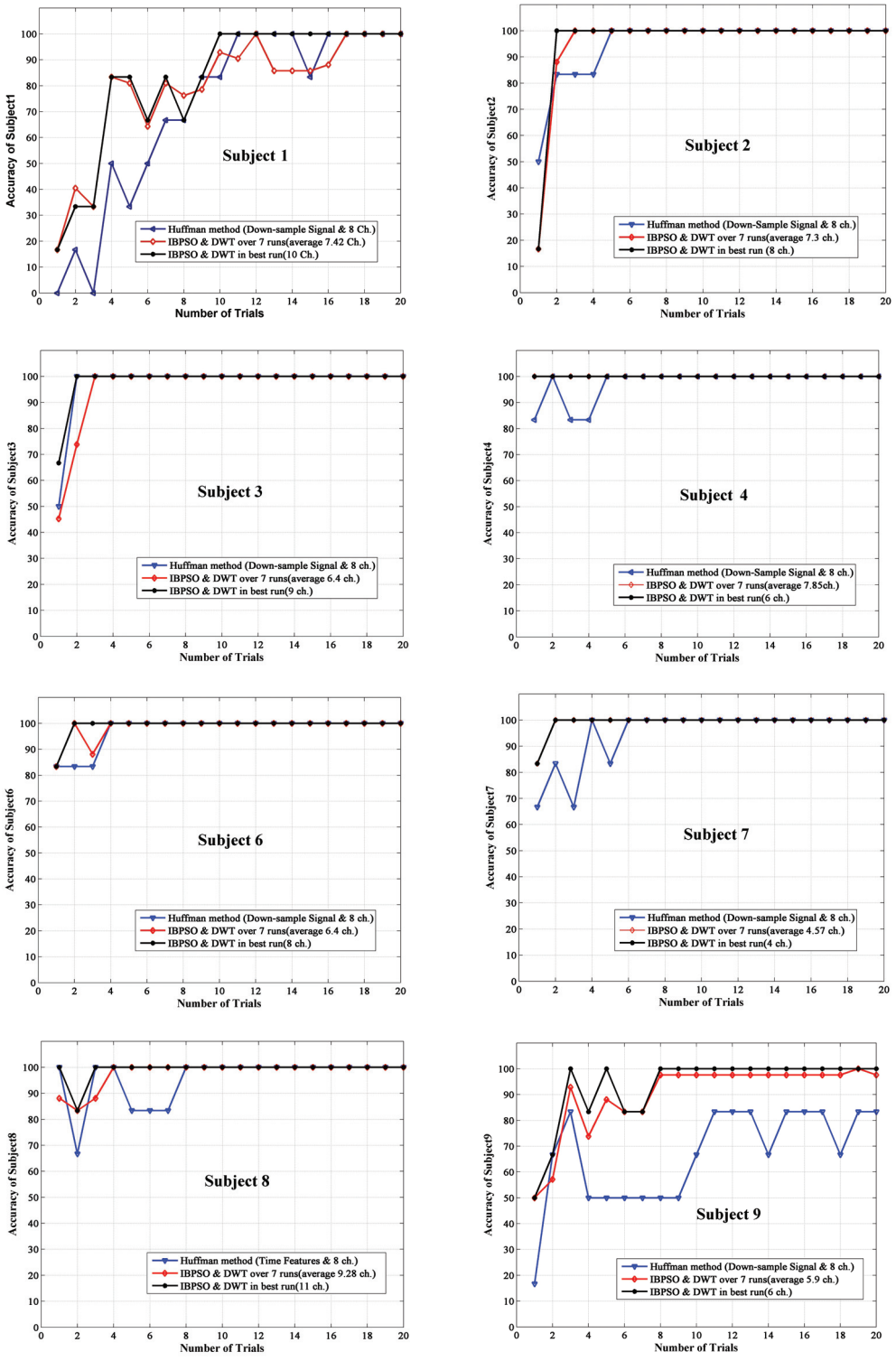

Figure 13: Classification accuracy of the best channel set and the average classification accuracies over 7 channel sets in our approach and those obtained by using the method in[11] for $CH_{set\ 2}$, for disabled subjects (subject 1-subject 4) and able-bodied subjects (subject 6-subject 9)

of the IBPSO. Extracted feature vectors from single trials (including targets and non-targets) are used to train a BLDA classifier. Classification accuracy is computed by using the extracted features of the test data (the data from the fourth session) over different trials and for seven channel sets.

