

An On-Line BCI System for Hand Movement Control Using Real-Time Recurrent Probabilistic Neural Network

Mohammad Ahmadi and Abbas Erfanian

Brain-Computer Interface Laboratory, Neural Technology Research Center, Department of Biomedical Engineering,
Iran University of Science and Technology (IUST)
Tehran, Iran
mahmadiamiri@ee.iust.ac.ir, erfanian@iust.ac.ir

Abstract—This paper presents a new EEG-based Brain-Computer Interface (BCI) for on-line controlling the hand movement in a virtual reality environment. The goal of this research is to develop an interaction technique that will allow the BCI to be effective in real-world scenarios for hand grasp control. For this purpose, two classifiers are designed. The first classifier which is based on the imagination of right-hand movement is for controlling the hand grasping, holding and opening. The second classifier, which is based on the imagination of left-hand movement is designed for error correction and activating the first classifier. One important issue in developing an on-line BCI is the robust and accurate classification of EEG signal which is characterized with a time-varying distribution. In this work, we present a real-time recurrent probabilistic neural network for classifying the EEG signals. The results show that the subjects were able to achieve an accuracy more than 80% during the first session of experiment without off-line training and 73%-91% during the last session using single-trial classification with no adaptation.

Keywords—on-line training; brain-computer interface (BCI); electroencephalogram (EEG); subject training; probabilistic neural network

I. INTRODUCTION

Although significant progress has been made in the area of brain-computer interface in recent years [1]-[2], there are still several issues that need to be addressed before BCIs can be used for real-world tasks. The BCI systems translate the brain activity into signals that control the external devices. Thus, event detection and classification of brain signals is an important issue in developing an EEG-based BCI. In this context, effective attempts have been done to improve the classification accuracy and capacity of the BCI systems [3]. However, robust and accurate EEG discrimination still remains a challenge in developing an on-line EEG-based BCI. The significant considerations in classifier design are computational complexity, generalization performance, fast convergence, and robustness to time varying environment.

EEG signal is a very complex and non-stationary signal which is characterized by significant day-to-day and subject-to-subject variations and time-varying probability

distributions. The measured values of EEG signal can be considered realizations of a random variable with a certain distribution. In this case, the pattern classification problem usually reduces to the construction of a model that estimates the class conditional densities $p(x|k)$ of the data and the respective prior probabilities $p(k)$ for each class k . Then, using Bayes' theorem, the posterior probabilities $p(x|k)$ can be computed

$$p(x|k) = \frac{p(x|k)p(k)}{\sum_l p(x|l)p(l)} \quad (1)$$

In order to classify an unknown pattern x , we select the class with the highest posterior probability $p(x|k)$ as suggested by the Bayes' rule. The accuracy of probabilistic classification relies on the accuracy of the probability density function (pdf) estimation, which can be obtained by parametric, nonparametric, or semiparametric methods. Parametric approaches are easy to implement, but the assumed pdf may not always match the original data distribution very well. In the nonparametric approach, it is assumed that a functional form of probability densities is unknown. The semiparametric estimation of the pdf, having a flexible structure, can represent any distribution and include a set of parameters for particular distributions. The unknown distribution is defined as a weighted sum of a number of component distributions (e.g., Gaussian distribution). Some researchers have proposed semiparametric estimation of the pdf using artificial neural networks [4]. Particularly, Tsuji *et al.* [5] proposed a recurrent log-linearized Gaussian mixture network (R-LLGMN), which is based on the hidden Markov model (HMM). The backpropagation-through-time (BPTT) algorithm [6] was used to train the network. Obviously, the epoch-wise back-propagation-through-time algorithm is not suitable for the real-time operation of a recurrent network (e.g., R-LLGMN). In this paper, we introduce a real-time recurrent learning algorithm for R-LLGMN and use it for on-line classification of EEG signals during imagination of hand movement.

