Multi-Scale Wavelet Kernel Extreme Learning Machine for EEG Feature Classification

Qi Liu[1], Xiao-guang Zhao[1], Zeng-guang Hou[1] and Hong-guang Liu[2]
1 The State Key Laboratory of Management and Control for Complex Systems
Institute of Automation, CAS
Beijing, PRC
2 Institute of crime, Chinese People's Public Security University
Beijing, PRC

***Abstract*—In this paper, the principle of the kernel extreme learning machine (ELM) is analyzed. Based on that, we introduce a kind of multi-scale wavelet kernel extreme learning machine classifier and apply it to electroencephalographic (EEG) signal feature classification. Experiments show that our classifier achieves excellent performance.**

Keywords—EEG classification; ELM; multi-scale wavelet kernel

I. Introduction

Electroencephalographic (EEG) is a kind of typical and important biological signal. It reflects the electrical activity and the functional status of the brain. Also, it has been proved that EEG has inevitable connection with human intention. EEG has been applied to clinical medicine [1] and cognitive science [2].

In a practical application, how to recognize EEG features effectively is the most critical part in EEG signal processing. The recognition procedure mainly includes feature extraction and classification. Many methods on that have been developed. Feature extraction techniques are roughly divided into four categories [3]: time or frequency methods, conventional time-frequency methods, model parameter methods, and wavelet decomposition-based methods. Also, classification methods widely used include hidden Markov models (HMM) [4], k-means clustering [5], k-nearest neighbors (kNN) [6], neural networks [7], support vector machines (SVM) [8, 9], etc. In spite of good effects the mentioned classifiers achieved, due to the poor signal-to-noise ratio (SNR) of raw EEG signals and actual application needs, the fast and accurate classification of EEG signals is still challenging.

Recently, a new machine learning algorithm referred to as extreme learning machine (ELM) proposed by Huang et al. has been widely adopted in pattern classification in recent years. [10]. Compared to other algorithms, ELM can achieve satisfactory classification accuracy but require less training time. Many problems encountered by traditional gradient-based neural network learning algorithms, including local minima and various training parameters (training efficiency, stopping criteria, learning epochs and the hidden layer unit number) are avoided in ELM. Also, ELM has higher generalization performance than the established gradient-based learning methods. For its superior performance, ELM has been applied to EEG signal feature classification [11, 12].

This paper focuses on the classification process in the recognition procedure for EEG signals. We present a new algorithm, which introduces the multi-scale wavelet as the kernel function into ELM. The effectiveness of the algorithm is validated through experiments on the dataset Ⅱ of the brain-computer interface (BCI) Competition Ⅲ (P300 speller).

II. Kernel ELM

Compared to traditional single-hidden layer feedforward neural networks (SLFNs), ELM not only tends to reach the smallest training error but also the smallest norm of output weights, which will make it have better performance [13]. In ELM, the hidden layer need not be tuned iteratively. The hidden layer parameters can be given randomly at the beginning and fixed during the process of training. Then the output weights can be resolved using the lease-square method [10]. Moreover, Huang et al. [14] put forward that kernel method can also be applied to ELM, which will makes the algorithm obtain more stable and better generalization performance.

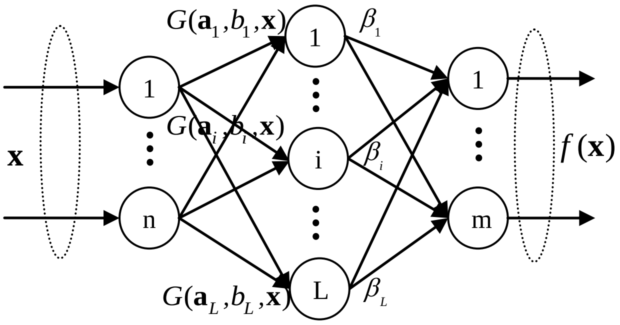
A. Basic ELM

Suppose there are N arbitrary samples $(\mathbf{x}_i, \mathbf{t}_i)$, where $\mathbf{x}_i = [x_{i1}, x_{i2}, \cdots, x_{in}]^{\mathrm{T}} \in \mathbf{R}^n$, $\mathbf{t}_i = [t_{i1}, t_{i2}, \cdots, t_{im}]^{\mathrm{T}} \in \mathbf{R}^m$. Then standard SLFNs with L hidden nodes can be mathematically expressed as following:

$$f_L(\mathbf{x}) = \sum_{i=1}^{L} \beta_i g_i(\mathbf{x}) = \sum_{i=1}^{L} \beta_i G(\mathbf{a}_i, b_i, \mathbf{x}_j), j = 1, \cdots, N, \quad (1)$$

where $\mathbf{a}_i = [a_{i1}, a_{i2}, \cdots, a_{in}]^{\mathrm{T}}$ is the weight vector connecting the i th hidden node and the input nodes, b_i is the threshold of the i th hidden node. $\beta_i = [\beta_{i1}, \beta_{i2}, \cdots, \beta_{im}]^{\mathrm{T}}$ is the weight vector connecting the i th hidden node and the output nodes. g_i denotes the output function $G(\mathbf{a}_i, b_i, \mathbf{x})$ of the ith hidden node (cf. Fig. 1).

Support by the National Natural Science Foundation China (Grant No. 61271432 and 61333016).



$G(\mathbf{a}_1,b_1,\mathbf{x})$
β_1
$G(\mathbf{a}_i,b_i,\mathbf{x})$
β_i
$G(\mathbf{a}_L,b_L,\mathbf{x})$
β_L
x
1
n
i
L
m
$f(\mathbf{x})$

Fig.1 Single hidden layer feedforward network

If the SLFN with activation function $g(x)$ can approximate these N training samples $(\mathbf{x}_i,\mathbf{t}_i)$ with zero error, the following liner system is set up.

$$\mathbf{H}\beta = \mathbf{T} \tag{2}$$

where

$$\mathbf{H} = \begin{bmatrix} h(x_1) \\ \vdots \\ h(x_N) \end{bmatrix}_{N\times L}$$

$$= \begin{bmatrix} G(\mathbf{a}_1,b_1,\mathbf{x}_1) & \cdots & G(\mathbf{a}_L,b_L,\mathbf{x}_1) \\ \vdots & \cdots & \vdots \\ G(\mathbf{a}_1,b_1,\mathbf{x}_N) & \cdots & G(\mathbf{a}_L,b_L,\mathbf{x}_N) \end{bmatrix}_{N\times L},$$

$$\beta = \begin{bmatrix} \beta_1^{\mathrm{T}} \\ \vdots \\ \beta_L^{\mathrm{T}} \end{bmatrix}_{L\times m}, \quad \mathbf{T} = \begin{bmatrix} \mathbf{t}_1^{\mathrm{T}} \\ \vdots \\ \mathbf{t}_N^{\mathrm{T}} \end{bmatrix}_{N\times m}$$

The weight β can be obtained by solving the following equations by least-square method.

$$\left\|\mathbf{H}\hat{\beta} - \mathbf{T}\right\| = \min_{\beta}\left\|\mathbf{H}\beta - \mathbf{T}\right\| \tag{3}$$

The solution is

$$\hat{\beta} = \mathbf{H}^{+}\mathbf{T} \tag{4}$$

The training steps of ELM algorithm is as follows.

Step1: Randomly assign input weights a_i and biases b_i according to some continuous probability density function;

Step2: Calculate the hidden layer output matrix $\mathbf{H}$;

Step3: Calculate the output weights $\hat{\beta} = \mathbf{H}^{+}\mathbf{T}$.

