Performance Evaluation of Five Classification Algorithms in Low-Dimensional Feature Vectors Extracted From EEG Signals

Onder Aydemir, Mehmet Ozturk and Temel Kayikcioglu

***Abstract*—There are lots of classification and feature extraction algorithms in the field of brain computer interface. It is significant to use optimal classification algorithm and fewer features to implement a fast and accurate brain computer interface system. In this paper, we evaluate the performances of five classical classifiers in different aspects including classification accuracy, sensitivity, specificity, Kappa and computational time in low-dimensional feature vectors extracted from EEG signals. The experiments show that naïve Bayes is the most appropriate classifier for low dimensional feature vectors compared to k-nearest neighbor, support vector machine, linear discriminant analysis and decision tree classifiers.**

***Keywords*— Brain computer interface, classification accuracy, classification performance, computational time, Kappa, low-dimensional feature vector, sensitivity, specificity.**

I. Introduction

THE brain computer interface (BCI) provides a new communication channel for subjects to interact the external world without using their muscles. Electroencephalogram (EEG) based BCI systems analyzed electrical brain activity recorded from electrodes placed the subject's scalp and extract features to determine the intents of the user. Then, translate them into the control signals that are used to control external devices (e.g., an electromechanical arm, a wheelchair) [1], [2].

Input signals of a EEG based BCI system are naturally non-stationary, have poor signal to noise ratio, dependent on physical or mental tasks, contaminated with various artifacts, such as electromyogram (EMG) and electrooculogram (EOG). All these disadvantages motivate the researchers to improve substantially the speed and accuracy of all components of the communication system between the brain and a BCI output device. So, it is significant to use optimal classification algorithm and fewer features to implement a fast and accurate brain computer interface system [3], [4], [5].

In literature, several classification algorithms have been used for specific features such as linear discriminate analysis (LDA) [6], [7], k-nearest neighbor (k-NN) [3], [8], support vector machines (SVM) [9], [10], hidden Markov classifier [11], neural networks [12], [13]. Most of them have evaluated the performance of classifier just in terms of classification accuracy (CA). On the other hand, feature vector dimension influences classifiers performance [14]. So, in order to propose the most appropriate classifier, it is essential to predict properties of features, such as whether it is low or high dimension. There are also some studies in literature which compare the performance of different classifiers [15]-[17]. But none of them have investigated classifiers in terms of low-dimensionality. In this paper, based on the theory of optimal classifier and fewer features, we evaluate the performances of five classical classifiers in different aspects including CA, sensitivity (SE), specificity (SP), Kappa (κ) and computational time (CT) in low-dimensional feature vectors extracted from EEG signals.

For our study, we used BCI Competition 2003 Data Set III which is described in the following subsection. We extracted six features by calculating alpha frequency (8-13 Hz), beta frequency (13-20 Hz) and total frequency band powers of the signals. Then, we classified the signals with five classifiers including k-NN, SVM, LDA, naïve Bayes (NB) and decision tree (DT).

The paper is organized as follows: Section II describes the materials and methods. The results are provided in Section III. The conclusions and discussions are given in Section IV. Finally, future work is described in the last section.

II. Materials and Methods

A. Data Set Description

Our algorithm is performed on the BCI Competition 2003 Data Set III, which was taken from a single healthy female subject at the University of Technology Graz.

Brain activity was recorded with three bipolar EEG channels (C3, Cz, C4) with sampling frequency of 128 Hz and it was filtered between 0.5 and 30Hz. The recording length of a trial was set to 9 seconds. The first 2 seconds were quite. At t = 2 seconds, an acoustic stimulus indicated the beginning of the trial, and a cross ("+") was displayed for 1 seconds. Then, at t = 3 seconds, an arrow (left/ right) was displayed as a cue. The subject was asked to use imagination of left or right hand movements to move the feedback bar into the direction of the cue. The order of left and right cues was random. During the experiment the subject sat in a relaxing chair with armrests. We decided to use the last 6 seconds, while the first 3 seconds is the preparation period in which no event happened.

The experimental data set consists of 140 trials for training

Manuscript received May 17, 2011. This work was supported by the TUBITAK (The Scientific and Technological Research Council of Turkey), Project No: EEEAG-110E035.

O. Aydemir is with Karadeniz Technical University, Dep. of Electrical and Electronics Engineering, 61080 Trabzon, Turkey (corresponding author phone: +90-462-377 4015; e-mail: onderaydemir@ktu.edu.tr).

