Wavelet Transform Based Classification of Invasive Brain Computer Interface Data

Onder AYDEMIR, Temel KAYIKCIOGLU

Dept. of Electrical and Electronics Engineering, Karadeniz Technical University, 61080, Trabzon, Turkey

onderaydemir@ktu.edu.tr, tkayikci@ktu.edu.tr

Abstract. *The input signals of brain computer interfaces may be either electroencephalogram recorded from scalp or electrocorticogram recorded with subdural electrodes. It is very important that the classifiers have the ability for discriminating signals which are recorded in different sessions to make brain computer interfaces practical in use. This paper proposes a method for classifying motor imagery electrocorticogram signals recorded in different sessions. Extracted feature vectors based on wavelet transform were classified by using k-nearest neighbor, support vector machine and linear discriminant analysis algorithms. The proposed method was successfully applied to Data Set I of BCI competition 2005, and achieved a classification accuracy of 94 % on the test data. The performance of the proposed method was confirmed in terms of sensitivity, specificity and Kappa and compared with that of other studies used the same data set. This paper is an extended version of our work that won the Best Paper Award at the 33rd International Conference on Telecommunications and Signal Processing.*

Keywords

Brain computer interface, classification, electrocorticogram, pattern recognition, wavelet transform.

1. Introduction

One of fundamental aim of Brain Computer Interface (BCI) research is to enable paralyzed patients to control a computer, a robotic arm or a variety of neuroprosthesis by constructing an interface. The input signals of BCIs may be either electroencephalogram (EEG) recorded from scalp or electrocorticogram (ECoG) recorded with subdural electrodes [1], [2].

EEG based BCIs are easy to use. However, because of their low spatial resolution it is difficult to predict subject's wishes and often require extensive user training sessions. Also, information transfer rate of such systems are still low [3], [4]. One method to boost predicted wishes accuracy (classification accuracy) of the subject is to increase the quality of input signal of a BCI system. In contrast with EEG, ECoG signals have better spatial resolution and higher signal to noise ratio [5]. These properties provide advantages to ECoG to be used as an input signal source of BCIs with positive results.

Recent studies seek to design appropriate preprocessing, feature extraction and classification algorithms suited to a motor imagery BCI application [6]. Different preprocessing algorithms have been proposed as input for a BCI, such as principal/independent component analysis (P/ICA) [7], [8], filtering methods (e.g. Laplacian filtering, time-frequency filtering) [9], [10], normalization methods (e.g. time domain normalization, noise normalization) [10], [11] and subsampling [12].

Leuthardt et al. reported for the first time that ECoG activity can enable users to control one-dimensional computer cursor rapidly and accurately [13]. They achieved 74–100% classification accuracy (CA) in a one-dimensional binary task study with four epilepsy patients over brief training periods of 3–24 minutes by modulation of carefully chosen spectral power features that are selected via a screening task. Lal et al. classified ECoG data from tongue and hand imagined movements [14]. They used autoregressive model coefficients as features for each channel, and recursive channel elimination for selecting the best channels for classification. They achieved classification accuracy of 82.5% using 1.5 seconds data with only two channels. In another ECoG based study Graimann et al. used wavelet packet analysis and genetic algorithms for selecting features related to event-related desynchronization and synchronization (ERD/ERS) that detect single-trial movement of body parts such as finger and lip [15]. They reported the presence of ERD/ERS features in delta (<3.5 Hz), beta (12.5-30 Hz) and gamma (70-90 Hz) bands associated with onset of a single discrete movement. And Symlet wavelets were found to be suitable for the analysis of the ERD of the brain rhythms by them.

In this study, we propose a method for classifying motor imagery ECoG signals recorded in different sessions. The method consists of three main parts as preprocessing, feature extraction based on Wavelet transform coefficients (WTCs) and the classification of motor imagery ECoG signals with *k*-nearest neighbor (*k*-NN), support vector machine (SVM) and linear discriminant analysis (LDA) algorithms. This idea was first presented in

[16]. Here, we further study the properties of the method and provide more theory and experimental results. More detailed analysis of the results is also presented by calculating three metrics: sensitivity, specificity and Kappa coefficient. The block diagram of the proposed method, shown in Fig. 1, summarizes the technical concepts. The details of the block diagram are described in the following sections.

For a BCI application, utilizing classification algorithm which provides higher classification accuracy, using fewer channels and features that are more representative of the tasks are key issues for efficient communication and control. The Data Set I signals, which are described in section 2 in detail, were also analyzed with different methods by other researchers. A performance comparison with those of other studies is provided in terms of classification accuracy and speed in section 4.