B. Kernel ELM

If HH^{T} is nonsingular, to improve the stability of ELM, we can have:

$$\beta = \mathbf{H}^{T}\left(\frac{1}{C} + \mathbf{H}\mathbf{H}^{T}\right)^{-1}\mathbf{T}, \tag{5}$$

where $1/C$ is a positive value, and the corresponding output function of ELM is:

$$h(\mathbf{x})\beta = h(\mathbf{x})\mathbf{H}^{T}\left(\frac{1}{C} + \mathbf{H}\mathbf{H}^{T}\right)^{-1}\mathbf{T} \tag{6}$$

If the hidden layer feature mapping $h(\mathbf{x})$ is unknown to users, an ELM kernel function can be constructed to replace $\mathbf{H}\mathbf{H}^{T}$ [13]:

$$\mathbf{\Omega}_{ELM} = \mathbf{H}\mathbf{H}^{T} = \begin{bmatrix} K(x_1,x_1) & \cdots & K(x_1,x_N) \\ \vdots & \cdots & \vdots \\ K(x_N,x_1) & \cdots & K(x_N,x_N) \end{bmatrix} \tag{7}$$

Thus equation (6) can be written as:

$$h(\mathbf{x})\beta = \begin{bmatrix} K(\mathbf{x},\mathbf{x}_1) \\ \vdots \\ K(\mathbf{x},\mathbf{x}_N) \end{bmatrix}^{T}\left(\frac{1}{C} + \mathbf{\Omega}_{ELM}\right)^{-1}\mathbf{T} \tag{8}$$

For binary classification, the decision function of kernel ELM is:

$$f(\mathbf{x}) = \operatorname{sign}\left(\begin{bmatrix} K(\mathbf{x},\mathbf{x}_1) \\ \vdots \\ K(\mathbf{x},\mathbf{x}_N) \end{bmatrix}^{T}\left(\frac{1}{C} + \mathbf{\Omega}_{ELM}\right)^{-1}\mathbf{T}\right) \tag{9}$$

Compared to traditional ELM, kernel ELM has more powerful function approximation ability.

III. MUTI-SCALE WAVELET KERNEL ELM

A. Wavelet Kernel Function

Compared to other kernel functions, the approximation ability of the wavelet kernel function is more powerful. It has been introduced to SVM [16]. Moreover, it has been proved that the ability of SVM with wavelet kernel dealing with the nonlinear classification problem is stronger than common kernels. This section will introduce the wavelet kernel to ELM to analyze.

The essence of the wavelet analysis is to express or approximate a signal or function through a family of functions generated by dilations and translations of a function called the mother wavelet. The following equation can express the wavelet base function:

$$h_{a,b}(x) = \sqrt{|a|}\,h\left(\frac{x-b}{a}\right), \tag{10}$$

where a and b are the dilation factor and translation factor, respectively.

A multidimensional wavelet function can be written as the tensor product of multiple one-dimensional wavelets:

$$h(\mathbf{x}) = \prod_{i=1}^{n} h(x_i) \tag{11}$$

According to the above formula, if $\boldsymbol{x}, \boldsymbol{x}^{'} \in R^n$, we can construct the translation invariant kernel function as follows:

$$K(\mathbf{x},\mathbf{x}') = \prod_{i=1}^{n} h\left(\frac{x_i - x_i'}{a}\right) \tag{12}$$

We select the Morlet wavelet function:

$$h(x) = \cos(1.75x)\exp\left(-\frac{x^2}{2}\right), \tag{13}$$

then the corresponding kernel function can be represented as below：

$$K(\mathbf{x},\mathbf{x}^{'}) = \prod_{i}^{n}[\cos(1.75\times(\frac{x_i - x_i'}{a})\exp(-\frac{(x_i - x_i')^2}{2a^2})] \tag{14}$$

In some complicated cases, the kernel machine constructed by single kernel function cannot satisfy for the request of applications such as uneven distribution of training samples, huge sample size. The combination of multiple kernel functions can obtain better approximation ability.

Commonly, we can construct a hybrid kernel function by means of superposition of different kernel functions to improve the classifying ability, i.e.,

$$K = a_1 * K_1 + a_2 * K_2 + \cdots + a_n * K_n \tag{15}$$

The wavelet kernel function not only has the characteristic of nonlinear mapping but also integrates the characteristic of wavelet analysis which describe the non-stationary input parameters level by level detailed. ELM with wavelet kernel function adopted will also have this ability.

