[6] B. Stec and A. Dobrowolski, "Estimation of internal distribution of temperature inside biological tissues by means of multifrequency microwave thermograph," in *Proc.13th Int. Conf. Microwaves, Radar and Wireless Communications*, Wroclaw, Poland, 2000, pp. 577–580.

[7] ——, "Estimation of internal distribution of temperature inside biological tissues by means of multifrequency microwave thermograph," *J. Telecommun. Inform. Technol.*, no. 1/2002, pp. 39–42, 2002.

[8] B. Stec, A. Dobrowolski, and W. Susek, "Estimation of temperature distribution inside biological tissues by means of multifrequency microwave thermograph," presented at the 23rd Annu. Int. Conf. IEEE Engineering in Medicine and Biology Society, Istanbul, Turkey, 2001.

[9] ——, "Estimation of deep-seated profile of temperature distribution inside biological tissues by means of multifrequency microwave thermograph," in *IEEE Microwave Theory Tech.-S Int. Microwave Symp.*, Seattle, WA, 2002, pp. 2261–2264.

[10] W. Kazanowska, P. Knapp, A. Mazurek, T. Filipkowski, and B. Stec, "An attempt to employ microwave thermography for the evaluation of potential danger of breast cancer," presented at the 8th Int. Meet. Gynaecological Oncology, Barcelona, Spain, 1993.

[11] T. Maeda, "Diagnosis of mammary gland diseases using microwave thermography: A study on adjunctive diagnostic methods in relation to different factors in breast diseases," *Nippon-Geka-Gakkai-Zasshi*, vol. 91, no. 5, pp. 622–630, 1990.

Massively Parallel Classification of Single-Trial EEG Signals Using a Min-Max Modular Neural Network

Bao-Liang Lu*, Jonghan Shin, and Michinori Ichikawa

***Abstract*—This paper presents a method for classifying single-trial electroencephalogram (EEG) signals using min-max modular neural networks implemented in a massively parallel way. The method has three main steps. First, a large-scale, complex EEG classification problem is simply divided into a reasonable number of two-class subproblems, as small as needed. Second, the two-class subproblems are simply learned by individual smaller network modules in parallel. Finally, all the individual trained network modules are integrated into a hierarchical, parallel, and modular classifier according to two module combination laws. To demonstrate the effectiveness of the method, we perform simulations on fifteen different four-class EEG classification tasks, each of which consists of 1491 training and 636 test data. These EEG classification tasks were created using a set of non-averaged, single-trial hippocampal EEG signals recorded from rats; the features of the EEG signals are extracted using wavelet transform techniques. The experimental results indicate that the proposed method has several attractive features. 1) The method is appreciably faster than the existing approach that is based on conventional multilayer perceptrons. 2) Complete learning of complex EEG classification problems can be easily realized, and better generalization performance can be achieved. 3) The method scales up to large-scale, complex EEG classification problems.**

***Index Terms*—Brain-computer interface, classification of EEG, min-max modular neural network, module combination, parallel learning, single-trial EEG, task decomposition, wavelet transform.**

I. Introduction

The electroencephalogram (EEG) is one of the most important sources of information used to study brain functions [1], to pinpoint the origins of some neurological disorders [2], and to recognize mental tasks for brain-computer interfaces (BCIs) [3]–[6]. Recent developments in hardware and software designed for acquiring EEG signals make the recording of high-resolution EEGs from large electrode arrays feasible. In a typical dense-array EEG experiment, large amounts of digitized EEG data are generated. Therefore, to be analytically tractable, more efficient data analysis and classification methods are required for automatically processing large amounts of highly complex EEG data.

In recent years, artificial neural networks, such as multilayer perceptrons (MLPs) [7], cascade-correlation neural networks [8], and recurrent neural networks [9], have been applied to the classification of EEG signals. These are clearly recognized as useful tools one might use for this problem [10]. While most neural networks successfully process small and simple tasks, few of them successfully deal with large amounts of high-dimensional EEG data. The reason is that no efficient learning algorithm has been available for training large-scale neural networks, and with previous algorithms, satisfactory learning and generalization accuracy could not be consistently achieved, even when very long computing times were used. To overcome this problem, almost all existing methods typically use just a few features extracted from the EEG signals as inputs [11], or use a small number of training data [4]. Although feature extraction with various techniques such as wavelet transform is necessary for successfully classifying EEG signals, greatly condensing the number of features for neural network inputs means that much useful information in the original EEG signals will be necessarily lost, and generalization accuracy of the trained networks will be significantly degraded.