To compare the classification accuracy of our scheme with that of the method proposed in[11], we use the same pre-processed signal samples and the same four different channel sets consisting of 4, 8, 16, and 32 electrodes. In both cases, we use the data from the first three sessions for each subject to

select features and channels, and train the classifier; and the data from the fourth session to compute the classification accuracy. Note that the four channel sets used in[11] are $CH_{set1} = \{F_Z, C_Z, P_Z, O_Z\}$, $CH_{set2} = \{F_Z, C_Z, P_Z, O_Z, P_3, P_4, P_7, P_8\}$, $CH_{set3} = \{F_Z, C_Z, P_Z, O_Z, P_3, P_4, P_7, P_8, FC_1, FC_2, C_3, C_4, CP_1, CP_2, O_1, O_2\}$, and $CH_{set4} = \{\text{all 32 channels}\}$ in Figure 2b. Figure 13 compares the classification accuracies of the best channel set and the average classification accuracies over seven channel sets in our approach with those in[11] for CH_{set2} for each subject. For the best channel set, the performance of our method for all subjects and trials except for one case (the first trial of Subject 2) is significantly better than those in[11] for CH_{set2}. As shown in Figure 13, the average classification accuracy over seven chanwnel sets except for very few trials for Subjects 1, 2, 3, 8, 9 is better than those in[11] for CH_{set2}. The performance of our proposed scheme for both disabled and able-bodied subjects does not differ much.

In Figure 14, the average classification accuracy for all subjects in our proposed scheme for the truncated DWT coefficients and the channels identified by the IBPSO algorithm is compared with those in[11] that utilizes the down-sampled signal and four different channel sets. As can be seen, compared to CH_{set1}, CH_{set2}, and CH_{set3} channel sets, our proposed scheme performs better or the same as in.[11] Moreover, the average classification accuracy over seven sets of channels obtained by the IBPSO algorithm is approximately the same as those in[11] for CH_{set4} (with 32 channels), while we use less channels (with average 6.9 channels per subject).

Note that the results of using down-sampled signal and four different channel sets in Figures 13 and 14 are different from those in[11] due to the fact that classification accuracy in the latter is obtained by averaging over four sessions, whereas we only use the fourth session to compute classification accuracy. For a better comparison, we repeated our proposed procedure four times, and each time, we used three different sessions for selecting features and channels, and for training the classifier. The fourth session is used for computing the classifier accuracy. Figure 15 compares the average classification accuracy of our method over four sessions and over all subjects with those in[11] for 8 channels and 32 channels. As can be seen, the average classification accuracy of our method (with average 7.3 channels per subject) over four sessions and over all subjects is approximately the same as the best result (with 32 channels) in[11], confirming the results in Figures 13 and 14.

DISCUSSION

Analysis of EEG signals in the BCI system consists of preprocessing, feature extraction, channel selection, and data classification. While in[8-11] the focus is mainly on channel selection, and in[7,13], the focus is on feature selection, we focus on both channel and feature selection with a view

