

# Multiple classifier system for EEG signal classification with application to brain–computer interfaces

Amir Ahangi · Mehdi Karamnejad ·  
Nima Mohammadi · Reza Ebrahimpour ·  
Nasoor Bagheri

Received: 22 February 2012 / Accepted: 3 July 2012  
© Springer-Verlag London Limited 2012

**Abstract** In this paper, we demonstrate the use of a multiple classifier system for classification of electroencephalogram (EEG) signals. The main purpose of this paper is to apply several approaches to classify motor imageries originating from the brain in a more robust manner. For this study, dataset II from BCI competition III was used. To extract features from the brain signal, discrete wavelet transform decomposition was used. Then, several classic classifiers were implemented to be utilized in the multiple classifier system, which outperforms the reported results of other proposed methods on the dataset. Also, a variety of classifier combination methods along with genetic algorithm feature selection were evaluated and compared in order to diminish classification error. Our results suggest that an ensemble system can be employed to boost EEG classification accuracy.

**Keywords** EEG classification · Motor imagery · Wavelet feature extraction · Feature selection · Multiple classifier system (MCS)

## 1 Introduction

A brain–computer interface (BCI) presents one of the most promising assistive technologies for improving the quality of life for physically impaired individuals, by providing a communication channel between the brain of the patient and the outside world, such that mental activities can be used to coordinate actions taking place in the environment [1]. The electrical activity of the brain, which is monitored using invasive or noninvasive methods during specific cognitive tasks, is associated with various controlling commands by a set of pattern recognition algorithms comprising the brain–computer interface.

BCIs are primarily used as a rehabilitation strategy for patients in a late stage of amyotrophic lateral sclerosis (ALS) or locked-in syndrome [2]. Other applications of BCI include research in neuroscience, controlling robotic elements, gaming and virtual reality [3–8]. Sensorimotor cortex in humans is responsible for generating neural activity related to execution or imagination of movement. Whenever one imagines movement (i.e., motor imagery), these specific brain rhythms first become attenuated, and then stronger. These two changes are called event-related desynchronization (ERD) and synchronization (ERS), respectively. Hence, the patterns generated by motor imagery can be exploited in a BCI. Figure 1 visualizes ERD/ERS components.

A BCI based on motor imagery interprets movement-related oscillatory patterns of a subject during imagination of different tasks (e.g., moving hands and feet) into

---

A. Ahangi · M. Karamnejad · N. Mohammadi ·  
R. Ebrahimpour (✉) · N. Bagheri  
Brain Intelligent and Systems Research Lab (BISLAB),  
Department of Electrical and Computer Engineering,  
Shahid Rajaei Teacher Training University (SRTTU),  
P.O.Box: 16785-163, Tehran, Iran  
e-mail: ebrahimpour@ipm.ir; rebrahimpour@srttu.edu

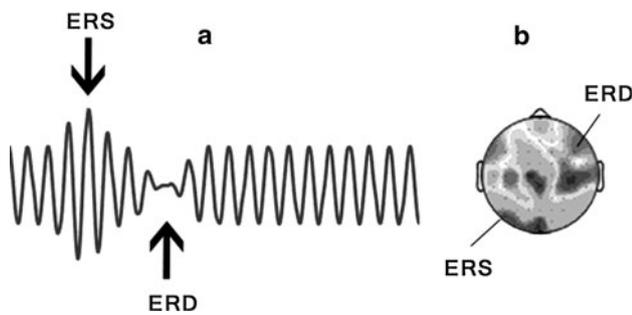
A. Ahangi  
e-mail: Amir.Ahanghi@srttu.edu

M. Karamnejad  
e-mail: Mehdi.Karamnejad@srttu.edu

N. Mohammadi  
e-mail: Nima.Mohammadi@srttu.edu

N. Bagheri  
e-mail: NBagheri@srttu.edu

R. Ebrahimpour  
School of Cognitive Sciences (SCS), Institute for Research  
in Fundamental Sciences (IPM), Niavaran,  
P.O.Box: 19395-5746, Tehran, Iran



**Fig. 1** a ERD/ERS components in an observed brain signal. b Topographical map of ERD/ERS components in a hand movement motor imagery (The image originally appeared in [9])

controlling commands [10–12].  $\mu$  (8–12 Hz) and  $\beta$  (13–30 Hz) waves originating from sensorimotor cortex are the ones depicting the manifestation of ERD/ERS phenomena [13–17].

In the frequency domain, Butterworth band-pass filter is commonly used to discard undesirable portions of a signal [18–21].

Based on the fact that EEG signals are non-stationary, traditional methods for feature extraction like Fourier transform are not appropriate for analyzing such data. Wavelet is a time-frequency analysis method which decomposes the signal into several scales and is capable of discarding some of decomposed signals which makes it a suitable choice for signal processing [22–25]. Other methods used for feature extraction include CSP, AAR parameters, AR spectral power and principal component analysis (PCA) [26].