M. Ozturk and T. Kayikcioglu are with Karadeniz Technical University, Dep. of Electrical and Electronics Engineering, 61080 Trabzon, Turkey.

set (70 trials for right hand movement, RHM and 70 trials for left hand movement, LHM) and 140 trials for test set (70 trials for RHM and 70 trials for LHM). For further information about the data set, please refer to [18], [19].

B. *Feature Extraction*

Feature extraction is a crucial step in BCI system, because its capability directly influences the performance of the classifier. However, it requires a lot of research to extract useful features among the existing feature extraction methods or from a newly developed method.

In this study power spectral density (PSD) technique is used as feature extraction method. This technique has always been a popular method for classifying EEG signals [20]-[22]. The first step in EEG classification is to determine if the signals have distinguishable features in their power spectrum. With a close examination, we observed that the alpha, the beta and the total band powers of the signals, recorded from C3 and C4, show difference between left and right hand movement imaginations.

In order to obtain band power (BP) of the signals firstly we calculated the fast Fourier transform (FFT) coefficients as follows:

$$X(k)=\frac{1}{N}\sum_{n=0}^{N-1}x(n)\,e^{-j2\pi(k-1)(n-1)/N},\quad k=0,1,\ldots,N \qquad (1)$$

where N is number of EEG samples taken for analysis, $x(n)$ is the EEG signal, $X(k)$ is the k^{th} FFT coefficient. Then, BP is computed by:

$$BP\Big|_{f_{LOW}}^{f_{UPPER}}=\sum\left\|X(k)\Big|_{f_{LOW}}^{f_{UPPER}}\right\|^2 \qquad (2)$$

where $X(k)\Big|_{f_{LOW}}^{f_{UPPER}}$ denotes FFT coefficients between low cutoff frequency (f_{LOW}) and upper cutoff frequency (f_{UPPER}). For the alpha band $f_{LOW}=8$ *Hz* and $f_{UPPER}=13$ *Hz*, for the beta band $f_{LOW}=13$ *Hz* and $f_{UPPER}=20$ *Hz* and for the total band $f_{LOW}=0.5$ *Hz* and $f_{UPPER}=30$ *Hz* (because EEG data set was filtered between 0.5 and 30Hz when it was recorded).

The calculated band powers of the training set are illustrated in Fig. 1. In this figure, Figure 1(a) shows the alpha BP of Channel C3, which is defined as feature 1 (f1), Figure 1(b) shows the alpha BP of Channel C4, which is defined as feature 2 (f2), Figure 1(c) shows the beta BP of Channel C3, which is defined as feature 3 (f3), Figure 1(d) shows the beta BP of Channel C4, which is defined as feature 4 (f4), Figure 1(e) shows the total BP of Channel C3, which is defined as feature 5 (f5), Figure 1(f) shows the total BP of Channel C4, which is defined as feature 6 (f6). The horizontal axis is the index of trial numbers, and the vertical axis is the values of power. Plus points stand for the trials of RHM, and circle points stand for the trials of LHM. As seen from the figure, the band powers for the RHM and the LHM trials have clustered the opposite directions. Based on this strong clue, we considered these values can be selected as features to classify the two tasks.

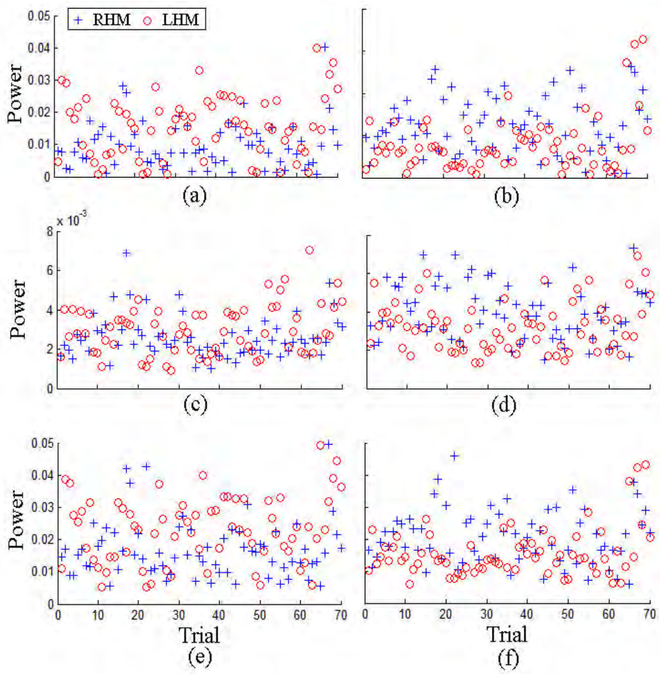

Fig. 1. Band powers, (a) Alpha BP of Channel C3, (b) Alpha BP of Channel C4, (c) Beta BP of Channel C3, (d) Beta BP of Channel C4, (e) Total BP of Channel C3, (f) Total BP of Channel C4.