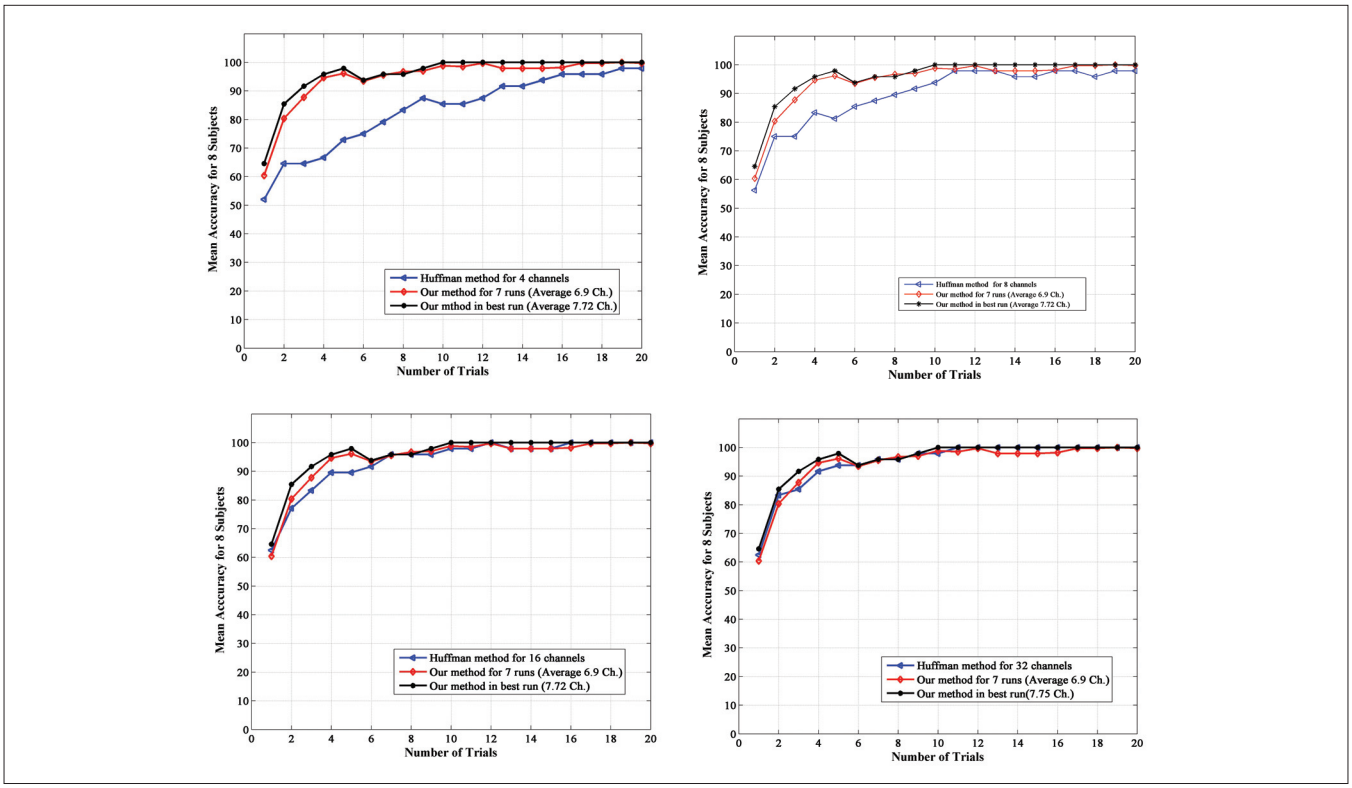

Figure 14: The average classification accuracy for all subjects in our proposed scheme (truncated DWT coefficients for best run and an average of 7 runs for the IBPSO algorithm) and those in[11] for $CH_{set\ 1}$, $CH_{set\ 2}$, $CH_{set\ 3}$, and $CH_{set\ 4}$ channel sets

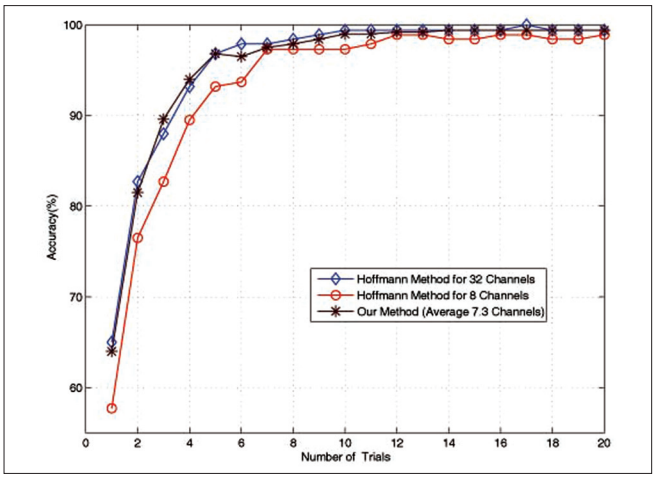

Figure 15: The average classification accuracy of our method over four sessions and over all subjects and those in[11] for CH_{set2}, and CH_{set4} channel sets

to improving classification accuracy. The proposed scheme needs less features and provides more accurate classifications for almost all trials and subjects in real time. However, our method for selecting proper features and channels during training is not as simple as those in[10,11].