In this paper, we present a method implemented in a massively parallel neural network for efficiently classifying high-dimensional, single-trial EEG signals. The method is based on a min-max modular (M^3) neural network model, an alternative committee machine proposed in our previous work [12], [13]. The method has several advantages over existing methods based on traditional neural networks. 1) A large-scale, complex EEG classification problem can be simply divided into a number of two-class subproblems, as small as needed. 2) All the two-class subproblems can be simply learned by individual smaller network modules in parallel, and therefore, a large set of high-dimensional EEG data can be learned efficiently. 3) The classifiers constructed by the method behave better in generalization performance than those of the existing approaches.

The remainder of the article is organized as follows. In Section II, we describe the experimental setup and the EEG classification task. In Section III, we introduce wavelet transform techniques used for extracting input features from nonaveraged, single-trial EEG signals. In Section IV, a classification method is descirbed which is based on a min-max modular neural network model. The simulation results on fifteen different EEG classification problems are presented in Section V, and the performance of our new method is compared with that of a traditional approach in Section VI. Finally, Section VII comprises the discussion.

II. Hippocampal EEG Signals

Particular hippocampal EEG signals are associated with particular cognitive process and behaviors, such as attention, learning, and voluntary movement [1]. The hippocampal EEG signals used in the present study were recorded from eight adult male hooded rats weighting between 300 and 400 g. They were housed in individual cages with free access to food and water until the behavioral training began. One week after surgical implantation of hippocampal electrodes, the rats were

Manuscript received August 4, 2002; revised August 10, 2003. *Asterisk indicates corresponding author.*

*B. L. Lu is with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, 1954 Hua Shan Rd., Shanghai 200030, P. R. China.

J. Shin is with the Division of Life Science, Korea Institute of Science and Technology, Seoul, South Korea.

M. Ichikawa is with the Laboratory for Brain-Operative Device, RIKEN Brain Science Institute, 2-1 Hirosawa, Wako-shi, Saitama, 351-0198, Japan.

Digital Object Identifier 10.1109/TBME.2003.821023

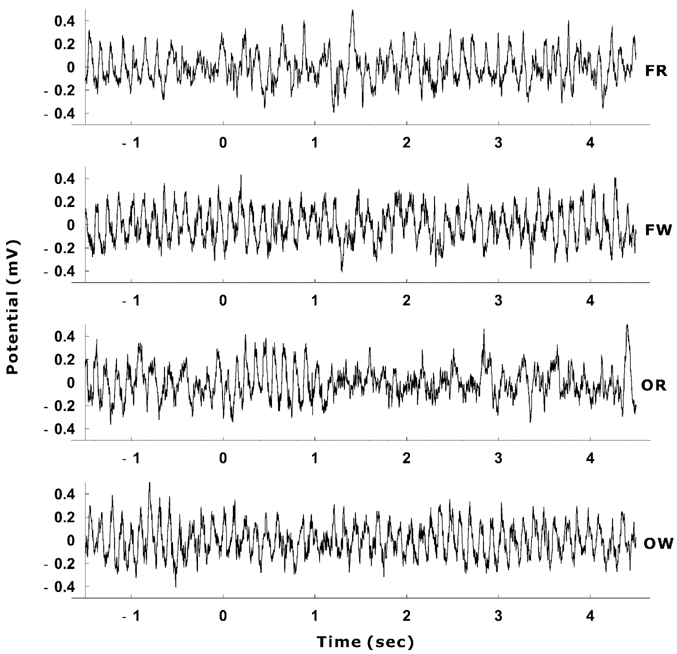

Fig. 1. Four nonaveraged, single-trial hippocampal EEG signals recorded during the FW, FR, OR, and OW trials of the "oddball" task. Here, the exact time of 0 s is when a stimulus is presented and the stimulus last 50 ms.

water-deprived and trained in an oddball paradigm [14], in which occasional "target" stimuli have to be detected in a train of frequent "nontarget" stimuli. All training occurred in a standard behavioral training chamber. We used a low-frequency tone ("odd" tone) as "target" stimuli and a high-frequency tone ("frequent" tone) as "nontarget" stimuli. The animals were rewarded by water whenever they correctly discriminated "target" from "nontarget" tones and crossed a light beam in front of the water tube.