C. *Classification Algorithms*

A classifier is an algorithm which has to be trained with labeled training examples to be able to distinguish new unlabeled examples between a fixed set of classes. In our study we trained k-NN, SVM, LDA, NB and DT algorithms by using all training data set. In this section, we briefly review aspects of the five classifiers.

1) *k-Nearest Neighbor:* The k-NN classifier is a common classification algorithm, which determines a testing sample's class by the majority class of the *k* closest training samples [23]. Performance of a k-NN algorithm depends on the distance metric and the value of the closest training sample parameter, *k*. In our study, we used Euclidean distance metric and leave-one-out cross-validation (LOOCV) technique to determine *k* value. The most appropriate *k* value was searched in interval between 1 and 15, with step size of 1. Appendix describes in detail the LOOCV technique.

2) *Support Vector Machine:* SVM performs classification tasks by constructing the best hyper plane in a multidimensional space by finding the maximum possible margin [24]. We utilized SVM with a radial basis function kernel. We have chosen this kernel due to the fact that the number of hyper parameters of this kernel is smaller than those of other kernels. This kernel function is specified by the scaling factor σ. To determine optimum value for the scaling factor, the same validation procedure used in the k-NN classification algorithm. The most appropriate σ value was searched in interval between 0.1 and 1.5, with step size of 0.1.

3) *Linear Discriminant Analysis*: LDA classifies two classes based on the assumption that both classes are under normal distribution with equal covariance matrices. The separating hyper plane is obtained by finding the

projection of the labeled training data that maximizes the distance between the two classes' means and minimizes the interclass variance. The main aim is to solve the problem

$$y = w^T x + w_0 \,, \tag{3}$$

where x is the feature vector. The vectors w and w_0 are determined by maximization of the interclass means and minimization of interclass variance [25].

4) *Naïve Bayes:* Naïve Bayes classifier is a simple probabilistic algorithm based on applying Bayes' theorem [26] with naïve independence assumptions. Consider a set of training trials where each trial is made up from m discrete-valued features and a class from a finite set C. The naïve Bayes classifier can probabilistically predict the class of an unknown trial using the available training trial set to calculate the most probable output. The most probable class C_{NB} of an unknown trial with the conjunction $A=a_1, a_2, ..., a_m$ is calculated by

$$C_{NB} = \underset{c \in C}{\arg\max}\; p(c \setminus \mathrm{A}) \,. \tag{4}$$

5) *Decision Tree:* This algorithm constructs a decision tree with branche(s) and node(s) based on training set. Each branch descending from a node corresponds to one of the possible values of the feature specified at that node. And each test results in branches, which represent different outcomes of the test. For a detailed description of the method see [27]. In this paper, to construct the decision tree default configuration of the *classregtree* function in Matlab R2010b was used.

D. Performance Criteria

1) *Classification Accuracy:* Classification accuracy is defined as the percentage of the number of trials classified correctly in the test set over the total trials.

2) *Sensitivity and Specificity:* Sensitivity and specificity are calculated by the following formulas, respectively:

$$sensitivity = \frac{TP}{TP + FN} x100 \tag{5}$$

$$specificity = \frac{TN}{TN + FP} x100 \tag{6}$$

where TP is the number of positive samples correctly predicted, TN is the number of negative samples correctly predicted, FP the number of positive samples incorrectly predicted and FN is the number of negative samples incorrectly predicted. In our study, we defined the RHM imageries as the positive samples and the LHM imageries as the negative samples. So, the sensitivity refers to the ratio of correctly classified RHMs to the total population of RHM cases, whereas specificity is the ratio of correctly classified LHMs to the total population of LHM cases.

3) *Kappa:* Kappa statistics is defined as the proportion of correctly classified samples after accounting for the probability of chance agreement. It is calculated by:

$$Kappa = \frac{P(D) - P(E)}{1 - P(E)} \tag{7}$$

where P(D) denotes the proportion of overall agreement and P(E) is the probability of expected agreement by chance. The Kappa coefficient value is ranged between 1 and −1, which corresponds to a perfect and a completely wrong classification, respectively. A Kappa coefficient with value 0 means that the performance is equal to random guess.

4) *Computational Time:* We computed the training and testing times of the classifiers. All the runtime experiments were conducted on a PC with Intel ® Core ™ i7 CPU 1.73 GHz, 4 GB RAM.