II. METHODS

A. Recurrent Log Linearized Guassian Mixture Network (R-LLGMN)

The R-LLGMN proposed by Tsuji *et al.* [5] is a neural network which is based on a hidden Markov model (HMM). The details of R-LLGMN can be found in [5]. Let us consider a HMM with C classes where the class c is composed of K_c states. According to this model, given a time series $\hat{x}(t)$, the *a posteriori* probability for class c , $p(c | \hat{x})$, is given by

$$P(c | \hat{x}) = \sum_{k=1}^{K_c} P(c, k | \hat{x}) = \sum_{k=1}^{K_c} \frac{\alpha_T^c(k)}{\sum_{c=1}^C \sum_{k'=1}^{K_c} \alpha_T^{c'}(k')} \quad (1)$$

Here, $\alpha_t^c(k)$ is the forward variable, which is defined as the probability for partial time series $(x(1), x(2), \dots, x(t))$ to be generated from class c and input vector $x(t)$ occurs from state k in class c , $\alpha_t^c(k)$ can be computed according the forward algorithm as follows:

$$\alpha_1^c(k) = \pi_k^c b_k^c(x(1)) \quad (2)$$

$$\alpha_t^c(k) = \sum_{k'} \alpha_{t-1}^{c'}(k') \gamma_{k',k}^{c'} b_k^c(x(t)), \quad 1 < t \leq T \quad (3)$$

where $\gamma_{k',k}^{c'}$ is the probability of state changing from k' to k in class c' , and $b_k^c(x(t))$ is defined as the *a posteriori* probability for state k in class c corresponding to $x(t)$. The observation probability of state in class c is approximated with Gaussian mixture model (GMM). Using GMM, the right side of (2) can be approximated as

$$\gamma_{k',k}^{c'} b_k^c(x(t)) = \sum_{m=1}^{M_{c,k}} \gamma_{k',k}^{c'} r_{c,k,m} g(x(t); \mu^{(c,k,m)}, \Sigma^{(c,k,m)}) \quad (4)$$

where $r_{c,k,m}$, $\mu^{(c,k,m)}$ and $\Sigma^{(c,k,m)}$ are the mixing proportion, the mean vector, and the covariance matrix of each component (c, k, m) . Applying the *log-linearization* to (4), we obtain

$$\begin{aligned} \xi_{k',k,m}^{c'}(t) &\triangleq \log \gamma_{k',k}^{c'} r_{c,k,m} g(x(t); \mu^{(c,k,m)}, \Sigma^{(c,k,m)}) \\ &= \beta_{k',k,m}^{c'} T X(t) \end{aligned} \quad (5)$$

where $X(t)$ is defined as

$$X(t) = (1, x^T, x_1^2, x_1 x_2, \dots, x_1 x_d, x_2^2, x_2 x_3, \dots, x_2 x_d, \dots, x_d^2) \quad (6)$$

The model (4) can be developed by the neural network structure in which $\beta_{k',k,m}^{c'}$ is used as the weight coefficients of the network. To satisfy the statistical constraints of the elements $\beta_{k',k,m}^{c'}$ during training the network, the new coefficient vector $w_{k',k,m}^{c'}$ are defined as follows

$$w_{k',k,m}^{c'} = \beta_{k',k,m}^{c'} - \beta_{K_c, K_c, M_{c,K_c}}^{c'} \quad (7)$$

Subsequently, (3) can be written as

$$\alpha_t^c(k) = \sum_{k'=1}^{K_c} \alpha_{t-1}^{c'}(k') \exp \left[w_{k',k,m}^{c'} T X(t) \right] \quad (8)$$

Tsuji *et al.* [5] transformed this model to a recurrent neural network with five layers and used backpropagation-through-time (BPTT) algorithm for training the network. In this work, we introduce a real-time learning algorithm for R-LLGMN.

B. Real-Time Learning Algorithm for R-LLGMN

The real-time recurrent learning (RTRL) algorithm is one of the successful learning algorithms where the gradient of errors is propagated forward in time rather than backward in time. Therefore, it is well suited for on-line training of recurrent neural networks.