B. Muti-scale wavelet kernel ELM classifier

The wavelet kernel function itself has the ability of multi-scale extension. As expressed in (15), wavelet kernel functions with different scale can compose a multi-scale wavelet kernel function which is shown as follows $(n = 1,2,...N)$.

$$\begin{aligned} K = &\prod_{i}^{n}[\cos(1.75\times(\frac{x_i - x_i'}{a_1})\exp(-\frac{(x_i - x_i')^2}{2a_1^2})] \\ &+\prod_{i}^{n}[\cos(1.75\times(\frac{x_i - x_i'}{a_2})\exp(-\frac{(x_i - x_i')^2}{2a_2^2})] \\ &+...+\prod_{i}^{n}[\cos(1.75\times(\frac{x_i - x_i'}{a_N})\exp(-\frac{(x_i - x_i')^2}{2a_N^2})] \end{aligned} \tag{16}$$

which can be abbreviated to

$$K = \sum_{l}^{L}\prod_{i}^{n}[\cos(1.75\times(\frac{x_i - x_i'}{a_l})\exp(-\frac{(x_i - x_i')^2}{2a_l^2})] \tag{17}$$

For binary classification, ELM only needs to set up two outputs, then the class of the sample through the competition mechanism. Moreover, we can adopt a single ELM classifier to solve multi-class problem. If the samples belong to n categories, the output of ELM needs to be set at n. The number of the maximum output value is the class of the sample.

IV. EXPERIMENTS

Ideally, a good classifier for BCI should produce high classification accuracy with minimal computational complexity. In this section, we evaluate the proposed classification method in accuracy and computation time, by comparing it with different classifiers on the same data set (BCI Competition Ⅲ dataset Ⅱ). As shown in Fig .2, the EEG signal processing mainly includes data collection, pre-processing, feature exaction and classification.

A. Data Set Description

The proposed multi-wavelet kernel ELM approach for the classification of the EEG signals is carried out on BCI Competition Ⅲ dataset Ⅱ (i.e.P300 speller BCI data) by MATLAB R2012a.

This dataset represents a complete record of P300 evoked potentials. The objective is to predict the correct character in each of the provided character selection epochs. In this P300 speller paradigm, a 6×6 matrix containing 36 symbols is presented to the subjects. As shown in Fig. 3, the row-column P300 speller paradigm is adopted, and the highlight row is the one intensified. For the spelling of each character, all rows and columns of this matrix are randomly intensified. The sets of 12 intensifications are repeated 15 times for each character epoch. The flashing of the rows or columns containing the target characters will evoke P300 of subjects.

For each of the two subjects, the 64-channel EEG signals are collected and digitized at 240Hz. The recorded EEG data contains one training set (85characters) and one test set (100 characters) for each subject.

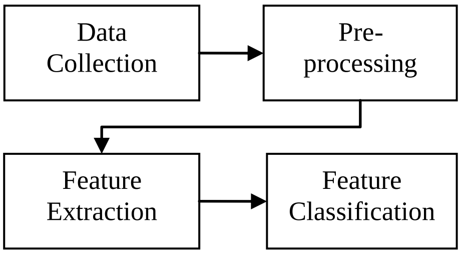
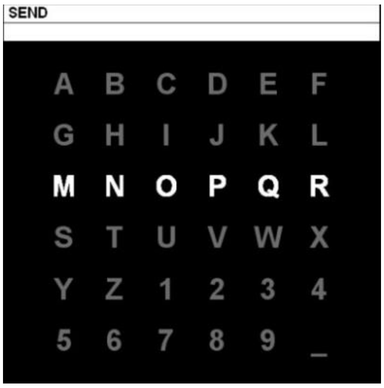


Fig. 2 The flowchart of EEG signal processing



SEND

A B C D E F
G H I J K L
M N O P Q R
S T U V W X
Y Z 1 2 3 4
5 6 7 8 9 _

Fig. 3 P300 speller paradigm

According to the process described in section 3, signals of the 10 selected channels are band-pass filtered from 0.5Hz to 30Hz.