III. Results

The BCI Competition 2003 Data Set III was tested with five classifiers using double, triple, quadruple, quintuple and sextuple combination(s) of the six features. Table I presents results for the classifiers in terms of four metrics including CA, SE, SP and κ. In the Table, the best results of the metrics are written in boldface and averages of the four metrics are given in the last line. In case of using f1, f3, f4 and f6 features together, NB classifier provided the best CA, SP and κ performance which are 82.9%, 88.6% and 0.66, respectively. The best SE was obtained as 88.6% when f2 and f3 feature pair classified by using the *k*-NN algorithm. The worst case was obtained when f1 and f3 feature pair classified by using DT algorithm. In this case CA, SE, SP and κ were calculated as 52.9%, 61.4%, 44.3% and 0.06, respectively. On the other hand, SVM reached its highest CA, SE, SP and κ values as 80%, 80%, 82.9% and 0.60, respectively. LDA classifier reached its highest CA, SE, SP and κ values as 81.4%, 82.9%, 84.3% and 0.63, respectively.

According to the averages of four metrics NB classifier achieves the highest average values of CA, SP and κ which are 76.4%, 77.8% and 0.53, respectively. The highest average value of SE is obtained as 75.7% by k-NN algorithm.

We also computed the computational times of the classifiers for training and testing stages. Table II presents the average computational times of the both stages (the values are given in seconds). It can be seen in the table that the fastest training and testing times were obtained as 0.005 CPU seconds with LDA and NB classifiers. Conversely, the slowest time for the training and testing stages was obtained by SVM.

IV. Conclusion and Discussion

This paper evaluated the performances of five classifiers in different aspects including CA, SE, SP, κ and CT in low-dimensional feature vectors extracted from the BCI Competition 2003 Data Set III. The experiments proved that NB achieves better classification performance. However, DT provides the worst performance according to the average values of the CA, SE, SP and κ.

The experiments showed that if a classifier has any tune parameter it causes time consuming especially in training phase. Compared to the other classifiers, SVM takes much more time to be trained. The testing times of the k-NN and the SVM classification algorithms are 4 and 8 times longer than those of the LDA and the NB algorithms, respectively.