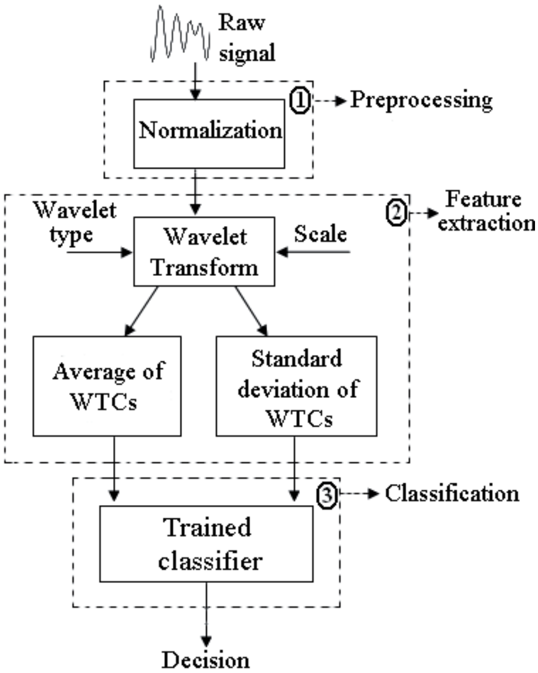

Fig. 1. Block diagram of the proposed method.

2. Material and Method

In the following subsections, first, the used data set is described, then, the parts of the proposed method are described in detail.

2.1 Data Set Description

For our study, we used the BCI competition 2005 Data Set I which was taken from an epilepsy subject on two different days with about one week of delay. In the both sessions the subject was asked to imagine of either the left small finger or the tongue movement.

The provided data sets consist of 278 trials (training data, 139 trials for finger movements, 139 trials for tongue movements) performed during the first session and 100 trials (test data) from the second session. Each trial's duration was 3 seconds. To avoid visually evoked potentials being reflected by the data, the recording intervals started 0.5 seconds after the visual cue had ended. Electrical brain activity was recorded with an 8x8 ECoG platinum electrode grid (totally from 64 points) which was placed on the contralateral (right) motor cortex. All recordings were performed with a sampling rate of 1 kHz (acquired 3000 samples per channel for every trial). For further information about the data set, please refer to [14], [17].

The purpose is to categorize the trials in the test set as finger or tongue movement imagery.

2.2 Wavelet Analysis

The continuous wavelet transform (CWT) is defined as the convolution of the signal $x(t)$ with the wavelet function $\psi_{\tau,s}(t)$ and is given by

$$CWT_x^{\psi}(\tau,s) = \frac{1}{\sqrt{|s|}} \int x(t)\,\psi^*\left(\frac{t-\tau}{s}\right)dt \tag{1}$$

where $\psi_{\tau,s}(t)$ is the dilated and shifted version of the wavelet function $\psi(t)$ and is defined as follows:

$$\psi_{\tau,s}(t) = \frac{1}{\sqrt{s}}\psi\left(\frac{t-\tau}{s}\right). \tag{2}$$

Here t, τ and s denote the time parameter, the shift parameter and the scale parameter, respectively [18]. The wavelet function $\psi_{\tau,s}(t)$ has a zero mean as given in (3).

$$\int_{-\infty}^{+\infty} \psi_{\tau,s}(t)dt = 0\,. \tag{3}$$

An important point of wavelet analysis is the choice of a particular wavelet function $\psi(t)$. Among other wavelets, we selected the Morlet wavelet function because it is well localized in the frequency domain. On the other hand the similarity of the wavelet function and the signals to be analyzed is significant to extract useful information. Compared with other types of wavelet, the Morlet wavelet has the most similar shape to that of the signals to be analyzed.

Wavelet transform is used for many BCI data analysis applications to extract feature(s) [19], [20], [21]. It has an advantage over other feature extraction methods that operate in only one domain, such as the Fourier transform, or autoregressive modeling.

2.3 Preprocessing

Subject's psychological and mental states influence his/her electrical brain activities in different sessions. This is one of the difficulties in EEG/ECoG based BCI technology. In the provided data the training set and the test set recorded in different sessions with about one week in between. This is an important point to pay attention to. Fig. 2

shows the averaged amplitude spectra of all trials obtained from both tasks (motor imagery of finger/tongue) in the training set and test set on randomly selected four channels (Channels 12, 29, 39 and 61). The averaged amplitude spectra axis was plotted in logarithmic scale. If the electrical brain activity had not been affected by the subject's psychological and mental state the two curves should have lain very close to each other. However, as seen in this figure, the averaged amplitude spectrum of the test set is far above that of the training set. This session to session variation can directly influence classification performance of the test set. Therefore a normalization process should be implemented to the training set and the test set in order to alleviate the impact of the magnitude change. In this study a variance normalization [22] process is implemented to the all trials.