CSP is used to filter channels containing the most informative and distinctive data from EEG. However, in this study, the raw EEG data being utilized are already containing three channels with the most informative data (i.e., Cz, C3, C4 that are considered to have the most informative data in a BCI based on motor imagery [27]); thus, despite the CSP being customary in motor-imagery-based BCI, it was not our first choice for feature extraction phase. Additionally, for the ensemble system to be productive, a diverse and distinctive feature space should be fed to experts. Also, in the genetic algorithm feature selection phase, there must be enough number of features to be optimized. Using wavelet transform and statistical measurements, a reasonably adequate number of features are generated making this transform ideal for our study. On the other hand, CSP does not generate as many diverse features as required for the GA feature selection and the ensemble system in our dataset.

Ensemble systems can be used to improve the productivity achieved by a single classifier [28]. They move toward improving the performance of classification by considering decisions made by different types of classifiers. There are several reasons, which motivates one to exploit

this technique such as small number of data samples for training and high dimension of data samples that makes classification process difficult for a single classifier [29–31]. Esmaeili in [32] reported better EEG classification accuracy using multiple classifier combination.

Combining procedures can be divided into different categories from a variety of perspectives, three of which are considered here. The first perspective divides combining strategies into classifier selection and classifier fusion. Classifier selection follows a divide-and-conquer approach by assigning a particular part of the problem space to each classifier [33]. On the other hand, in classifier fusion, all classifiers are trained over the entire problem space [31].

From the second viewpoint, two taxonomies are considered: trainable and non-trainable combining strategies. Non-trainable combiners are fixed algebraic rules, such as max, min, average or majority voting, while trainable combination rules, such as Stacked Generalization, determine their parameters during a learning procedure [29].

At last, considering how input data are involved with construction of ensemble, there are two main types of combining classifiers: static and dynamic. Dynamic techniques choose an ensemble specifically for each sample from a large pool of classifiers [34], as opposed to static ensemble construction methods which rely on the same set of classifiers for all samples.

In the present study, movement thoughts of left and right hand are primarily represented distinctively using discrete wavelet transform (DWT) and then classified using an ensemble of classifiers. To perform the classification in a better manner, we strived to train each classifier in the ensemble system using different training samples to increase diversity of each classifier. This also improves the accuracy of the ensemble system.

The organization of this study is as follows: The dataset used for this study is elucidated in Sect. 2. The approach used for preprocessing the raw signal using Butterworth band-pass filter is described in Sect. 3.1. Feature extraction using discrete wavelet transform is discussed in Sect. 3.2. Different single classifiers utilized is briefly presented in Sect. 3.3. Multiple classifier system used to improve EEG signal classification is demonstrated in Sect. 3.4. The experimental results are presented in Sect. 4. Finally, the study is concluded in Sect. 5

## 2 EEG data

For this study, dataset III of BCI competition II was used [35]. This dataset was obtained from a normal 25-year-old female sitting on a relaxing chair. The tasks assigned to the subject were motor imageries of left or right hand occurring randomly.

The dataset concludes 280 9-second fixed trials. The first two seconds of each trial are quiet. At  $t = 3$ , an arrow cue pointed at right or left is displayed for 1 s and the subject is required to move a bar into the direction of the cue. g.tech amplifier and Ag/AgCl electrodes utilizing three bipolar EEG channels, measured over C3, Cz, and C4, were used to record the signals.

The EEG was sampled at 128 Hz and band-pass filtered between 0.5 and 30 Hz afterward. 140 trials are reserved for training and 140 trials are reserved for testing, which were randomly selected from the entire 280 trials.

### 3 Methodology

#### 3.1 Preprocessing

In order to prepare the original signal obtained from each channel for feature extraction, we first extracted from  $t = 4$  to  $t = 9$ . Then, we extracted the portion of signal consisting the  $\mu$  and  $\beta$  frequency bands, in which motor imageries occur. To achieve this goal, we used a sixth-order Butterworth band-pass filter. Figure 2 demonstrates raw and preprocessed signal from C3 channel in one epoch in the frequency domain.

#### 3.2 Feature extraction

Experimenting different permutation of available channels led to the conclusion that using C3 and C4 channels for extracting features results in a more discriminative feature space.