TABLE I
PERFORMANCES OF THE CLASSIFIERS

Features	k-NN				SVM				LDA				NB				DT			
	CA	**SE**	**SP**	**κ**	**CA**	**SE**	**SP**	**κ**	**CA**	**SE**	**SP**	**κ**	**CA**	**SE**	**SP**	**κ**	**CA**	**SE**	**SP**	**κ**
f1-f2	78.6	75.7	81.4	0.57	78.6	77.1	80.0	0.57	77.9	77.1	78.6	0.56	78.6	78.6	78.6	0.57	78.6	78.6	78.6	0.57
f1-f3	65.0	70.0	60.0	0.30	65.0	67.1	62.9	0.30	65.7	61.4	70.0	0.31	65.7	65.7	65.7	0.31	52.9	61.4	44.3	0.06
f1-f4	69.3	70.0	68.6	0.39	75.7	72.9	78.6	0.51	80.0	78.6	81.4	0.60	76.4	78.6	74.3	0.53	62.9	67.1	58.6	0.26
f1-f5	63.6	67.1	60.0	0.27	65.0	68.6	61.4	0.30	65.0	62.9	67.1	0.30	65.7	62.9	68.6	0.31	65.0	62.9	67.1	0.30
f1-f6	78.6	74.3	82.9	0.57	79.3	75.7	82.9	0.59	78.6	74.3	82.9	0.57	79.3	80.0	78.6	0.59	75.0	78.6	71.4	0.50
f2-f3	73.6	**88.6**	58.6	0.47	72.1	80.0	64.3	0.44	79.3	81.4	77.1	0.59	75.7	74.3	77.1	0.51	65.0	82.9	47.1	0.30
f2-f4	63.6	77.1	50.0	0.27	67.9	72.9	62.9	0.36	71.4	70.0	72.9	0.43	69.3	68.6	70.0	0.39	60.0	72.9	47.1	0.20
f2-f5	79.3	75.7	82.9	0.59	77.1	77.1	77.1	0.54	76.4	74.3	78.6	0.53	77.1	75.7	78.6	0.54	71.4	61.4	81.4	0.43
f2-f6	65.0	78.6	51.4	0.30	70.7	74.3	67.1	0.41	71.4	72.9	70.0	0.43	72.1	70.0	74.3	0.44	54.3	62.9	45.7	0.09
f3-f4	67.9	77.1	58.6	0.36	66.4	74.3	58.6	0.33	70.0	70.0	70.0	0.40	70.0	65.7	74.3	0.40	65.7	77.1	54.3	0.31
f3-f5	63.6	64.3	62.9	0.27	63.6	60.0	67.1	0.27	63.6	57.1	70.0	0.27	65.7	67.1	64.3	0.31	56.4	55.7	57.1	0.13
f3-f6	70.7	82.9	58.6	0.41	70.7	74.3	67.1	0.41	75.0	74.3	75.7	0.50	76.4	75.7	77.1	0.53	65.7	75.7	55.7	0.31
f4-f5	69.3	67.1	71.4	0.39	75.7	72.9	78.6	0.51	76.4	75.7	77.1	0.53	74.3	75.7	72.9	0.49	67.1	67.1	67.1	0.34
f4-f6	69.3	77.1	61.4	0.39	66.4	68.6	64.3	0.33	69.3	70.0	68.6	0.39	70.0	70.0	70.0	0.40	53.6	58.6	48.6	0.07
f5-f6	77.9	74.3	81.4	0.56	76.4	74.3	78.6	0.53	78.6	77.1	80.0	0.57	77.9	77.1	78.6	0.56	77.1	82.9	71.4	0.54
f1-f2-f3	79.3	78.6	80.0	0.59	79.3	80.0	78.6	0.59	77.9	77.1	78.6	0.56	81.4	82.9	80.0	0.63	75.7	80.0	71.4	0.51
f1-f2-f4	78.6	77.1	80.0	0.57	79.3	78.6	80.0	0.59	77.9	77.1	78.6	0.56	80.7	75.7	85.7	0.61	78.6	80.0	77.1	0.57
f1-f2-f5	77.9	74.3	81.4	0.56	79.3	78.6	80.0	0.59	77.9	77.1	78.6	0.56	77.1	78.6	75.7	0.54	77.1	74.3	80.0	0.54
f1-f2-f6	78.6	72.9	84.3	0.57	80.0	78.6	81.4	0.60	80.7	80.0	81.4	0.61	80.7	78.6	82.9	0.61	80.0	78.6	81.4	0.60
f1-f3-f4	70.0	70.0	70.0	0.40	72.9	74.3	71.4	0.46	77.9	78.6	77.1	0.56	75.7	78.6	72.9	0.51	65.7	70.0	61.4	0.31
f1-f3-f5	65.0	67.1	62.9	0.30	66.4	65.7	67.1	0.33	65.0	61.4	68.6	0.30	65.7	64.3	67.1	0.31	55.7	58.6	52.9	0.11
f1-f3-f6	78.6	75.7	81.4	0.57	78.6	80.0	77.1	0.57	79.3	74.3	84.3	0.59	80.0	80.0	80.0	0.60	77.1	77.1	77.1	0.54
f1-f4-f5	69.3	72.9	65.7	0.39	75.0	72.9	77.1	0.50	76.4	75.7	77.1	0.53	74.3	72.9	75.7	0.49	70.0	80.0	60.0	0.40
f1-f4-f6	79.3	75.7	82.9	0.59	79.3	78.6	80.0	0.59	78.6	74.3	82.9	0.57	80.7	75.7	85.7	0.61	75.0	78.6	71.4	0.50
f1-f5-f6	77.9	72.9	82.9	0.56	77.1	74.3	80.0	0.54	78.6	77.1	80.0	0.57	75.7	77.1	74.3	0.51	78.6	82.9	74.3	0.57
f2-f3-f4	72.1	85.7	58.6	0.44	72.1	78.6	65.7	0.44	75.0	75.7	74.3	0.50	71.4	68.6	74.3	0.43	62.9	77.1	48.6	0.26
f2-f3-f5	79.3	75.7	82.9	0.59	75.0	74.3	75.7	0.50	75.7	72.9	78.6	0.51	77.1	78.6	75.7	0.54	72.9	67.1	78.6	0.46
f2-f3-f6	71.4	85.7	57.1	0.43	75.0	80.0	70.0	0.50	78.6	78.6	78.6	0.57	73.6	72.9	74.3	0.47	65.0	77.1	52.9	0.30
f2-f4-f5	78.6	75.7	81.4	0.57	80.0	80.0	80.0	0.60	77.9	74.3	81.4	0.56	78.6	72.9	84.3	0.57	71.4	62.9	80.0	0.43