For each channel, all data samples between 0 to 677 ms posterior to the beginning of an intensification are extracted. At this point, an extracted signal from a single channel is composed of 160 points. Thus, the training set is composed of 1020 (85×12) post-stimulus feature vectors and each feature vector contains 160 elements. The test set is composed of 1200 (100×12) feature vectors of 160 dimension.

B. Preprocessing

Raw EEG signals mixed with a lot of interference signals such as EOG and power line interference. Therefore preprocessing is necessary to construct high-level signal characteristics suitable for classification. The process comprises selection of electrodes, signal segmentation, superposition, filtering and data normalization.

For the time-locked assumption between the stimulus and the response [17], we took the values of the signal during 0-700ms (P300 is one of late positive component.) after stimulus onset from the electrode channels Fz, Cz, Pz, Oz, C3, C4, P3, P4, PO7, and PO8.

For low SNR of EEG signals, the repetitive stimulations are superimposed to reduce the interference signals and enhance the desired information. Moreover, since the frequency of P300 is mainly distributed in low frequency area, an 6-order band pass Chebyshev Type I filter which cut-off frequencies are 0.5 and 30 Hz is designed to filter each extracted signal.

C. Feature Extraction

The purpose of feature exaction progress is to find effective features to characterize the cognitive components. The exacted feature vectors of different tasks are expected to have obvious differences.

Since EEG signal is nonlinear, time-varying and non-stationary, traditional analysis methods cannot clearly distinguish the frequency components contained in a certain time range and some transient minutiae feature. We choose the discrete wavelet packet decomposition (WPD) which describes information in various time windows and frequency bands to extract EEG features [18].

WPD implements the equal width decomposition not only in the low frequency band but also in the high frequency band, which provides a more precise way to complex EEG signals. Based on that, each single epoch is decomposed into three levels by wavelet packet transform. Quadratic B-Spline functions are used as mother wavelets due to their similarity with the evoked responses. Eight sets of coefficient within the following frequency bands are obtained: 0.5-4Hz, 4-8Hz, 8-12Hz, 12-16Hz, 16-20Hz, 20-24Hz, 24-28Hz, and 28-30Hz.

Feature vectors are constructed by the wavelet packet energy and entropy of each node. The wavelet packet energy indicates the strength of the signal as it gives the area under the curve of power. The energy of EEG signal of finite length is given by (18).

$$En\left(s\right)=\sum_{i}s_{i}^{2} \tag{18}$$

where s_i represents the projection coefficients of signal s in an orthonormal basis. The energy feature of each epoch is:

$$\begin{aligned}&\mathbf{Enx}_{i}=\left[En_{ij}(s_{3}^{0}),En_{ij}(s_{3}^{1}),\cdots,En_{ij}(s_{3}^{7})\right],\\&i=1,2,\cdots,n,\ j=1,2,\cdots,10\end{aligned} \tag{19}$$

where n is the amount of epochs. j represents the selected 10 channels.

The wavelet packet entropy is calculated according to (20) to measure the complexity of EEG signals. The Shannon entropy is employed.

$$Ent(s)=-\sum_{i}s_{i}^{2}\log(s_{i}^{2}) \tag{20}$$

where s_i also represents the projection coefficients of signal s in an orthonormal basis. Therefore, the entropy feature vector of each epoch is:

$$\begin{aligned}&\mathbf{Entx}_{i}=\left[Ent_{ij}(s_{3}^{0}),Ent_{ij}(s_{3}^{1}),\cdots,Ent_{ij}(s_{3}^{7})\right],\\&i=1,2,\cdots n,\ j=1,2,\cdots,10\end{aligned} \tag{21}$$

Consequently, the feature vector of each epoch is constructed as following:

$$\mathbf{x}_{i}=\left[\mathbf{Enx}_{i},\mathbf{Entx}_{i}\right],i=1,2,\cdots,n$$

According to above procedure, a 160-dimensional feature vector for each epoch is extracted from EEG signals.