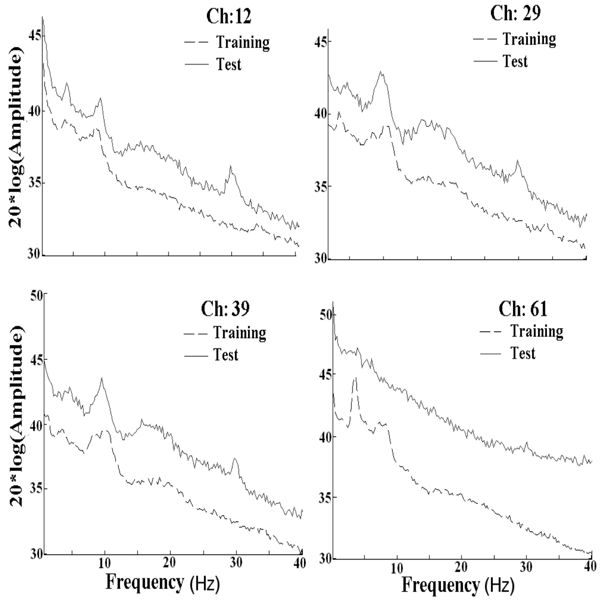

Fig. 2. The averaged amplitude spectra of the all trials obtained from both tasks in the training set and test set on randomly selected four channels, Ch: Channel. Amplitude spectra calculated using the MATLAB FFT function.

All trails are normalized to its own standard deviation as:

$$T_N = \frac{T}{std(T)} \tag{4}$$

where T_N and $std(T)$ denote the normalized trial signal and standard deviation of the trial signal, respectively. The estimated standard deviation of a trial was calculated by the following formula:

$$std(T) = \sqrt{\frac{\sum (T - \bar{T})^2}{n-1}} \tag{5}$$

where $\bar{T}$ is the mean value of all the samples of the trial and n is the length of the trial which is equal to 3000.

As seen from Fig. 3 that after the normalization process the averaged amplitude spectrum of the test set has resemblance to that of the training set.

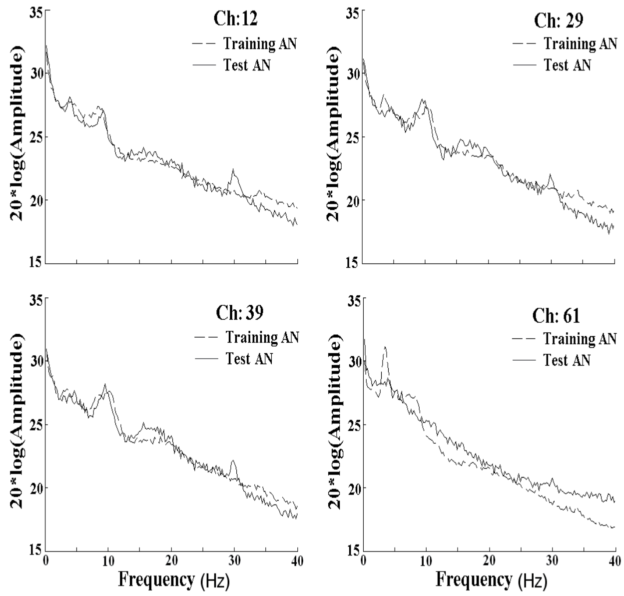

Fig. 3. The averaged amplitude spectra after normalization (AN).

2.4 Feature Extraction

The calculated WTCs provide a compact representation that shows the energy distribution of the ECoG signal in time and frequency. In this study, in order to extract features continuous wavelet transform was applied to each trial and each channel, separately. By using Morlet wavelet transform, the statistical feature analysis on the training set demonstrated that the averages and the standard deviations of the absolute values of the WTCs of the 12th and 29th channels can be used for classification of the both tasks. The averages and the standard deviations were calculated by the following formulas, respectively:

$$WTCs^{Avr} = \frac{\sum |WTCs|}{L_{WTCs}}, \tag{6}$$

$$WTCs^{std} = \sqrt{\frac{\sum (|WTCs| - WTCs^{Avr})^2}{L_{WTCs} - 1}} \tag{7}$$

where L_{WTCs} is the length of the WTCs. All computed WTCs of the selected channels were used to calculate features.

Fig. 4 shows the four features: Fig. 4(a) (first feature, *f1*) and Fig. 4(c) (third feature, *f3*) the averages of the absolute values of the WTCs ($WTCs^{Avr}$), Fig. 4(b) (second feature, *f2*) and Fig. 4(d) (fourth feature, *f4*) the standard deviations of the absolute values of the WTCs ($WTCs^{std}$) of the selected channels.

To calculate WTCs we used Morlet wavelet function from the Matlab. The scales of the wavelet function was set to integer values between 1 and 110 with step size of $\tau = 4$. The scale 1 and 110 correspond to 812.5 Hz and 7.3 Hz, respectively.