f2-f4-f6	69.3	84.3	54.3	0.39	67.9	75.7	60.0	0.36	67.9	70.0	65.7	0.36	70.0	68.6	71.4	0.40	60.0	71.4	48.6	0.20
f2-f5-f6	80.7	77.1	84.3	0.61	80.0	78.6	81.4	0.60	77.9	75.7	80.0	0.56	79.3	77.1	81.4	0.59	72.1	65.7	78.6	0.44
f3-f4-f5	69.3	70.0	68.6	0.39	72.1	72.9	71.4	0.44	75.7	75.7	75.7	0.51	73.6	74.3	72.9	0.47	65.7	71.4	60.0	0.31
f3-f4-f6	70.0	82.9	57.1	0.40	70.7	77.1	64.3	0.41	73.6	74.3	72.9	0.47	73.6	70.0	77.1	0.47	65.7	75.7	55.7	0.31
f3-f5-f6	78.6	75.7	81.4	0.57	77.9	77.1	78.6	0.56	78.6	77.1	80.0	0.57	80.0	82.9	77.1	0.60	74.3	72.9	75.7	0.49
f4-f5-f6	78.6	74.3	82.9	0.57	79.3	78.6	80.0	0.59	77.9	77.1	78.6	0.56	78.6	72.9	84.3	0.57	77.9	81.4	74.3	0.56
f1-f2-f3-f4	80.0	78.6	81.4	0.60	79.3	78.6	80.0	0.59	77.9	77.1	78.6	0.56	81.4	77.1	85.7	0.63	76.4	81.4	71.4	0.53
f1-f2-f3-f5	77.9	74.3	81.4	0.50	78.6	77.1	80.0	0.57	78.6	78.6	78.6	0.57	75.6	77.1	74.3	0.51	72.9	74.3	71.4	0.46
f1-f2-f3-f6	80.0	77.1	82.9	0.60	78.6	80.0	77.1	0.57	78.6	77.1	80.0	0.57	81.4	77.1	85.7	0.63	75.7	78.6	72.9	0.51
f1-f2-f4-f5	80.0	74.3	85.7	0.60	79.3	78.6	80.0	0.59	77.1	75.7	78.6	0.54	79.3	80.0	78.6	0.59	76.4	75.7	77.1	0.53
f1-f2-f4-f6	77.9	75.7	80.0	0.56	77.9	77.1	78.6	0.56	77.1	75.7	78.6	0.54	77.9	74.3	81.4	0.56	79.3	80.0	78.6	0.59
f1-f2-f5-f6	77.9	74.3	81.4	0.56	79.3	78.6	80.0	0.59	80.7	80.0	81.4	0.61	77.9	77.1	78.6	0.56	78.6	74.3	82.9	0.57
f1-f3-f4-f5	67.9	71.4	64.3	0.36	71.4	74.3	68.6	0.43	77.1	77.1	77.1	0.54	71.4	72.9	70.0	0.43	67.1	65.7	68.6	0.34
f1-f3-f4-f6	79.3	77.1	81.4	0.59	77.1	77.1	77.1	0.54	79.3	74.3	84.3	0.59	**82.9**	77.1	**88.6**	**0.66**	77.1	77.1	77.1	0.54
f1-f3-f5-f6	78.6	74.3	82.9	0.57	78.6	80.0	77.1	0.57	78.6	77.1	80.0	0.57	73.6	74.3	72.9	0.47	76.4	77.1	75.7	0.53
f1-f4-f5-f6	77.1	72.9	81.4	0.54	77.1	74.3	80.0	0.54	77.9	77.1	78.6	0.56	77.9	78.6	77.1	0.56	78.6	82.9	74.3	0.57
f2-f3-f4-f5	80.0	77.1	82.9	0.60	77.9	78.6	77.1	0.56	75.7	72.9	78.6	0.51	80.0	75.7	84.3	0.60	72.1	62.9	81.4	0.44
f2-f3-f4-f6	70.7	81.4	60.0	0.41	72.9	78.6	67.1	0.46	72.1	75.7	68.6	0.44	74.3	71.4	77.1	0.49	63.6	77.1	50.0	0.27
f2-f3-f5-f6	80.7	77.1	84.3	0.61	77.1	77.1	77.1	0.54	77.1	74.3	80.0	0.54	82.1	78.6	85.7	0.64	70.7	61.4	80.0	0.41
f2-f4-f5-f6	79.3	75.7	82.9	0.59	77.9	77.1	78.6	0.56	78.6	75.7	81.4	0.57	75.7	72.9	78.6	0.51	72.1	65.7	78.6	0.44
f3-f4-f5-f6	78.6	75.7	81.4	0.57	78.6	78.6	78.6	0.57	77.1	77.1	77.1	0.54	82.1	77.1	87.1	0.64	72.9	72.9	72.9	0.46
f1-f2-f3-f4-f5	78.6	74.3	82.9	0.57	78.6	80.0	77.1	0.57	78.6	78.6	78.6	0.57	79.3	81.4	77.1	0.59	73.6	75.7	71.4	0.47
f1-f3-f4-f5-f6	77.9	72.9	82.9	0.56	78.6	77.1	80.0	0.57	77.9	77.1	78.6	0.56	78.6	80.0	77.1	0.57	77.1	77.1	77.1	0.54
f1-f2-f4-f5-f6	78.6	75.7	81.4	0.57	78.6	77.1	80.0	0.57	80.7	82.9	78.6	0.61	81.4	77.1	85.7	0.63	77.9	75.7	80.0	0.56
f1-f2-f3-f5-f6	77.1	75.7	78.6	0.54	77.9	77.1	78.6	0.56	81.4	80.0	82.9	0.63	80.7	81.4	80.0	0.61	72.9	71.4	74.2	0.46
f1-f2-f3-f4-f6	78.6	75.7	81.4	0.57	78.6	80.0	77.1	0.57	77.1	75.7	78.6	0.54	80.0	75.7	84.3	0.60	76.4	80.0	72.9	0.53
f2-f3-f4-f5-f6	77.9	75.7	80.0	0.56	79.3	78.6	80.0	0.59	77.9	74.3	81.4	0.56	79.3	75.7	82.9	0.59	70.7	61.4	80.0	0.41
f1-f2-f3-f4-f5-f6	76.4	75.7	77.1	0.53	77.9	78.6	77.1	0.56	81.4	82.9	80.0	0.63	79.3	75.7	82.9	0.59	73.6	72.9	74.3	0.47
Averages	74.9	75.6	74.2	0.50	75.4	74.7	74.7	0.51	76.2	75.1	77.4	0.53	76.4	75.0	77.8	0.53	70.6	72.8	68.4	0.41