We truncated the DWT coefficients to reduce the number of features, while in[9] all DWT coefficients in each level are used. Furthermore, the number of features in our scheme is less than the number of preprocessed signal samples in[8,10,11]. Note that, we can reduce the number of features up to 30% while maintaining the same accuracy in different trials for all subjects. We also showed that using shift-invariant wavelet transform with a large number of features does not produce better results than using DWT that is shift-varying [Figures 10 and 12].

In order to improve the accuracy, we removed ineffective channels by applying a two-step channel selection algorithm (Bhattacharyya distance and IBPSO algorithm). For dataset 1, we used 22 channels for Subject A, and 21 Channels for Subject B. This is in contrast to[8,9,12] that use almost all 64 channels and more features, resulting in more calculations. In dataset 1 for some trials, the performance of our scheme is below that of the first ranked competitor and.[9,10] For Subject B, our proposed algorithm provides better results as compared to[8,10] for all trials. In dataset 2, we can approximately achieve the same classification accuracy with an average 6.9 channels per subject as compared to[10] with 32 channels and more features. Compared to three other channel sets (i.e., 4, 8, and 16 channels) in[10], our results are better or equal in all trials.

Another important issue in BCI is choosing a classifier that provides fast discrimination between classes. SVM is a well-known and powerful classifier used by the first and the second ranked competitors, but it requires more calculations to tune its parameters, and gets worse when the training data is extensive. In this study, we use the BLDA classifier instead of SVM as in.[10,11] As can be seen in Table 5, the accuracy of our proposed classifier in almost all trials is higher than those of the first ranked competitor. In[9], the FLDA classifier (which is slightly simpler than the BLDA classifier) is used for evaluating classification accuracy in a configuration that consists of 10 parallel classifiers. However, our proposed scheme is more accurate than,[9] except for Subject A with less than five trials.

The results show that the selected channels and sub-bands were different among subjects in both datasets. This indicates that the set of optimal electrodes and the set of optimal DWT sub-bands are subject dependent.

CONCLUSIONS

Three performance indicators, namely computation cost, real time, and accuracy, are essential in BCI applications. To achieve these objectives, we proposed a new scheme for selecting a minimal set of features by utilizing DWT and mother wavelet db4, and choose the more effective channels. In particular, we used truncated wavelets when the coefficients' values are small (near zero) and selected optimal DWT sub-bands for each subject. We also used the BD and the IBPSO algorithm to select fewer channels for attaining accurate classification as compared to existing methods. In particular, using BD to eliminate one half of channels significantly reduces calculations in the two different P300-BCI datasets that include 10 disabled and able-bodied subjects. Our method is subject-dependent, and uses a two-stage procedure in the training phase to select the best sets of sub-bands and channels, resulting is more accurate classification, with less features and less channels.