A total of 2127 nonaveraged, single-trial hippocampal EEG signals were selected for classification analysis from a data set recorded in rats trained to discriminate target from nontarget stimuli. Here, we discarded the following two kinds of single trials: 1) EEG signals, during which a movement artifact occurred, and 2) EEG signals, during which a behavioral state transition occured (e.g., from waking state to drowsiness). EEG epochs were 6 s in duration and belonged to one of four classes defined by the target type and animal's response: FR, FW, OR, and OW. "FR" refers to a trial in which a frequent tone was presented, and the rat performed the correct behavior (no go); "FW" refers to a frequent tone and incorrect behavior (go); "OR" refers to an odd tone and correct behavior (go); and "OW" refers an odd tone and incorrect behavior (no go). Fig. 1 illustrates four nonaveraged, single-trial EEG signals recorded during FR, FW, OR, and OW trials, respectively. In the simulations described below, we used 1491 EEG signals for training and the remaining 636 EEG signals for testing. For simplicity of description, the four classes, FR, FW, OR, and OW, are represented as $\mathcal{C}_1$, $\mathcal{C}_2$, $\mathcal{C}_3$, and $\mathcal{C}_4$, respectively, throughout the paper. Table I shows the distributions of the training and test data.

III. Feature Extraction With Wavelet Transform

In order to quantify changes in single-trial hippocampal EEG signals in both frequency and time, we used wavelet transform techniques [15] to extract the features of EEG signals. The original EEG signals were convolved using the Morlet wavelet $w(t, w_o)$ with a Gaussian shape,

TABLE I
Distributions of the Training and Test Data

Class	No. data	
	Training	Test
Frequent tone/NO-Go (FR)	1027	430
Frequent tone/GO (FW)	136	68
Odd tone/GO (OR)	307	128
Odd tone/No-GO (OW)	21	10
Total	1491	636

TABLE II
Two Scenarios for Extracting Features From Nonaveraged, Single-Trial EEG Signals

Case	Duration (sec)	No. samples in time	Bandwidth (Hz)	No. wavelet coefficients	No. features
$\mathcal{A}$	-1.5 $\sim$ 4.5	400	5 $\sim$ 12	5	2000
$\mathcal{B}$	-1.5 $\sim$ 4.5	1200	3.03 $\sim$17.11	11	13200

both in the time domain and in the frequency domain around its central frequency w_o:

$$W(t, w_o) = \exp\left(jw_0 t - \frac{t^2}{2}\right). \quad (1)$$

These wavelets can be compressed by the scale factor a and shifted in time by the parameter b. Convolving the signal and the shifted, dilated wavelet leads to a new signal,

$$S_a(b) = \frac{1}{\sqrt{a}} \int W\left(\frac{t-b}{a}\right) x(t)dt \quad (2)$$

where W is the conjugate of the complex wavelet and $x(t)$ is the hippocampal EEG signal.

The new signals $S_a(b)$ are computed for different scaling factors a. It is known that hippocampal theta rhythm (5–12 Hz) is a sinusoidal-like EEG signal and has been related to arousal, attention, stimulus evaluation, learning and memory, and voluntary movements [1], [14]. To study the relationship between the generalization accuracy of trained pattern classifiers and hippocampal EEG frequency bandwidths, two different frequency bandwidths were selected, one from 5 to 12 Hz and the other from 3.03 to 17.11 Hz. For these two frequency bandwidths, each of the 2127 nonaveraged, single-trial EEG 6-s epochs was down-sampled from 12 000 samples to 400 and 1200 samples over the duration of $-1.5 \sim 4.5$ s according to the sampling theorem. The numbers of the corresponding wavelet coefficients were 5 and 11, respectively, which were selected based on corresponding frequency bands. Here, the 5-wavelet coefficients were selected by corresponding theta band (5 – 12 Hz) and the 11-wavelet coefficients were selected by the extended frequency band (3.03 – 17.11 Hz). Table II shows the related parameters used for extracting the features of the EEG signals in two different cases.