TABLE II
AVERAGE COMPUTATIONAL TIMES OF THE CLASSIFIERS

Stage	k-NN	SVM	LDA	NB	DT
Training	5	27	0.005	0.005	0.04
Testing	0.02	0.04	0.005	0.005	0.02

We believe that this paper will provide a significant contribution in the field of classifier for BCI.

V. FUTURE WORK

Future work can proceed in a number of directions:

1) *More EEG data:* We are aware that results presented in this paper are somewhat limited. Therefore, we plan to experiment classification methods on a different EEG dataset.

2) *Improve classification performance:* We plan to further improve motor imagery EEG classification performance by using other feature extraction technique(s).

3) *More classifiers:* In the future we can test more classification algorithms such as multilayer perceptron and hidden Markov classifier.

All these future works can help provide more insights to the researchers to determine which approach to use.

APPENDIX

LEAVE-ONE-OUT CROSS-VALIDATION

In LOOCV technique, the training phase is performed using T-1 trials, where T is the total number of trials, and the validation is carried out using the excluded trial. If this trial is misclassified an error is counted. This is repeated T times, each time excluding a different trial. On the other hand, because we had 140 trials, for 140 times, 1 of the trial-data was chosen as the validation trial and the classifier was trained by the remaining 139 trials, and then the trained classifier was applied on the 1 validation data.

We used LOOCV technique to estimate the most appropriate classifier parameter, which provides the highest average value of the four metrics (CA, SE, SP and κ) result on validation set. Average of the four metrics was computed as follows:

$$Performance_{Avg} = \frac{CA + SE + SP + \text{Kappa x } 100}{4}. \quad (8)$$

We utilized LOOCV, since it makes the best use of the available data and avoid the problems of random selections.

REFERENCES

[1] T.M. Vaughan, "Brain–computer interface technology: A review of the second international meeting", *IEEE T. Neur. Sys. Reh.*, 11: 94–109, 2003.

[2] J.R. Wolpaw, N. Birbaumer, D.J. McFarland, G. Pfurtscheller, T.M. Vaughan, "Brain–computer interfaces for communication and control", *Clin. Neurophysiol.,* 113:767–791, 2002.

[3] T. Kayikcioglu and O. Aydemir, "A polynomial fitting and k-NN based approach for improving classification of motor imagery BCI data", *Pattern Recognition Letters,* 31(11):1207-1215, 2010.

[4] G. Pfurtscheller, T. Solis-Escalante, et al., "Self-paced operation of an SSVEP-based orthosis with and without an imagery-based "Brain Switch:" A feasibility study towards a hybrid BCI", *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(4): 409-414, 2010.

[5] I. Guler and E.D. Ubeyli, "Multiclass support vector machines for EEG-signals classification", *IEEE Transactions on Information Technology in Biomedicine*, 11(2): 117-126, 2007.