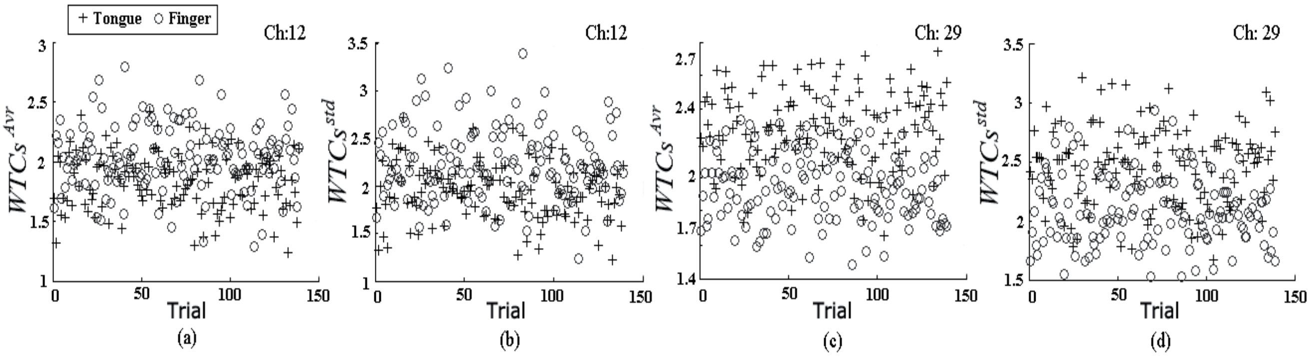

Fig. 4. Extracted feature vectors, (a) *f1*, (b) *f2*, (c) *f3*, (d) *f4*.

2.4 Classification Algorithms

The aim of this part is to introduce the *k*-NN as a powerful classifier for this study. But, to show its power and to make a fair comparison, it is necessary to compare it with some other classifiers especially which were used by other researchers for the same dataset. Hence, we also utilized the SVM and the LDA other than the *k*-NN.

The *k*-NN classifier is a traditional classification algorithm, which assigns an unseen point to the dominant class among its *k* nearest neighbors within the feature set [23]. Although *k*-NN algorithms are not very popular in the BCI community, especially they may be efficient with low dimensional feature vectors [24].

The SVM aims at finding a hyper-plane in the feature space, which simultaneously minimizes empirical classification errors and maximizes the distance between this hyper-plane and the nearest data point of each class. The architecture of the SVM depends on the regularization parameter C and the type of the kernel function. There are various kernel functions including: linear, polynomial, radial basis function (RBF), and sigmoid.

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

$$y = w^T x + w_0 \qquad (8)$$

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.

The LDA classifier is more robust than the *k*-NN and the SVM algorithms since it has only limited flexibility (less free parameters to tune) and is less prone to over fitting [25]. Some of the good advantages of the LDA and the *k*-NN classifiers are that they are simple to use and have very low computational and training requirements which make them suitable for the online BCI system [24], [26], [27].

2.5 Training Procedures

In order to train the classifiers we used two popular techniques which are K-fold cross-validation (K-FCV) and leave-one-out cross-validation (LOOCV). In the K-FCV, we first randomly split the training set into K equal subsets, using K-1 subsets for the training and 1 subset for the testing (validation set). This procedure is repeated until each subset is used for testing exactly once. The results obtained from each subset are averaged to produce a single estimation. In our study, we chose $K = 10$ for determining the folds.

The LOOCV is a special case of K-FCV with $K = M$, the size of the training set. Hence, the validation sets are all of size 1.

Based on these techniques, we determined optimum values of parameters of classification algorithms, optimum in the sense of maximizing the average of the validation set's accuracy rate. The 10-FCV routine was repeated 30 times and the average classification accuracy over all runs was presented as a measure of classifier performance. Since the LOOCV provides one of the best uses of the available training data and avoids the problems of random selections, its routine was utilized only once.

3. Results

In the following subsections the training and test accuracy results of the classification algorithms for Data Set I of BCI competition 2005 are presented.

3.1 Results of the *k*-NN Classification

For the *k*-NN algorithm we used *knnclassify* function from the MATLAB Bioinformatics Toolbox. The best *k* value is searched with the Euclidean distance function in interval between 1 and 25. The averages and standard deviations of the classification accuracy of the training data and the classification accuracy of the test data are shown in Tab. 1. The classification accuracy was defined as the percentage of the number of trials classified correctly over the size of the data set. As seen in Tab. 1, 10-FCV appears to produce somewhat better test accuracy results than the LOOCV, with accuracies in range between 67% and 94% and 68% and 92%, respectively. This table also shows the calculated optimal *k* values which were used to classify the test data.

Features	10-FCV			LOOCV		
	Training CA	Test CA	*k*	Training CA	Test CA	*k*
f1+f2	68.1 ±7.0	67	19	63.0	68	23
f1+f3	86.9 ±5.5	90	17	84.2	90	11
f1+f4	**81.2 ±6.7**	**94**	**20**	79.9	92	23
f2+f3	86.6 ±4.5	89	19	83.5	88	20
f2+f4	82.3 ±5.8	90	19	80.6	90	16
f3+f4	79.6 ±6.3	78	17	78.1	79	13

Tab. 1. Results of the *k*-NN algorithm.