D. Classification Result

The classification performance of the proposed multi-scale wavelet ELM classifier is evaluated on the dataset described above. The number of hidden neurons is 1000. Three wavelet kernel functions are adopted. The corresponding parameters are set as a_1=1.38, a_2= a_1/10, a_3= a_1/100. C is set to 100.

The accuracy of predicted characters is used to evaluate the classification accuracy. Each target character is detected by the intersection of the row and the column containing P300.

Thus, the classifier is trained for binary classification, and the instances are labeled with “1”/“-1” for P300 presence/ absence. The maximum score of the discriminant function (17) indicates the presence of a P300.

Tables Ⅰ to Ⅲ show the results. Specifically, Table Ⅰ shows the spelling accuracy of our method on the test datasets of the two subjects with respect to the number of repetitions used in superposition. Table Ⅱ compares several effective methods with 15 repetitions. Table Ⅲ shows the training and testing time as different classifier is applied.

From the results, we can see that the proposed algorithm performs well in the recognition of P300. Table Ⅱ shows the recognition accuracies are almost the same when respectively using SVM, BP neural network (BPNN), ELM and the method proposed by us. However, Table Ⅲ shows that SVM spends the longest time to train the classifier, and the time consumed by BP network is over ten times longer than ELM and multi-scale wavelet kernel ELM. As shown in Table Ⅱ ,the performance of our algorithm is superior to the standard ELM algorithm and is comparable with it in terms of efficiency.

V. CONCLUSION

In this paper, by introducing multi-scale wavelet function into ELM as the kernel function, a kernel ELM based approach was proposed to address the cognitive component classification problem of EEG signals.

A comparative study on performance is conducted among different classifiers. Experiments on the BCI speller dataset shows the improved performance of the proposed algorithm comparing with original ELM. Moreover the proposed algorithm can achieve similar recognition accuracy with much less training time.

TABLE I. CLASSIFICATION PERFORMANCES IN % OF CORRECTLY RECOGNIZE CHARACTERS

Subject	Repetitions			
	1	***5***	***10***	***15***
A	20	50	89	96
B	31	59	93	97

TABLE II. COMPARISON OF CLASSIFICATION PERFORMANCES IN % OF CORRECTLY RECOGNIZED CHARACTERS

Subject	Classifier			
	SVM	***BPNN***	***ELM***	***Ours***
A	92	92	94	95
B	90	91	95	97

TABLE III. COMPARISON OF THE TRAINING AND TESTING TIME

Time(s)	Classifier			
	SVM	***BPNN***	***ELM***	***Ours***
Training	10.981	1.676	0.075	0.186
Testing	3.732	0.022	0.019	0.020

Since the multi-scale wavelet kernel function is the combination of multiple wavelet kernel functions with different scale, the choice of kernel parameters of it can greatly get relaxed, or even diluted. The overall implementation of the algorithm is easy to understand, and the computational burden is low. Moreover, the multi-scale wavelet kernel ELM has high approximation capacity, which makes it more convenient to use and has better recognition performance. In addition, if we consider practical applications, such as crime information identification based on EEG signals, the advantages of short training time and good generalization ability, the proposed method would have the ability to deal with the complex application environment and unpredictable interference. The related experiment is now in progress.

Nevertheless, there remains scope of this algorithm for improvement in various aspects such as data compression and feature selection. To make the algorithm be applied in practice, more experiments on training and testing time and other ERP components are required. In practice, since labeled data are usually insufficient for effective classification, a semi-supervised version of the algorithm should be considered.

REFERENCES

[1] Q. Yuan, W. Zhou, S. Li, “Epileptic EEG classification based on extreme learning machine and nonlinear features,” Epilepsy research, vol. 96, no. 1, pp. 29-38, 2011.