[6] A. Schlogl, F. Lee, H. Bischof, G. Pfurtscheller, "Characterization of four-class motor imagery EEG data for the BCI-competition", *J. Neural Eng.*, 2, L14–22, 2005.

[7] D. Coyle, G. Prasa, T.M. McGinnity, "A time-series prediction approach for feature extraction in a brain–computer interface*", IEEE Trans. Neural Syst. Rehabil. Eng.*, 13 (4), 461–467, 2005.

[8] K.R. Muller, M. Krauledat, G. Dornhege, G. Curio, B. Blankertz, "Machine learning techniques for brain–computer interfaces", *Biomed. Technol.*, 49, 11–22, 2004.

[9] C.L. Zhao, C.X. Zheng, M. Zhao, et al., "Automatic classification of driving mental fatigue with EEG by wavelet packet energy and KPCA-SVM", *Int. J. Innov. Comput. I.*, 7(3): 1157-1168, 2011.

[10] M. Besserve, J. Martinerie, L. Garnero, "Improving quantification of functional networks with EEG inverse problem: Evidence from a decoding point of view", *Neuroimage*, 55(4):1536-1547, 2011.

[11] T.H. Falk, M. Guirgis, S. Power, et al., "Taking NIRS-BCIs outside the lab: Towards achieving robustness against environment noise", *IEEE T. Neur. Sys. Reh.*, 19(2):136-146, 2011.

[12] J.L.M. Perez and A.B. Cruz, "Adaptive RBF-HMM Bi-Stage Classifier Applied to Brain Computer Interface", *Biomedical Engineering Systems and Technologies*, 127:152-165, 2011.

[13] H. Cecotti, A. Graser, "Convolutional neural networks for P300 detection with application to brain-computer interfaces", *IEEE T. Pattern. Anal.*, 33(3): 433-445, 2011.

[14] S. Wang, D. Li, et al., "A feature selection method based on improved fisher's discriminant ratio for text sentiment classification" *Expert Systems with Applications*, 38(7):8696-8702, 2011.

[15] Q.L. Tran, K.A. Toh, et al., "An Empirical Comparison of Nine Pattern Classifiers", *IEEE T. Syst Man Cy B*, 35(5):1079-1091, 2005.

[16] S.J. Dixon and R.G. Brereton, "Comparison of performance of five common classifiers represented as boundary methods: Euclidean Distance to Centroids, Linear Discriminant Analysis, Quadratic Discriminant Analysis, Learning Vector Quantization and Support Vector Machines, as dependent on data structure", Chemometr Intell Lab., 95(1):1-17, 2009.

[17] A.C. Lorena, L.F.O. Jacintho, et al., "Comparing machine learning classifiers in potential distribution modeling", *Expert Systems With Applications*, 38(5):5268-5275, 2011.

[18] S.C. Lin, Y.C.I. Chang, W.N. Yang, "Meta-learning for imbalanced data and classification ensemble in binary classification", *Neurocomputing*, 73(1-3):484-494, 2009.

[19] C. Neuper, A. Schlögl, G. Pfurtscheller "Enhancement of left-right sensorimotor EEG differences during feedback-regulated motor imagery", *J. Clin. Neurophysiol*. 16(4):373-82, 1999.

[20] L. Svoboda, A. Stancak, P. Sovka, "Detection of cortical oscillations induced by SCS using power spectral density", *Radioengineering*, 16(4):38-45, 2007.

[21] S. Shahid, G. Prasad, "Bispectrum-based feature extraction technique for devising a practical brain-computer interface", *J. Neural Eng.*, 8(2), Article Number: 025014, 2011.

[22] P. Herman, G. Prasad, T.M. McGinnity, D. Coyle, "Comparative analysis of spectral approaches to feature extraction for EEG-based motor imagery classification", *IEEE T. Neur. Sys. Reh.*, 16(4):317-326, 2008.

[23] R.O. Duda, P.E. Hart, D.G. Stork, "*Pattern Classification",* New York: Wiley, 2nd edition, 2001.

[24] V. N. Vapnik, "*Statistical Learning Theory*" New York: Wiley, 1998.

[25] K.R. Muller, C.W. Anderson, G.E. Birch, "Linear and nonlinear methods for brain-computer interfaces", *IEEE Trans. Neural Syst. Rehabil. Eng.*, 11(2):165–169, 2003.

[26] T.M. Mitchell, "*Machine Learning*", McGraw-Hill, Singapore, 1997.

[27] L. Breiman, et al., "*Classification and Regression Trees*", Chapman & Hall, Boca Raton, 1993.