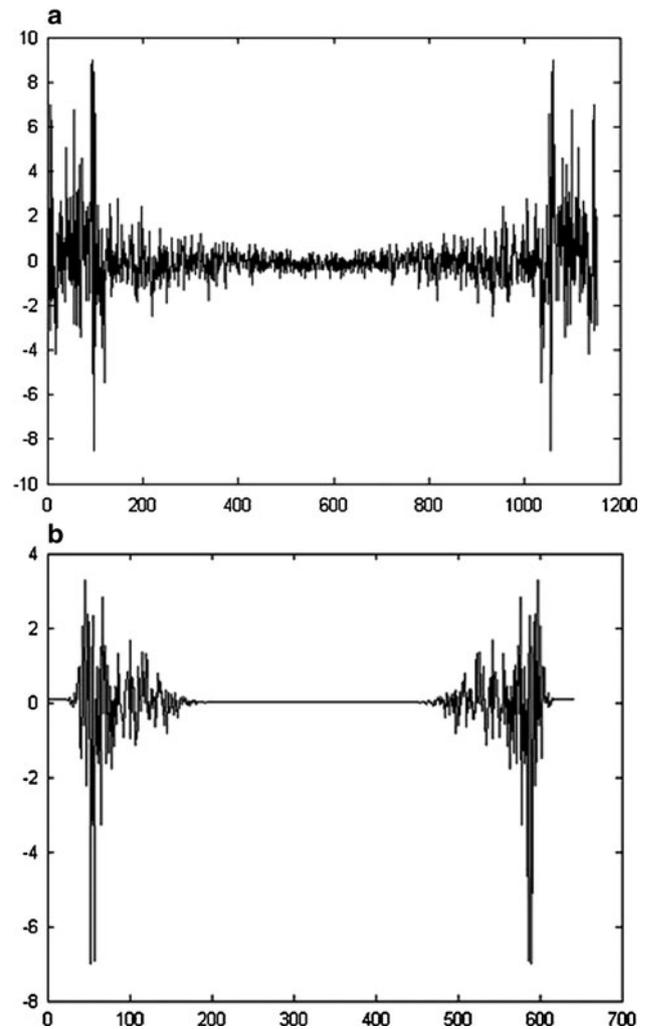
We applied discrete wavelet transform in each stage and decomposed the signal into detail and approximation coefficients, respectively, representing low-frequency and high-frequency components. Then, we used wavelet coefficients at each level as features and reduced the dimension of feature space by extracting mean, min, max and standard deviation parameters from them. Figure 3 demonstrates the discrete wavelet transform decomposition process.

#### 3.3 Classification

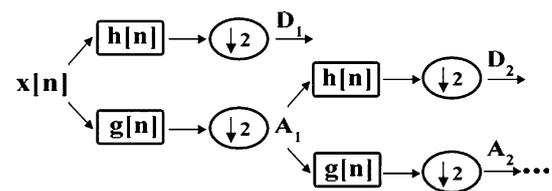
In our experiment, following classifiers were exerted and evaluated:

##### 3.3.1 K nearest neighbor

The nearest neighbor is a classic classifier and is considered as one the simplest of all. K nearest neighbor (KNN) classification is based on finding closest training samples to an unseen point and assigning it to the most dominant class. Even though KNN due to the high dimension of EEG



**Fig. 2** a Shows original signal from C3 channel representing left-hand motor imagery, b shows preprocessed signal ready for feature extraction



**Fig. 3** Decomposition of discrete wavelet transform:  $h[n]$  is the high-pass filter,  $g[n]$  is the low-pass filter,  $A$  is the approximation of the input and  $D$  is the detail

data is not a suitable choice [36], we chose it to increase the diversity among the base classifiers of our ensemble system. Based on empirical results for the dataset, we have come to the conclusion that using 13 nearest neighbors renders to better results for this classifier. To calculate distance between a target sample and other samples in the feature space, euclidean distance measure was used:

$$d(p, q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \tag{1}$$

where  $d(p, q)$  is the distance between the samples  $p$  and  $q$ ,  $p_i$  and  $q_i$  are the  $i$ th feature of the sample and  $n$  is the number of features.

### 3.3.2 Multilayer perceptron

For our experiment, we wanted to evaluate a non-statistical classifier for comparing other classifiers. multilayer perceptron (MLP) fulfills this demand along with simpler implementation comparing to other neural networks [37]. For revising the weights of neurons, we have used back propagation algorithm. First, the weights are set randomly and then the values of hidden and output layers are calculated:

$$O = \frac{1}{1 + e^{-o}} \tag{2}$$

$$Y = \frac{1}{1 + e^{-y}} \tag{3}$$

where  $O$  are the hidden layer neurons and  $Y$  are the output layer neurons.

$$Z' = z(1 - z) \tag{4}$$

where  $z$  is the sigmoid function and  $Z'$  is its derivative. Considering  $w1$  as the value of hidden and input layer weights, and  $w2$  as the value of hidden and output layer weights,  $\Delta W$  for output neurons is calculated:

$$G = y(1 - y)(d - y), \Delta W = (G * O) * \eta \tag{5}$$

where  $d$  is the deterministic output and  $\eta$  is the learning rate. Consequently, the value of  $\Delta W$  for hidden neurons must be obtained:

$$G(o) = O(1 - O)(w2 * G), \Delta W(O) = (xG(o)) * \eta \tag{6}$$

finally weights are updated:

$$w1 = w1 + \Delta W(O), w2 = w2 + \Delta W \tag{7}$$

### 3.3.3 Naive Bayesian

Bayesian classifier is a simple classic probabilistic classifier which is based on Bayes' theorem. In this classifier, each class with highest post-probability will be addressed as the resulting class [38]. The simplicity of this classifier makes it an appropriate candidate for evaluating other classifiers. Power of rejection, meaning the capability of classifier to address an input sample as unpredictable, makes this classifier useful for dealing with uncertain conditions in EEG signal. Another compelling reason for using this classifier is the ability to produce outputs used for soft level combining in the ensemble system [30]. The

naive Bayesian classifier assumes that features are independent in each class and predicts the class of an incoming instance  $X$  containing features  $[x_1.. x_n]$  by calculating the highest probability of  $C_i$  given  $X$  [39]:

$$P(C_i|X) = \frac{P(C_i) \prod_j P(x_j|C_i)}{P(X)} \tag{8}$$

### 3.3.4 Linear discrimination analysis

Linear discrimination analysis (LDA) is a linear classifier, which assumes that the two classes are linearly separable [38]. LDA separates the data using a hyperplane, which is obtained by seeking a projection such that Fisher criterion (i.e., simultaneously, maximizing the distance between centroids of each population while minimizing the inter-population variance) is satisfied [38]. Its drawback is reflected in its linearity, which yields poor results for complex nonlinear data. The within-class scatter matrix  $S_w$  and the between-class scatter matrix  $S_b$  are defined as:

$$S_w = \sum_{k=1}^c \sum_{x \in C_i} (x - \mu_i)(x - \mu_i)^T \tag{9}$$

$$S_b = \sum_{k=1}^c (\mu_i - \mu)(\mu_i - \mu)^T \tag{10}$$

where  $\mu_i$  is the mean of the class  $C_i$ ,  $\mu$  is the mean of all samples and  $c$  denotes the number of classes. Then, we seek a transformation matrix  $W$ , which maximizes the between-class scatter while minimizing the within-class scatter. We achieve that when the Fisher criterion satisfied:

$$w^* = \operatorname{argmax}_w \left\{ \frac{w^T S_b w}{w^T S_w w} \right\} \tag{11}$$

### 3.3.5 Support vector machine

Support vector machine (SVM), another well-known binary linear classifier, also tries to select a hyperplane, with the exception that it improves its discrimination by maximizing the margins (i.e., the distance from the hyperplane to the nearest training samples called support vectors) [40, 41]. Margin maximization will result in increased generalization capability for unseen data points. SVM is a good choice for classification in a high-dimensional space and is known for not being sensitive to the risk of over training [42]. Linear SVMs realize the large margin (i.e., optimal hyperplane) by minimizing the cost function below

$$\frac{1}{2} \|w\|^2 + C \sum_{i=1}^n \xi_i \tag{12}$$

under the constraints

$$y_i(w^T x_i + b) \geq 1 - \xi_i \quad \text{and} \quad \xi_i \geq 0 \quad \forall k = 1, \dots, K \tag{13}$$

where  $\|\cdot\|^2$  denotes the Euclidean norm,  $\xi$  is a vector of slack variables,  $b$  is the bias and  $C$  is a regularization parameter. It is of great importance to select an appropriate value for  $C$ , which controls the trade-off between the complexity and the number of non-separable points.

### 3.4 Classifier ensemble

Several methods exist for creating an ensemble system which we have implemented the followings:

*Bagging:* In the bootstrap aggregating algorithm (aka bagging) given the data containing  $m$  training samples,  $n$  number of subsamples with the same size as the original data is selected (with replacement). An instance may appear more than once in a subsample or may not appear at all [43]. Subsamples are used to train weak learners. Finally, a new instance is classified by winning a vote among constructing classifiers.

*Adaboost:* The Adaboost is a well-known algorithm for improving the accuracy of weak learners [44]. It prognosticates the distribution over the training samples and denotes a weight for each classifier by using a weighted majority vote for predicting labels. Firstly, Adaboost designates a weight distribution to each sample and a subset of samples is chosen by each classifier for learning phase. The initial weights distribution is uniform. Next, a weak classifier creates a hypothesis in order to calculate the error of the classifier (Eq. 14) [29].

$$\varepsilon_t = \sum_{h_t(x_i \neq y_i)} D_t(i) \tag{14}$$

where  $\varepsilon_t$  is the error term of each expert,  $x_i$  is a sample,  $y_i$  is the predicted output and  $D_t(i)$  is the weight of each sample.

For each classifier, there is a defined weight (Eq. 15) which is used in updating weight distribution for the next expert (Eq. 16).

$$\beta_t = \frac{z_t}{1 - z_t} \tag{15}$$

$$D_{t+1}(i) = \frac{D_t(i)}{z_t} \beta_t \quad h_t(x_i) = y_i \tag{16}$$

where  $z_t$  is a normalized term and is equal to the sum of weights distribution,  $\beta_t$  is the weight for an expert,  $D_{t+1}(i)$  is the weight of sample for next expert and  $h_t$  is a hypothesis. For predicting a test sample, all experts weighted votes are received for each class and the class that receives the highest vote in a voting process is considered as the final decision (Eq. 17).

$$v_j = \sum_{h_t(x)=w_j} \log \frac{1}{\beta_t}, j = 1, 2, \dots, C \tag{17}$$

where  $v_j$  is the result of experts supporting class  $j$  and  $W_j$  represents class  $j$ .