REFERENCES

1. Wolpaw J, Birbaumer N, McFarland DJ, Pfurtscheller G, Vaughan TM. Brain-computer interfaces for communication and control. Clin Neurophysiol 2002;113:767-91.
2. Sutton S, Braren M, Zubin J, John ER. Evoked-potential correlates of stimulus uncertainty. Science 1965;150:1187-8.
3. Farwell LA, Donchin E. Talking off the top of your head: A mental prosthesis utilizing event-related brain potentials. Electroencephalogr Clin Neurophysiol 1988;70:510-23.
4. Donchin E, Spencer KM, Wijesinghe R. The mental prosthesis: Assessing the speed of a P300-based brain–computer interface. IEEE Trans Rehabil Eng 2000;8:174-9.
5. Salvaris M, Sepulveda F. Visual modifications on the P300 speller BCI paradigm. J Neural Eng 2009;6:1-8.
6. Sellers EW, Krusienski DJ, McFarland DJ, Vaughan TM, Wolpaw JR. A P300 event-related potential brain–computer interface BCI: The effects of matrix size and inter stimulus interval on performance. Biol Psychol 2006;73:242-52.
7. Takano K, Komatsu T, Hata N, Nakajima Y, Kansaku K. Visual stimuli for the P300 brain–computer interface: A comparison of white/gray and green/blue flicker matrices. Clin Neurophysiol 2009;120:1562-6.
8. Rakotomamonjy A, Guigue V. BCI competition III: Dataset II-ensemble of SVMs for BCI P300. IEEE Trans Biomed Eng 2008;55:1147-54.
9. Salvaris M, Sepulveda F. Wavelets and ensemble of FLDs for P300 classification. Proc. Int. IEEE EMBS Conf. on Neural Engineering. Antalya, Turkey: 2009. p. 339-42.

10. Selim AE, Wahed MA, Kadah VM. Machine learning methodologies in P300 speller Brain-Computer Interface systems. Proc. National Radio Science Conf. New Cairo, Egypt: 2009. p. 1-9.
11. Hoffmann U, Vesin JM, Ebrahimi T, Diserens K. An efficient P300-based brain-computer interface for disabled subjects. J Neurosci Methods 2008;167:115-25.
12. Liu Y, Zhou Z, Hut D, Dong G. T-weighted approach for neural information processing in P300 based brain-computer interface. Proc. Int. Conf. on Neural Networks and Brain. Beijing, China: 2005. p. 1535-9.
13. Bostanov V. 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 2004;51:1057-61.
14. Markazi S, Qazi S. Wavelet filtering of the P300 component in event-related potentials. Proc. IEEE EMBS Annual Int. Conf. New York City, USA: 2006. p. 1719-22.
15. Markazi SA, Stergioulas LK. Latency corrected wavelet filtering of the P300 event-related potential in young and old adults. Proc. Int. IEEE EMBS Conf. on Neural Engineering. Hawaii, USA: 2007, p. 582-6.
16. Yong YP, Hurley NJ, Silvestre GC. Single-trial EEG classification for brain-computer interface using wavelet decomposition. Proc. European Signal Processing Conf. Antalya, Turkey: 2005.
17. Subasi A. EEG signal classification using wavelet feature extraction and a mixture of expert model. Expert Sys Appl 2007;32:1084-93.
18. Thulasidas M, Guan C. Optimization of BCI speller based on P300 potential Proc. Annual Int. Conf. of the IEEE Engineering in Medicine and Biology Society. Shanghai, China: 2005. p. 5396-9.
19. Yang L, Li J, Yao Y, Li G. An algorithm to detect P300 potentials based on F-score channel selection and support vector machines. Proc. Int. Conf. on Natural Computation. Haikou, China: 2007. p. 280-4.
20. Hasan BA, Gan J, Lee W, Zhang Q. Multi-objective evolutionary methods for channel selection in brain-computer interfaces: Some preliminary experimental results. Proc. World Congress on Computational Intelligence. Barcelona, Spain: 2010. p. 3339-44.
21. Blankertz B. The BCI competition III [Online] Fraunhofer FIRST IDA. Available from: http://www.ida.first.fraunhofer.de/projects/bci/competition iii. [Last cited in 2005].
22. Fatourechi M, Birch GE, Ward RK. Application of a hybrid wavelet feature selection method in the design of a self-paced brain interface system. J Neuroeng Rehabil 2003;4:1-13.
23. Theodoridis S, Koutroumbas K. Pattern Recognition. 2nd ed. San Diego, CA: Elsevier; 2003.
24. Choi E, Lee C. Feature extraction based on the Bhattacharyya distance. Pattern Recognit Lett 2003;36:1703-9.
25. Mallat S. Theory for multiresolution signal decomposition: The wavelet representation. IEEE Trans Pattern Anal Mach Intell 1989;11:674-93.
26. Bradley AP. Shift-invariance in the discrete wavelet transform. Proc. Int. Conf. on Digital Image Computing: Techniques and Applications. Sydney, Australia: 2003. p. 29-38.
27. Tsiaparas NN, Golemati S, Andreadis I, Stoitsis JS, Valavanis I, Nikita KS. Comparison of multiresolution features for texture classification of carotid atherosclerosis from b-mode ultrasound. IEEE Trans Inf Technol Biomed 2011;15:130-7.
28. Addison PS, Walker J, Guido RC. Time–frequency analysis of biosignals: A wavelet transform overview. IEEE Eng Med Biol Mag 2009;28:14-29.
29. Bradley A, Wilson W. On wavelet analysis of auditory evoked potentials. Clin Neurophysiol 2004;115:1114-28.
30. Cabrera AF, Dremstrup K. Auditory and spatial navigation imagery in Brain–Computer Interface using optimized wavelets. J Neurosci Methods 2008;174:135-46.
31. Cvetkovic D, Ubeyli ED, Cosic I. Wavelet transform feature extraction from human PPG, ECG, and EEG signal responses to ELF PEMF exposures: A pilot study. Digit Signal Process 2008;18:861-74.
32. Lei X, Yang P, Yao D. An empirical Bayesian framework for brain–computer interfaces. IEEE Trans Neural Syst Rehabil Eng 2009;17:521-9.
33. Elbeltagi E, Hegazy T, Grierson D. Comparison among five evolutionary-based optimization algorithms. J Adv Engng Informatics 2005;19:43-53.
34. Hassan R, Cohanim B, Weck O, Venter G. A comparison of particle swarm optimization and the genetic algorithm. Proc. AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conf. Austin, Texas: 2005. p. 18-21.
35. Babaoglu I, Findik O, Lkera E. A comparison of feature selection models utilizing binary particle swarm optimization and genetic algorithm in determining coronary artery disease using support vector machine. Expert Syst Appl 2010;37:3177-83.
36. Lee S, Soak S, Oh S, Pedrycz W, Jeon M. Modified binary particle swarm optimization. Progress in Natural Science 2008;18:1161-6.
37. Kennedy J, Eberhart RC. A discrete binary version of the particle swarm algorithm. Proc. Int. Conf. on Systems, Man and Cybernetics. Orlando, USA: 1997. p. 4104-9.
38. Park J, Jeong Y, Lee W, Shin J. An improved particle swarm optimization for economic dispatch problems with non-smooth cost functions. Proc. Int. Conf. on Machine Learning and Cybernetics. Dalian, China: 2006. p. 396-401.