The RTRL algorithm is based on the *instantaneous energy function* as

$$J(t) = - \sum_{c=1}^C T_c(t) \log^{(5)} O^c(t) \quad (9)$$

where $T_c(t)$ denote the target response of the output neuron c at time t and $^{(5)}O^c(t)$ is the output of neuron c in the fifth layer. The objective is to adapt the weight coefficients $w_{k',k,m}^{c'}$ so as to minimize $J(t)$, that is, to maximize the likelihood that teacher vector $T(t)$ is obtained for the input vector $x(t)$. To accomplish this objective we may use the method of steepest descent as follows

$$\Delta w_{k',k',m,h}^{c'} = -\eta \frac{\partial J(t)}{\partial w_{k',k',m,h}^{c'}} \quad (10)$$

$$\frac{\partial J(t)}{\partial w_{k',k',m,h}^{c'}} = \sum_{c'=1}^C \sum_{k'=1}^{K_c} \frac{\partial (T_{c'} \log^{(5)} O^{c'}(t))}{\partial ^{(5)} O^{c'}(t)} \times \frac{\partial ^{(5)} O^{c'}(t)}{\partial ^{(4)} O_{k'}^{c'}(t)} \times \frac{\partial ^{(4)} O_{k'}^{c'}(t)}{\partial w_{k',k',m,h}^{c'}} \quad (11)$$

$$\frac{\partial J(t)}{\partial w_{k',k',m,h}^{c'}} = \sum_{c'=1}^C \sum_{k'=1}^{K_c} \frac{T_{c'}}{^{(5)}O^{c'}(t)} \times \pi_{k',k',m,h}^{c'}(t) \quad (12)$$

$$\pi_{k',k',m,h}^{c'}(t) = \left(\Gamma_{(c',k'),(c,k)} - ^{(4)}O_{k',k'}^{c'}(t) \right) \times \frac{^{(4)}O_{k'}^{c'}(t)}{^{(4)}I_{k'}^{c'}(t)} \quad (13)$$

$$\times \left(\pi_{k',k',m,h}^{c'}(t-1) \times ^{(3)}I_{k',k'}^{c'}(t) + \right. \\ \left. \times \left(^{(2)}O_{k',k',m}^{c'}(t) \times X_h(t) \times ^{(4)}O_{k'}^{c'}(t-1) \right) \right)$$

$$\Gamma_{(c',k'),(c,k)} = \begin{cases} 0, & (c = c', k = k') \\ 1, & (\text{otherwise}) \end{cases} \quad (14)$$

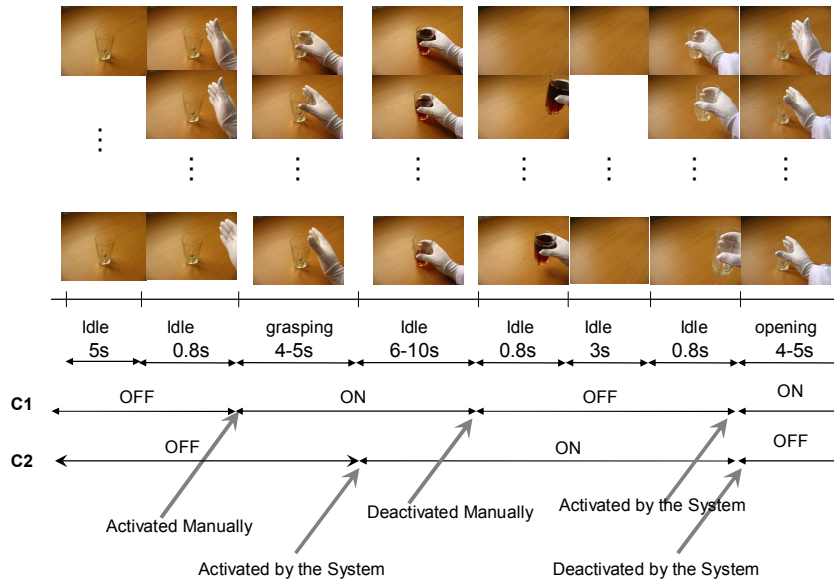


Fig. 1. The structure of a typical trial of experiment for hand movement control.

III. EXPERIMENTAL SETUP

A. Subjects and Recording

The experiments were carried out with two able-bodied volunteer subjects. EEG signals were recorded at a sampling rate of 256 from positions F3, F4, C3, and C4 by Ag/AgCl scalp electrodes placed according to the International 10-20 system and then were lowpass filtered (cutoff frequency 45 Hz). The eye blinks were recorded by placing an electrode on the forehead above the left eyebrow line. All recording channels were referenced to the right earlobe. The signals were continuously collected and processed during the experiments, while the subject was free to blink and to move his eyes. The experiment consisted of 8 sessions for each subject. Each session was conducted on a different day and consisted of 10 runs. Each run consisted of at least 3 feedback trials. A resting period of about 2 minutes was enforced between each run.