*Behavioral knowledge space (BKS):* BKS proposed by Haung and Suen uses the knowledge based on the behavior of classifiers. It is a table concluding  $k$  rows, each representing a classifier decision,  $t$  number of classes and number of arrangements of classifiers decisions (knowledge space) is equal to  $t^k$  [45]. In the training phase, BKS algorithm constructs a knowledge space with different arrangements of classifiers decisions and counts number of samples belonging to each class, the class with the most occurring samples is marked as the predicted label. In the test phase, using the decisions made by classifiers, a column is indexed in the knowledge space, which its label is considered as the predicted label.

*Majority voting:* Plurality majority voting technique gathers opinions of each classifier and checks which class label is most reported by classifiers and choose that label as the final decision for incoming test sample. Using the notion from [29], assume the opinion of an individual classifier as  $d_{t,j} \in \{0, 1\}$  which depicts support for class  $\omega_j$ ,  $t = 1 \dots T$  and  $j = 1 \dots C$ , where  $T$  is the number of classifiers and  $C$  is the number of classes. The formulation for choosing class  $\omega_j$  as the final decision would be expressed as Eq. 18.

$$\sum_{j=1}^T d_{t,j} = \max_{j=1}^C \sum_{i=1}^T d_{t,j} \tag{18}$$

*Weighted majority voting:* Knowing that some classifiers perform better than others, their decision could be weighted and have more influence than other classifiers. This approach may further improve the performance obtained by plurality voting. Finding a weight for a classifier can be accomplished via several measures. Using genetic algorithms and the performance of the classifier as the fitness function, we estimated weights for the classifiers in the ensemble system. Assuming  $W_t$  is the weight of the classifier, the formulation for choosing class  $j$  as the final decision via weighted majority would be expressed as Eq. 19.

$$\sum_{j=1}^T w_t d_{t,j} = \max_{j=1}^C \sum_{i=1}^T w_t d_{t,j} \tag{19}$$

*Combining continuous outputs:* Classifiers are capable of reporting continuous outputs, which demonstrates their tendency toward a specific class. In our study, we applied several non-trainable algebraic combiners. Consider the notion in Eq. 20 which is taken from [46].

$$\mu_j(x) = \xi[d_{1,j}(x), \dots, d_{T,j}(x)] \quad (20)$$

where each element of the vector holds a continuous value representing the tendency of sample  $j$  to a class. Then, using  $\xi$  which is one of the following functions,  $\mu_j(x)$  for each class is calculated and the class with the largest value is considered as the winner. Mean Rule: Using this rule, we calculated the average of all classifiers continuous output which shows support for  $\omega_j$ :

$$\mu_j(x) = \frac{1}{T} \sum_{i=1}^T w_i d_{i,j} \quad (21)$$

where  $1/T$  is the normalization factor Min/Max/Median Rule: As the names of these rules imply, we also selected the minimum, maximum or median of classifiers continuous output as functions to choose the largest value as the winner class.

$$\begin{aligned} \mu_j(x) &= \min_{t=1..T} \{d_{t,j}(x)\} \\ \mu_j(x) &= \max_{t=1..T} \{d_{t,j}(x)\} \\ \mu_j(x) &= \text{median}\{d_{t,j}(x)\} \end{aligned} \quad (22)$$

*Decision template:* Decision template is an approach initially introduced by Kuncheva [31] for combining continuous outputs of an ensemble system. It works based on the principle of decision profiles. Decision profile is a matrix, which its rows represent classes and its columns represent classifiers soft labels. The average of decision profile for each class is equal to decision template of the class in the training set:

$$DT_j = \frac{1}{N_j} \sum_{x_j \in W_j} DP(x_j) \quad (23)$$

where  $N_j$  is the number of all samples which belong to class  $j$ . Considering  $X_j$  as a training sample,  $W_j$  is a class,  $C$  as the number of classes,  $T$  as the number of experts and  $m_j(x)$  as distance between the decision profile of the test sample and decision template of class  $j$ , the Euclidean distance is calculated between decision template for each class and test sample decision profile (Eq. 23), thus each class having the minimum distance is the predicted label [29].

$$m_j(x) = \frac{1}{T * C} \sum_{t=1}^T \sum_{k=1}^C (DP_{t,k}(x_j) - DT(t,k))^2 \quad (24)$$

*Genetic ensemble feature selection:* Using genetic algorithms, which is an evolutionary optimization technique, has been proven to be an effective way for finding the optimal feature subsets [47]. Using GA for

ensemble feature selection was first proposed by [48] by using the accuracy of base classifiers as the fitness function. Various combinations of features, generated in each generation, are represented by binary strings (i.e., each bit denotes the absence or presence of each feature). Until a certain stop criterion is met, generation process is repeated by producing offspring chromosomes from previous population parents and evaluating them by the fitness function to find the suboptimal solution. Using genetic algorithm to select optimal features for an individual classifier also yields diverse decisions.