3.2 Results of the SVM Classification

For the SVM training and classifying, we used the *svmtrain* and *svmclassify* functions from the MATLAB Bioinformatics Toolbox, respectively. The SVM algorithm with Gaussian radial basis function was enabled. 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. The regularization parameter was set to its default value $C = 1$. The best σ value is searched in interval between 0.1 and 2.5, with step size of 0.1.

Features	10-FCV			LOOCV		
	Training CA	Test CA	σ	Training CA	Test CA	σ
f1+f2	65.4 ±0.8	64	1.0	64.4	62	2.0
f1+f3	84.2 ±0.5	91	1.0	83.5	91	1
f1+f4	81.1 ±0.8	90	1.2	79.9	89	0.8
f2+f3	84.2 ±0.6	91	1.2	83.8	90	1.5
f2+f4	81.4 ±0.6	92	1.0	80.2	87	0.4
f3+f4	77.9 ±0.6	78	1.0	77.3	77	0.6

Tab.2. Results of the SVM algorithm.

The results of this algorithm are shown in Tab. 2. For the 10-FCV the classification accuracies of the test data fall in range between 64% and 92%, whereas the range of the LOOCV extends from 62% to 91%. This table also shows the calculated optimal σ values which were used to classify the test data.

3.3 Results of the LDA Classification

For the LDA algorithm, we used the *classify* function from the MATLAB Statistics Toolbox. In classification, the same validation procedure was used as in the *k*-NN and the SVM classification algorithms described above. For the 10-FCV procedure the only difference was we did not seek any tune parameter as we utilized for the *k*-NN and the SVM training sections. However, a different LDA classifier was produced to classify each new validation set. The best LDA classifier was determined when the validation set classified at the highest classification accuracy. The results of the LDA algorithm are shown in Tab. 3. The 10-FCV achieved marginally better test accuracy results than the LOOCV, with accuracies in range between 62%-91% and 60%-90%, respectively.

Features	10-FCV		LOOCV	
	Training CA	Test CA	Training CA	Test CA
f1+f2	66.2	62	64.0	60
f1+f3	82.8	91	82.4	90
f1+f4	79.5	90	78.1	90
f2+f3	83.5	90	82.0	86
f2+f4	82.0	88	80.9	88
f3+f4	78.4	77	78.1	76

Tab. 3. Results of the LDA algorithm.

4. Performance Comparison

We compare the performance of the proposed method in the following three subsections. In the first subsection we measure the effectiveness of the classifiers in terms of sensitivity, specificity and Kappa coefficient. In the second subsection, the performance of the proposed method is compared with that of other studies in terms of the best CA and speed. In the last subsection the computational times of the 10-FCV and LOOCV techniques are presented.

4.1 Effectiveness of the Classifiers

Apart from the classification accuracy, we also measured the effectiveness of the classifiers by calculating three metrics: sensitivity (SE), specificity (SP) and Kappa (κ). Sensitivity and specificity are calculated by the following formulas, respectively:

$$sensitivity = \frac{TP}{TP + FN}, \tag{9}$$

$$specificity = \frac{TN}{TN + FP} \tag{10}$$

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

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

$$Kappa = \frac{P(A) - P(E)}{1 - P(E)} \tag{11}$$

where $P(A)$ 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 (or chance).

Sensitivity, specificity and Kappa values were calculated by analyzing the output data of classifiers for each feature pair obtained from the test set. The results of these three metrics for the *k*-NN, SVM and LDA algorithms are given in Tab. 4, Tab. 5 and Tab. 6, respectively.

Features	10-FCV			LOOCV		
	SE	SP	κ	SE	SP	κ
f1+f2	0.77	0.62	0.34	0.80	0.63	0.36
f1+f3	0.94	0.87	0.80	0.92	0.89	0.80
f1+f4	0.98	0.91	0.88	0.98	0.88	0.84
f2+f3	0.98	0.83	0.78	0.98	0.82	0.76
f2+f4	0.98	0.85	0.80	0.98	0.85	0.80
f3+f4	0.80	0.76	0.56	0.84	0.75	0.58

Tab. 4. Effectiveness of the *k*-NN algorithm.

Features	10-FCV			LOOCV		
	SE	SP	κ	SE	SP	κ
f1+f2	0.73	0.60	0.28	0.71	0.58	0.24
f1+f3	0.94	0.89	0.82	0.96	0.87	0.82
f1+f4	0.94	0.87	0.80	0.92	0.87	0.78
f2+f3	0.98	0.86	0.82	0.96	0.86	0.80
f2+f4	0.98	0.88	0.84	0.93	0.83	0.74
f3+f4	0.80	0.76	0.56	0.80	0.75	0.54

Tab. 5. Effectiveness of the SVM algorithm.