[2] L. Duan, H. Zhong, J. Miao, Z. Yang, W. Ma and X. A. Zhang, “Voting Optimized Strategy Based on ELM for Improving Classification of Motor Imagery BCI Data,” Cognitive Computation, Sensors & Transducers, vol. 169, no.4, pp. 235-240, 2014.

[3] B. H. Yang, L. Liu, P. Zan, and W.Y. Lu, “Wavelet Packet Based Feature Extraction for Brain-Computer Interfaces”, Int. Conf. Life System Modeling and Simulation (LSMS), pp. 19–26, 2010.

[4] S. GHelmy, T. Al-ani, Y. Hamam, et al. “P300 based brain-computer interface using Hidden Markov Models,” Int. Conf. Intelligent Sensors, Sensor Networks and Information Processing, IEEE, pp. 127-132, 2008.

[5] S. Yu wen, et al, “Programmable neural processing on a Smart dust for brain-computer interfaces,” IEEE Trans. Biomedical Circuits and Systems, vol. 4, no. 5, pp. 265-273, 2010.

[6] A. Yazdani, T. Ebrahimi, U. Hoffmann, “Classification of EEG signals using Dempster Shafer theory and a k-nearest neighbor classifier,” EMBS, pp. 327-330, 2009.

[7] U. Orhanu, M. Hekim, M. Ozer, “EEG signals classification using the K-means clustering and a multilayer perceptron neural network model,” Expert Systems with Applications, vol.38, no.10, pp.13475 — 13481, 2011.

[8] Alain Rakotomamonjy, Vincent Guigue, “BCI Competition III: Dataset II- Ensemble of SVMs for BCI P300 Speller,” IEEE Trans. Biomed. Eng, vol. 55, no.3, pp. 1147–1154, 2008.

[9] M. Kaper, P. Meinicke, U. Grossekathoefer, et al. “BCI competition 2003-data set IIb: Support vector machines for the P300 speller paradigm,” IEEE Trans. Biomed. Eng, vol. 51, no.6, pp. 1073-1076, 2004.

[10] G. B. Huang, Q. Y. Zhu, and C. K. Siew, “Extreme learning machine: theory and applications,” Neurocomputing, vol. 70, no. 1, pp. 489-501, 2006.

[11] L. Ch. Li, and B. L. Lu, “EEG-based vigilance estimation using extreme learning machines,” Neurocomputing, vol. 102, pp. 135-143, 2013.

[12] N. Y. Liang, P. Saratchandran, G. B. Huang, and N. Sundararajan, “Classification of mental tasks from EEG signals using extreme learning machine,” Int. J. Neural Systems, vol. 16, no. 1, pp. 29-38, 2006.

[13] G. B. Huang, D. H. Wang, and Y. Lan, “Extreme learning machines: a survey,” Int. J. Machine Learning and Cybernetics, vol .2, no. 2, pp. 107-122, 2011.

[14] G. B. Huang, H. Zhou, X. Ding, and R. Zhang, “Extreme learning machine for regression and multiclass classification,” IEEE Trans. Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 42, no. 2, pp. 513-529, 2012.

[15] P. L. Bartlett PL, “The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network,” IEEE Trans. Inf Theory, vol. 44, no. 2, pp. 525–536, 1998.

[16] L. Zhang, W. Zhou, and L. Ch. Jiao, “Wavelet support vector machine,” IEEE Trans. Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 34, no .1, pp. 34-39,2004.

[17] R. Q. Quiroga, “Quantitative analysis of EEG signals: time-frequency methods and chaos theory,” Institute of Physiology-Medical University Lubeck and Institute of Signal Processing-Medical University Lubeck, 1998.

[18] T. Wu, G.Z. Yan, B.H. Yang, H. Sun, “EEG Feature Extraction Based on Wavelet Packet Decomposition for Brain Computer Interface,” Measurement, vol.41, no. 6, pp. 618–625, 2008.