## 4 Experimental results

Table 1 demonstrates a summary of classification error obtained by groups which attended in BCI competition II and utilized dataset III. Each row denotes classification error of a certain group. Table 2 displays recognition error of the classification systems we implemented. Using genetic algorithm for feature selection and weighted majority voting as the classification approach yields the best recognition rate.

For choosing the optimum number of neighbors for KNN classifier, we evaluated several numbers which is demonstrated in Fig. 4.

Additionally, after several experiments, we found optimum number of hidden layer neurons for MLP which yields better recognition rate. Figure 5 demonstrates recognition rate for different number of hidden layers. Furthermore, as mentioned earlier, using C3 and C4 channels for feature extraction evolves the recognition rate. Table 3 demonstrates different permutations of channel selection and their recognition rates using different classifiers.

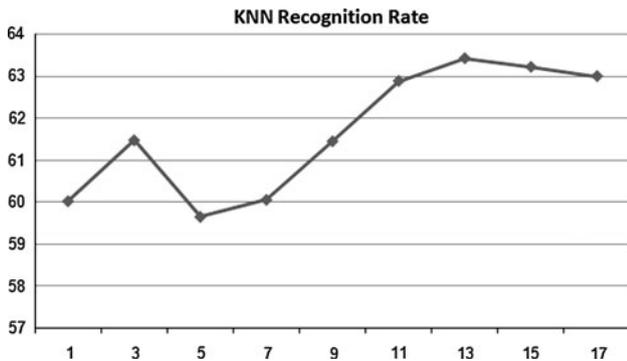
To show the classification results and compare single classifiers to proposed method, we used a confusion matrix (Table 4). A confusion matrix is a square matrix, which includes information about actual and predicted labels designated by a classification system.

**Table 1** BCI competition II, dataset III results

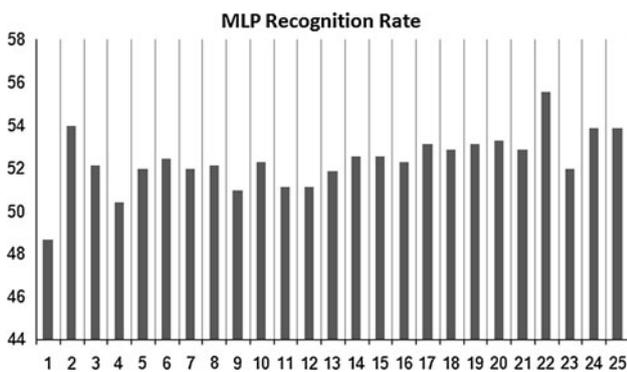
Group	Error (%)
C	10.71
F	15.71
B	17.14
A	13.57
G	17.14
I	23.57
E	17.14
D	32.14
H	49.29

**Table 2** Obtained results from classification experiments. Using weighted majority voting yields the best results

Classification method	Error (%)
KNN	15.72
Naive Bayesian	31.25
MLP	26
LDA	12.14
SVM	11.43
Mean	26.36
Max	30.43
Median	15.65
Adaboost	10.86
BKS	10.44
Bagging	11.15
Decision template	11.43
Majority voting	12.65
Weighted majority voting	<b>10</b>



**Fig. 4** Various number of nearest neighbors were evaluated, 13 number of nearest neighbors renders to the best results



**Fig. 5** Using 22 hidden layer neurons renders to an improved performance

**Table 3** Recognition rate of various channel(s) selection

Selected channel(s)	KNN (%)	SVM (%)	LDA (%)	Average (%)
Cz	48.57	51.42	55.71	51.9
C3	47.14	60	62.86	56.66
C4	52.85	52.85	51.43	52.37
C3,C4	58.57	61.42	60	59.99
C3,CZ	50	64.28	60	58.09
C4,CZ	51.42	64.28	62.86	59.52
C3,C4,CZ	61.42	58.57	54.29	58.09

**Table 4** Confusion matrix of the base classifiers and the proposed multiple classifier system

	Left-hand motor imagery	Right-hand motor imagery
<b>KNN</b>		
Left-hand motor imagery	59	11
Right-hand motor imagery	11	59
<b>Bayesian</b>		
Left-hand motor imagery	45	25
Right-hand motor imagery	20	50
<b>MLP</b>		
Left-hand motor imagery	48	22
Right-hand motor imagery	14	56
<b>LDA</b>		
Left-hand motor imagery	60	10
Right-hand motor imagery	8	62
<b>SVM</b>		
Left-hand motor imagery	59	11
Right-hand motor imagery	5	65
<b>Weighted majority voting</b>		
Left-hand motor imagery	62	8
Right-hand motor imagery	6	64

**5 Conclusion**

The contribution presents several approaches for classification of EEG signal based on ERS/ERD phenomena. We reduced the dimensions of the recorded data and extracted features using Wavelet transform. Based on the fact that EEG signals are vulnerable to noise, there may exist noisy and useless features in the feature vector, and therefore using a feature selection method such as genetic algorithm is salutary in finding the optimum feature space. Ultimately, weaker classifiers when joint can create an

ensemble system which increases precision when classifying motor imageries arising from the brain.