As seen in the tables, the best case was obtained when *f1* and *f4* feature pair classified by using the *k*-NN algorithm with 10-FCV technique. In this case, the all three metrics reached the highest values, where sensitivity = 0.98, specificity = 0.91 and Kappa = 0.88. The worst case was obtained when *f1* and *f2* feature pair classified by using the LDA algorithm with LOOCV technique. In this case, sensitivity, specificity and Kappa reached the lowest values with 0.68, 0.57 and 0.20, respectively.

Features	10-FCV			LOOCV		
	SE	SP	κ	SE	SP	κ
f1+f2	0.71	0.58	0.24	0.68	0.57	0.20
f1+f3	0.96	0.87	0.82	0.94	0.87	0.80
f1+f4	0.96	0.86	0.80	0.96	0.86	0.80
f2+f3	0.98	0.85	0.80	0.93	0.81	0.72
f2+f4	0.95	0.83	0.76	0.95	0.83	0.76
f3+f4	0.79	0.76	0.54	0.78	0.74	0.52

Tab. 6. Effectiveness of the LDA algorithm.

A close observation to the results in the tables reveals that the sensitivity values are greater than those of the specificity in all cases. This means that distinguishability of the finger movement imagery is bigger than the tongue movement imagery for the all feature pairs.

4.2 The Best Performances of Other Studies

Speed and accuracy are key issues in brain computer interface (BCI) technology. As we mentioned before the Data Set I of BCI Competition 2005 has been also analyzed with different methods by other researchers. They mostly utilized the SVM [5], [22], [28], [29] and the LDA [30], [31] classification algorithms.

Tab. 7 presents the comparison of the proposed method to other methods in terms of number of used channel(s), number of feature(s), characteristics of classifier and classification accuracy. The best classification accuracy results on the test data are listed on the last column. All accuracy results, including ours, were obtained by completely utilizing the training set. The results indicate that the proposed method achieved by 1% improvement over the best accuracy result of other studies.

Study	NOC	NOF	Classifier	CA
Qingguo et al. [5]	64	3	SVM	91
Dat et al. [22]	1	NC	SVM	92
Qin et al. [28]	32	20	SVM	90
Demirer et al. [29]	10	373	SVM	73
Yan Li et al. [30]	It is said "some channels"	4	LDA	92
Ince et al. [31]	64	3	LDA	93
Hu et al. [32]	4	4	KIII Model	92
Proposed method	2	2	*k*-NN	94

Tab. 7. Performance comparison (NOC: Number of channel(s), NOF: Number of feature(s), NC: Not clarified).

4.3 Computational Times

We also computed the training times of the 10-FCV and the LOOCV techniques, which are presented in Tab. 8 in seconds. For the both techniques the LDA algorithm was faster than the *k*-NN and the SVM algorithms. The fastest training time was obtained as 2 CPU seconds with the LDA by using 10-FCV. As seen in the table that LOOCV technique causes time-consuming, especially for the *k*-NN and the SVM algorithms.

Classifier	10-FCV	LOOCV
k-NN	7	150
SVM	56	750
LDA	2	5

Tab. 8. Training times of the classifiers.

For the entire test set, the average test times of the LDA, the *k*-NN and the SVM algorithms were approximately 0.006, 0.04, 0.85 seconds, respectively. All the runtime experiments were conducted on a PC with Intel Pentium® 4 processor at 2.80 GHz, 1 GB RAM.

5. Conclusion and Discussion

In this paper, we proposed a wavelet based method for improving the speed and classification accuracy of BCI ECoG signal for distinguishing tongue/finger movement imageries. Our approach has been successfully applied to the BCI Competition 2005 Data Set I.

The experiments proved that a normalization procedure is necessary in order to alleviate the impact of the magnitude change in between the training set and test set of motor imagery ECoG signals, recorded in different sessions. On the other hand it is shown that signal analysis based on WTCs can be reliably used as feature to accurately classify two types of imagined movements.

One of good attributes of the proposed method is its simplicity in the feature extraction procedure. By using only two features (*f1* and *f4*), extracted from the coefficients of the wavelet transform applied to the data of only two channels (Channel 12 and Channel 29), the *k*-NN algorithm trained with 10-FCV technique achieved 94% classification accuracy on the test set. This case was confirmed in terms of sensitivity, specificity and Kappa, where they reached their highest values.

The computation of the WTCs and the entities is fast and simple. Fig. 5 shows entire feature vectors extracted from the training set and the test set. Horizontal and vertical axes of this feature space are values of *f1* and *f4*, respectively. Plus points stand for trials of tongue movements, and circle points stand for the trials of finger movements.