How to cite this article: How to cite this article: Perseh B, Sharafat AR. An Efficient P300-based BCI Using Wavelet Features and IBPSO-based Channel Selection. J Med Sign Sens 2012;2:128-43.

Source of Support: Tarbiat Modares University, **Conflict of Interest:** None declared

BIOGRAPHIES

Bahram Perseh was born in Tehran, Iran in 1970. He received his B.S. in Electrical Engineering from Isfahan University of Technology in 1993 and his M.S. degree in Biomedical Engineering from Amirkabir University of Technology (The Tehran Polytechnic) in 1996. He is currently working towards the Ph.D. degree in Electrical and Computer Engineering at Tarbiat Modares University, Tehran, Iran. His research interests include biomedical signal processing, brain–computer interface (BCI), heart sound analysis, and pattern recognition.

E-mail: bahramperse@yahoo.com

Ahmad R. Sharafat is a professor of Electrical and Computer Engineering at Tarbiat Modares University, Tehran, Iran. He received his B.Sc. degree from Sharif University of Technology, Tehran, Iran, and his M.Sc. and his Ph.D. degrees both from Stanford University, Stanford, California, all in Electrical Engineering in 1975, 1976, and 1981, respectively. His research interests are advanced signal processing techniques, and communications systems and networks. He is a Senior Member of the IEEE and Sigma Xi.

E-mail: sharafat@modares.ac.ir