**Acknowledgments** The authors would like to thank the Graz University of Technology, Institute for Biomedical Engineering for providing the data. This work was supported by Shahid Rajaei Teacher Training University.

## References

- Lebedev MA, Nicolelis MAL (2006) Brain-machine interfaces: past, present and future. *TRENDS Neurosci* 29(9):536–546
- Wolpaw JR, Birbaumer N, McFarland DJ, Pfurtscheller G, Vaughan TM (2002) Brain-computer interfaces for communication and control. *Clin Neurophysiol* 113(6):767–791
- Bell CJ, Shenoy P, Chalodhorn R, Rao RP (2008) Control of a humanoid robot by a noninvasive Brain-Computer Interface in humans. *Neural Eng* 5:214–220
- Bayliss JD, Ballard DH (2000) A virtual reality testbed for Brain-Computer Interface research. *Rehabil Eng* 8(2):188–190
- Galán F, Nuttin M, Lew E, Ferrez PW, Vanacker G, Philips J, Millán Jdel RA (2008) Brain-actuated wheelchair: asynchronous and non-invasive brain-computer interfaces for continuous control of robots. *Clin Neurophysiol* 119:2159–2169
- Hashimoto I Y, Ushiba J, Kimura A, Liu M, Tomita Y (2010) change in brain activity through virtual reality-based brain-machine communication in a chronic tetraplegic subject with muscular dystrophy. *BMC Neurosci* 11(117). doi:10.1186/1471-2202-11-117
- Pfurtscheller G, Scherer R (2010) Brain-computer interfaces used for virtual reality control. In: ICABB-Venice
- Reuderink B, Poel M (2008) Robustness of the common spatial patterns algorithm in the BCI-pipeline. Technical report, HMI, University of Twente. Available at <http://doc.utwente.nl/64884/>
- Niedermeyer E, Da Silva F (2005) Electroencephalography: basic principles, clinical applications, and related fields. Doody's all reviewed collection. Lippincott Williams & Wilkins, Philadelphia, PA
- Millán JdR, Rupp R, Müller-Putz GR, Murray-Smith R, Giugliemma C, Tangermann M, Vidaurre C, Cincotti F, Kübler A, Leeb R, Neuper C, Müller K-R, Mattia D (2010) Combining brain-computer interfaces and assistive technologies: state-of-the-art and challenges. *Frontiers Neurosci* 4(9):161
- Keng Ang K, Guan C, Geok Chua S, Ti Ang B, Kuah C, Wang C, Soon Phua C, Yang Chin Z, Zhang H (2009) A clinical study of motor imagery-based brain-computer interface for upper limb robotic rehabilitation. In: International conference of the IEEE engineering in medicine and biology society, Minneapolis, MN
- Wang Y, Jao S (2005) Common spatial pattern method for channel selection in motor imagery based brain computer interface. *IEEE* 1(1):5392–5395
- Leuthardt EC, Schalk G, Wolpaw JR, Ojemann JG, Moran DW (2004) A brain-computer interface using electrocorticographic signals in humans. *J Neural Eng* 1:63–71
- McFarland TMVDJ, Miner LA, Wolpaw JR (2000) Mu and beta rhythm topographies during motor imagery and actual movements. *Brain Topogr* 12(3):177–186
- Pfurtscheller G, Neuper C, Guger C, Harkam W, Ramoser H, Schlögl A, Obermaier B, Pregenzer M (2000) Current trends in graz brain-computer interface (BCI) research. *Rehabil Eng* 8: 216–225
- Pfurtscheller G, Neuper C (1997) Motor imagery activates primary sensorimotor areas. *Neurosci Lett* 239:65–68
- Pfurtscheller G, Neuper C, Flotzinger D, Pregenzer M (1997) EEG-based discrimination between imagination of right and left hand movement. *Electroenc Clin Neurophys* 103(5):1–10
- Hoffmann U, Vesin J-M, Ebrahimi T, Diserens K (2008) An efficient p300-based brain-computer interface for disabled subjects. *J Neurosci Methods* 167:115–125
- Neupera C, Müller G, Kubler A, Birbaumer N, Pfurtscheller G (2003) Clinical application of an EEG-based brain-computer interface: a case study in a patient with severe motor impairment. *Clin Neurophysiol* 114:399–409
- Obermaier B, Neuper C, Guger C, Pfurtscheller G (2001) Information transfer rate in a five-classes brain-computer interface. *Neural Syst Rehabil Eng* 9(9):283–288
- Zhang H, Guan C, Wang C (2008) Asynchronous p300-based brain-computer interfaces: a computational approach with statistical models. *Biomed Eng* 55(6):1754–1763
- Adelia H, Zhou Z, Dadmehr N (2003) Analysis of EEG records in an epileptic patient using wavelet transform. *Neurosci Methods* 123(2):69–87
- Cvetkovic D, Derya beyli E, Cosic I (2008) Wavelet transform feature extraction from human ppg, ECG, and EEG signal responses to elf pemf exposures: a pilot study. *Elsevier Digit Signal Process* 18:861–874
- Übeyli ED (2009) Combined neural network model employing wavelet coefficients for EEG signals classification. *Elsevier Digit Signal Process* 19:297–308
- Subasi A (2007) EEG signal classification using wavelet feature extraction and a mixture of expert model. *Elsevier Expert Syst Appl* 32:1084–1093
- <http://www.bbc.de/competition/ii/results/> (2003) Results webpage for BCI competition II
- Miller-Gerking HRJ (2000) Optimal spatial filtering of single trial EEG during imagined hand movement. *IEEE Trans Rehabil Eng* 1(1):441–446
- Javadi M, Ebrahimpour R, Sajedin A, Faridi S, Zakernejad S (2011) Improving ECG classification accuracy using an ensemble of neural network modules. *Plos One* 6(10):e24386. doi:10.1371/journal.pone.0024386
- Polikar R (2006) Ensemble based systems in decision making. *IEEE Circuits Syst Mag* 6(3):21–45
- Ghaderi R (2000) Arranging simple neural networks to solve complex classification problems. University of Surrey, Guildford
- Kuncheva LI (2004) Combining pattern classifiers: methods and algorithms. Wiley-Interscience, New York
- Esmaili M (2007) Classifiers fusion for EEG signals processing in human-computer interface systems. In: Proceedings of the 22nd national conference on Artificial intelligence, vol 2. AAAI Press, Menlo Park, CA, pp. 1856–1857
- Ebrahimpour R, Esmkhani A, Faridi S (2010) Farsi handwritten digit recognition based on mixture of rbf experts. *IEICE Electron Express* 7(14):1014–1019
- Ebrahimpour R, Kabir E, Yousefi MR (2008) Teacher-directed learning in view-independent face recognition with mixture of experts using single-view eigenspaces. *J Franklin Inst* 345(2): 87–101
- <http://www.bbc.de/competition/ii/> (2003) Dataset III provided by Institute for Biomedical Engineering, Graz University of Technology
- Friedman JH, Fayyad U (1997) On bias, variance, 0/1-loss, and the curse-of-dimensionality. *Data Min Knowl Disc* 1:55–77
- Haykin S (1998) Neural networks. 2nd edn. Prentice Hall, Englewood Cliffs NJ
- Richard DGS, Duda O, Hart PE (2000) Pattern classification, 2nd edn. Wiley, New York
- Zheng Z (1998) Naive bayesian classifier committees. In: Ndellec C, Rouveilol C (eds) Machine learning: ECML-98, vol. 1398 of