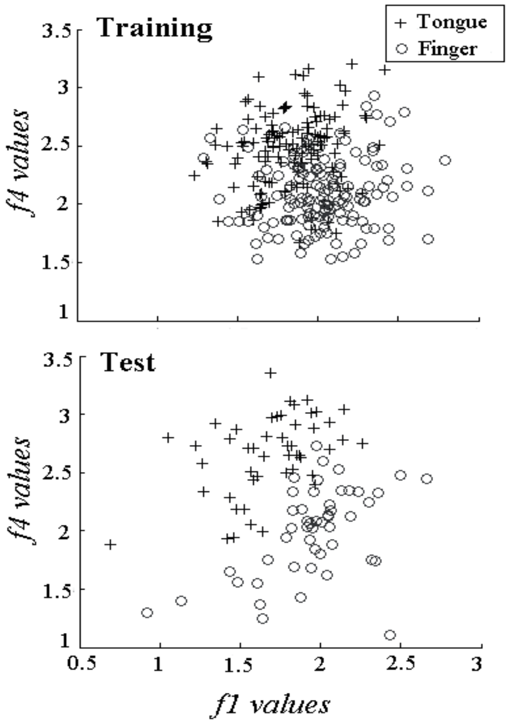

Fig. 5. Feature vectors for Channel 12 and 29.

The overall computational complexity of the proposed algorithm is low in both training and testing stages. We observed that our method outperformed the existing other researchers' results by using small number of features, sensors and by achieving higher classification accuracy with a faster algorithm, *k*-NN.

The results also showed that for low dimensional feature vectors the LDA algorithm is the fastest technique compared to the *k*-NN and the SVM algorithms in terms of training and testing speeds. However, it could obviously mentioned that the *k*-NN algorithm has reasonable speed and also achieved much better classification accuracy performance than the LDA and the SVM algorithms.

Based on these results, we believe that the proposed method has great potential for the construction of a robust ECoG based invasive BCI system.

Acknowledgements

The authors would like to thank all the researchers who are associated with the data acquisition and processing of the data set. This work was supported by the TUBITAK (The Scientific and Technological Research Council of Turkey), Project No: EEEAG-110E035.

References

[1] WEI, Q. G., LU, Z. W., CHEN, K., et al. Channel selection for optimizing feature extraction in an electrocorticogram-based brain-computer interface. *Journal of Clinical Neurophysiology*, 2010, vol. 27, no. 5, p. 321-327.

[2] VANSTEENSEL, M. J., HERMES, D., AARNOUTSE, E. J., et al. Brain-computer interfacing based on cognitive control. *Annals of Neurology*, 2010, vol. 67, no. 6, p. 809-816.

[3] FREEMAN, W. J., HOLMES, M. D., BURKE, B. C., VANHATALO, S. Spatial spectra of scalp EEG and EMG from awake humans. *Clin. Neurophysiol.,* 2003, vol. 114, p. 1053–1068.

[4] LEE, P. L., SIE, J. J., LIU, Y. J., et al. An SSVEP-actuated brain computer interface using phase-tagged flickering sequences: A cursor system. *Annals of Biomedical Engineering*, 2010, vol. 38, no. 7, p. 2383-2397.

[5] QINGGUO, W., FEI, M., YIJUN, W., XIARONG, G., SHANGKAI, G. Feature combination for classifying single-trial ECoG during motor imagery of different sessions. *Progress in Natural Science*, 2007, vol. 17, no. 7, p. 851-858.

[6] BRUNNER, C., NAEEM, M., LEEB, R., GRAIMANN, B., PFURTSCHELLER, G. Spatial filtering and selection of optimized components in four class motor imagery EEG data using independent components analysis. *Pattern Recognition Letters*, 2007, vol. 28, no. 8, p. 957-964.

[7] MING, D., AN, X. W., XI, Y. Y., et al. Time-locked and phase-locked features of P300 event-related potentials (ERPs) for brain-computer interface speller. *Biomedical Signal Processing and Control*, 2010, vol. 5, no. 4, p. 243-251.

[8] RUCKAY, L., STASTNY, J., SOVKA, P. ICA model order estimation using clustering method. *Radioengineering*, 2007, vol. 16, no. 4, p. 51-57.

[9] KAMOUSI, B., AMINI, A. N., HE, B. Classification of motor imagery by means of cortical current density estimation and Von Neumann entropy. *Journal of Neural Engineering*, 2007, vol. 4, no. 2, p. 17-25.

[10] GUTIERREZ, D., ESCALONA-VARGAS, D. I. EEG data classification through signal spatial redistribution and optimized linear discriminants. *Computer Methods and Programs in Biomedicine*, 2010, vol. 97, no. 1, p. 39-47.

[11] KHALID, M. B., RAO, N. I., RIZWAN-I-HAQUE, I., et al. A brain computer interface (BCI) using fractional Fourier

transform with time domain normalization and heuristic weight adjustment. *9th International Conference on Signal Processing*, 2008, vol. 1, no. 5, p. 2731-2734.

[12] HAMMON, P. S., DE SA, V. R. Preprocessing and meta-classification for brain-computer interfaces. *IEEE Transactions on Biomedical Engineering*, 2007, vol. 54, no. 3, p. 518-525.

[13] LEUTHARDT, E. C., SCHALK, G., WOLPAW, J. R., OJEMANN, J. G., MORAN, D. W. A brain-computer interface using electrocorticographic signals in humans. *J. Neural Eng.*, 2004, vol. 1, no. 2, p. 63–71.