B. Experimental Paradigm

The experiment was based on an interactive virtual reality environment. The subjects sat on a relaxing chair with armrests. At the start of trial, an empty glass was shown to the subject for 5 s. After 5 s, an open hand was beginning to reach the glass. After reaching the hand to the glass, at 5.8 s, an active feedback phase lasting 5 s was started in which the user should try to grasp the glass by imagination of right-hand grasping. Upon the detection of motor imagery by the first classifier (C1), the hand will be closed sequentially. The sequence of closing was controlled by the output of classifier. Following the grasping, the second classifier (C2) was activated and the subject should try to keep the hand closed until the glass filled by a liquid (i.e. relaxation phase). This is the idle state in which the subject does not perform any specific mental task. During the relaxation phase, if an error was made by the interface (a misinterpretation of a command that the user has given, i.e.,

detecting the right-hand imagination by the first classifier and opening the hand), the subject should try to correct this error and close the hand by imagination of left-hand grasping which is detected by the second classifier (C2). Upon detecting the imagination of the left-hand grasping by the C2, the hand will be closed.

Following the relaxation phase, the C1 was deactivated, but the C2 remained active. At this time, the subject is free to move his right-hand. Upon detection the imagination of the left-hand movement, the C1 was activated and the C2 was deactivated. The subject could open the hand by imagination of the right-hand movement. Fig. 1 shows the structure of a typical trial for hand movement control.

C. Real-time ocular artifact suppression

One of the major problems in developing an online EEG-based BCI is the ocular artifact suppression. In this work, during the online experiments, eye blink artifacts were suppressed automatically by using a neural adaptive noise canceller [7]. The primary signal was the measured EEG data from one of the EEG channels. The reference signal was the data recorded from the forehead electrode. Here the adaptive filter was implemented by means of a multi layer perceptron neural network. The detail can be found in [7].

D. Hardware and Software

In our case, we used Matlab Simulink under Windows XP for on-line data acquisition, filtering and ocular artifact suppression, feature extraction, classification and providing interactive virtual reality environments. The EEG was recorded with a bipolar EEG-amplifier (g.USBamp, g.tec, Guger Technologies, Graz, Austria).

IV. RESULTS

The EEG data was continuously recorded and filtered and the eye blink artifacts were removed online during each

run of experiment. At each point of EEG data, the features were extracted from 1-s sliding window and classified. The feature set was formed from the spectral power of EEG signal of two selected EEG channels in theta, lower alpha, upper alpha, lower beta, upper beta, and gamma frequency bands. The control signal was generated every 1 s. To form the control signal, the results of classifier within every 1 s was mediated by a simple logical majority vote function.

Two schemes of classification process were used here for virtual hand grasp control: adaptive and static classification. In adaptive scheme, the classifier was continuously updated while the static classifier was not. Adaptive scheme was used to train the classifier on-line during the first sessions.

Fig. 2 shows the results of on-line hand grasp control during different runs of the first and second sessions for two subjects where the adaptive classifier was used for all runs of the session. It is observed that the subject SP was able to reach a classification accuracy rate of 81% and the subject AS 83% after 3 runs of experiment without off-line training during the first session. The average accuracy rate obtained during the first session of experiment is 80.3% for the subject SP and 82.4% for the subject AS. The subject SP was able to reach an accuracy of 100% during the second session. The average accuracy is 90% during the second session of experiment.

Fig. 3 shows the average performance over all runs during each session for all subjects using both adaptive and static classification schemes. It is observed that the performance of static scheme is almost the same as that of adaptive. In subject SP, it is observed that the performance of BCI decreases from 89.8% to 81.0% when the classifier was switched from adaptive to static in the third session and begins to increase during next experimental sessions. During the sixth session, when the classifier was switched from static to adaptive mode, the performance increases from 84.7% to 87.2% and remains the same for the subsequent experiment sessions during static mode of classifier operation. The same results were obtained for the subject AS.