- lecture notes in computer science. Springer, Heidelberg, pp 196–207. doi:[10.1007/BFb0026690](https://doi.org/10.1007/BFb0026690)
40. Gunn S (1998) Support vector machines for classification and regression,” technical report, University of Southampton, Faculty of Engineering, Science and Mathematics School of Electronics and Computer Science
  41. Burges CJC (1998) A tutorial on support vector machines for pattern recognition. *Data Min Knowl Disc* 2:121–167
  42. Jain RPWJMAK (2000) Statistical pattern recognition: a review. *IEEE Trans Pattern Anal Machine Intell* 22(1):4–37
  43. Quinlan JR (1996) Bagging, boosting, and c4.5. *AAAI/IAAI* 1: 725–730
  44. Freund Y, Schapire RE (1997) A decision-theoretic generalization of on-line learning and an application to boosting. *J Comput Syst Sci* 55(1):119–139
  45. Huang YS, Suen CY (1993) The behavior-knowledge space method for combination of multiple classifiers. *Comput Vision Pattern Recogn* 6:347–352
  46. Kuncheva L, Bezdek J, Duin R (2001) Decision templates for multiple classifier fusion: an experimental comparison. *Pattern Recogn* 34(2):299–314
  47. Tsymbal PCA, Pechenizkiy M (2005) Sequential genetic search for ensemble feature selection. In: *IJCAI05*, 19th international joint conference on artificial intelligence, pp 877–882
  48. Kuncheva L (1993) Genetic algorithm for feature selection for parallel classifiers. *Inf Process Lett* 46:163–168