[14] LAL, T. N., HINTERBERGER, T., WIDMAN, G., et al. Methods towards invasive human brain interfaces. *Advances in NIPS*, 2005, vol. 17, p. 737-744.

[15] GRAIMANN, B., HUGGINS, J. E., LEVINE, S. P., PFURTSCHELLER, G. Towards a direct brain interface based on human subdural recordings and wavelet packet analysis. *IEEE Trans BME*, 2004, vol. 51, no. 6, p. 954–962.

[16] AYDEMIR, O., KAYIKCIOGLU, T. Classifying ECoG based mental tasks using wavelet transform features. In *33rd International Conference on Telecommunications and Signal Processing*. Baden near Vienna (Austria), 2010, p. 103-107.

[17] BCI Competition 2005, www.bbci.de/competition/iii

[18] ADELI, H., ZHOU, Z., DADMEHR, N. Analysis of EEG records in an epileptic patient using wavelet transform. *Journal of Neuroscience Methods*, 2003, vol. 123, p. 69–87.

[19] SUBASI, A. EEG signal classification using wavelet feature extraction and a mixture of expert model. *Expert Systems with Applications*, 2007, vol. 32, p. 1084-1093.

[20] 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 BME*, 2004, vol. 51, no. 6, p. 1057-1061.

[21] FATOURECHI, M., BIRCH, G. E., WARD, R. K. Application of a hybrid wavelet feature selection method in the design of a self-paced brain interface system. *J Neuroeng Rehabil.*, 2007, vol. 4, no. 11.

[22] DAT, T. H., SHUE, L., GUAN, C. Electrocorticographic signal classification based on time-frequency decomposition and nonparametric statistical modeling. In *Proceedings of the 28th IEEE EMBS international conference*. USA, 2006, p. 2292-2295.

[23] DUDA, R. O., HART, P. E., STORK, D. G. *Pattern Classification*. New York: Wiley, 2001. 2nd edition.

[24] LOTTE, F., CONGEDO, M., LECUYER, A., LAMARCHE, F., ARNALDI, B. A review of classification algorithms for EEG-based brain–computer interfaces. *Journal of Neural Engineering*, 2007, R1-R13.

[25] MULLER, K.-R., ANDERSON, C. W., BIRCH, G. E. Linear and nonlinear methods for brain-computer interfaces. *IEEE Trans. Neural Syst. Rehabil. Eng.*, 2003, vol. 11, no. 2, p. 165–169.

[26] LUGGER, K., FLOTZINGER, D., SCHLOGL, A., et al. Feature extraction for on-line EEG classification using principal components and linear discriminants. *Medical & Biological Engineering & Computing*, 1998, vol. 36, no. 3, p. 309-314.

[27] LENHARDT, A., KAPER, M., RITTER, H. J. An adaptive P300-based online brain-computer interface. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2008, vol. 16, no. 2, p. 121-130.

[28] QIN, J., LI, Y., SUN, W. A semisupervised support vector machines algorithm for BCI systems. *Computational Intelligence and Neuroscience,* 2007, 94397.

[29] DEMIRER, R. M., OZERDEM, M. S., BAYRAK, C. Classification of imaginary movements in ECoG with a hybrid approach based on multi-dimensional Hilbert-SVM solution. *Journal of Neuroscience Methods*, 2009, vol. 178, no. 1, p. 214 to 218.

[30] LI, Y., KOIKE, Y., SUGIYAMA, M. A Framework of adaptive brain computer interfaces. In *Proceedings of the 2nd International Conference on BioMedical Engineering and Informatics*. Tianjin (China), 2009, p. 1-5.

[31] INCE, N. F., GOKSU, F., TEWFIK, A. H. ECoG based brain computer interface with subset selection. *Communications in Computer and Information Science*, 2008, vol. 25, p. 357-374.

[32] HU, R., LI, G., HU, M., FU, J., FREEMAN, W. J. Recognition of ECoG in BCI systems based on a chaotic neural model. *Lecture Notes in Computer Science,* 2007, vol. 4491, p. 685-693.

About Authors...

Onder AYDEMIR was born in Trabzon, Turkey. He received his B.Sc. and M.Sc. degrees from Karadeniz Technical University (KTU) in 2005 and 2008, respectively. He studied for his master thesis in Vienna University of Technology for one year. He is currently a Ph.D. student at the Department of Electrical and Electronics Engineering, KTU. His research interests include biomedical signal processing, pattern recognition and classification.

Temel KAYIKCIOGLU was born in Trabzon, Turkey. He received his B.Sc. and M.Sc. degrees from KTU in 1984 and 1987, respectively and the Ph.D. degree from Texas Tech University, USA in 1993. He is a Professor at the Department of Electrical and Electronics Engineering, KTU. His research interests include biomedical signal and image processing, pattern recognition, computational neuroscience and telecommunication applications.