V. DISCUSSION AND CONCLUSIONS

During the present BCI experiment which is based on an interactive virtual reality environment, the subjects received feedback from beginning the experiments without any pre-training. The subjects could achieve an average accuracy of 81.4% during the first experimental session and 85.5% during the second session, while the adaptive classification was used. During the last session, an average accuracy of 83.6% over two subjects was obtained using classification with no adaptation.

In this paper, we addressed several major issues including subject training, machine learning, and error correction in designing an online BCI system for real-world tasks. One important issue for online BCIs is the self-initiation of the system without assistance from others which remains to be

addressed before the system can be used for controlling the artificial devices.

REFERENCES

- [1] P. S. Hammon and V. R. de Sa, "Preprocessing and meta-classification for brain-computer interfaces," *IEEE Trans. Biomed. Eng.*, vol. 54, no. 3, pp. 518-525, March 2007.
- [2] G. Dornhege, B. Blankertz, M. Krauledat, F. Losch, G. Curio and K. R. Mueller, "Combined optimization of spatial and temporal filters for improving brain-computer interfacing," *IEEE Trans. Biomed. Eng.*, vol. 53, no. 11, pp. 2274-2281, Nov. 2006.
- [3] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, B. Arnaldi, "A review of classification algorithms for EEG-based brain-computer interfaces," *Neural Engineering*, vol. 4, pp.1-13, 2007.
- [4] H. G. C. Trávén, "A neural network approach to statistical pattern classification by 'semiparametric' estimation of probability density functions," *IEEE Trans. Neural Networks*, vol 2, pp. 366-377, May 1997.
- [5] T. Tsuji, N. Bu, O. Fukuda and M. Kaneko, "A recurrent log-linearized gaussian mixture network," *IEEE Trans. Neural Networks*, vol. 14, pp. 304-316, Mar. 2003.
- [6] S. Haykin, *Neural Networks: A Comprehensive Foundation*, New York: Macmillan, 1994.
- [7] A. Erfanian and B. Mahmoudi, "Real-time ocular artifact suppression using recurrent neural network for electro-encephalogram based brain-computer interface," *Med. Biol. Eng. Comput.*, vol. 43, pp. 296-305, 2005.

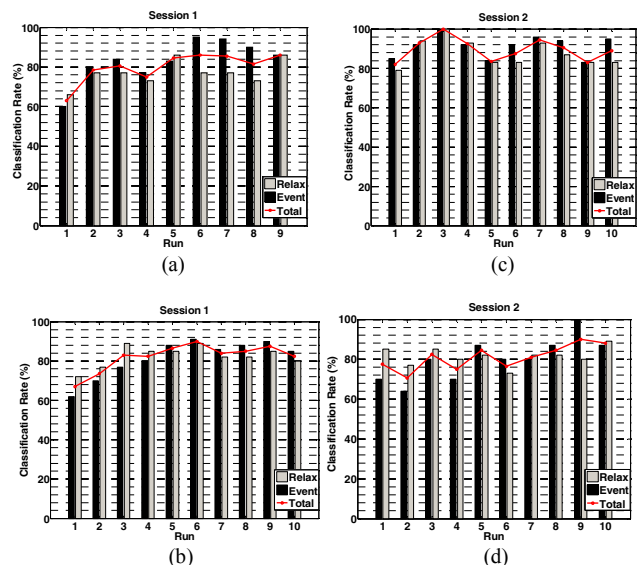


Fig. 2. Classification accuracies obtained during different runs of the first (left) and second session (right) for the subjects SP (a), (c) and AS (b), (d).

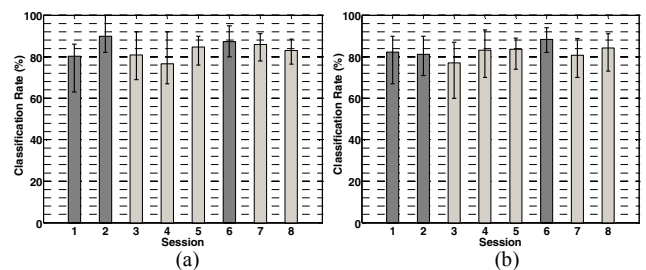


Fig. 3. The average performance obtained during each session for the subjects SP (a) and AS (b), using both adaptive (dark gray) and static (light gray) classification.