IV. Method

The proposed method is based on a min-max modular neural network model [12], [13]. The method has three main steps [16]. First, a large-scale, complex EEG classification problem is decomposed into a number of relatively smaller and simpler two-class classification subproblems, according to the class relationships among the given training data. Second, all the two-class subproblems are learned by smaller network modules in a massively parallel way. Finally, following two module combination laws, all of the individual trained network modules are integrated into a hierarchical, parallel, and modular pattern classifier that produces solutions to the original EEG classification problem.

A. Task Decomposition

For human beings, the only way to solve a complex problem is to divide it into smaller, more manageable subproblems. Breaking up a problem helps human beings deal with complex issues involved in its solution. This "divide-and-conquer" strategy is also helpful to neural networks in complex learning problems.

We have suggested that a K-class problem can be divided into $K(K-1)$ relatively smaller two-class subproblems [12], [13]. We have also shown that among these $K(K-1)$ two-class subproblems, only $\binom{K}{2}$ two-class subproblems need to be learned, and the remaining $\binom{K}{2}$ two-class subproblems are the same as the former ones from the point of view of pattern classification.

Although each of the $\binom{K}{2}$ two-class subproblems are smaller than the original K-class problem, this partition may not be adequate for parallel computation and fast learning due to the following reasons. 1) Some of the two-class subproblems might fall into a "load imbalance" situation. Since the speed of parallel learning is limited by the speed of the slowest subproblem, the undue burdening of even a single subproblem can dramatically degrade the overall performance of learning. 2) Some of the two-class subproblems might still be too large for learning. 3) Some of the two-class subproblems might be imbalanced in terms of types of examples in the training set, i.e., the training set contains many more data of the "dominant" class than the other "subordinate" class. For example, the standard back-propagation algorithm converges very slowly for imbalanced problems [17]. To speed up learning, all the large and imbalanced two-class subproblems are further divided into relatively smaller and more balanced two-class subproblems.

One of the most important features of the task decomposition method used is simple and straightforward, and neither domain specialization nor prior knowledge of the problem is required. Therefore, any user can perform this decomposition and divide a large K-class problem into many two-class subproblems, as small as needed.

B. Massively Parallel Learning

After task decomposition, each of the two-class subproblems can be treated as a completely independent, noncommunicating problem in the learning phase. Therefore, all the two-class subproblems can be efficiently learned in a massively parallel way. This is an ideal case, which is called *completely parallelizable* in the parallel computing literature [18].

In comparison with directly tackling a K-class problem, there are several merits in solving the corresponding smaller and simpler two-class subproblems. 1) Training time can be drastically reduced. From our experience [13], [19], the more complex the K-class problem, the greater the speedup rate that is achieved. 2) The two-class subproblems can be easily solved using various network models, learning algorithms, and a wide variety of computing resources from personal computers to supercomputers, since both the size and complexity of each two-class subproblem are much more manageable than the original K-class problems. 3) Complete learning of a complex K-class problem can be easily achieved because the size of each of the two-class subproblems can be reduced as much as needed, and the learning convergence of each of the two-class subproblems can be guaranteed.

In addition, the proposed method has two main advantages over traditional parallel implementations of neural network paradigms [20]. 1) Massively parallel learning can be easily implemented not only on supercomputers, but also on virtual supercomputers and many individual serial computers. 2) Existing neural network models, learning algorithms, and software packages can be directly used to construct the M^3 networks, and no specific programming techniques are required.

C. Module Combination

After training individual network modules assigned to learn associated two-class subproblems, all the trained network modules are integrated into a M^3 network with the MIN, MAX, or/and INV units according to two module combination laws [12], [13], [21], namely the *minimization* principle and the *maximization* principle.

Since the module combination procedure is completely independent of both the structure of individual trained network modules and their performance, we can easily replace any trained network modules with desired ones to achieve better generalization performance. In contrast to the task decomposition procedure mentioned earlier, the module combination procedure proceeds in a bottom-up manner. The smaller network modules are integrated into larger network modules first, and then the larger network modules are integrated into a M^3 network. After finishing module combination, the solutions to the original K-class problem can be obtained from the outputs of the entire M^3 network.

V. Classification of EEG Signals

In this section, we demonstrate how to use the proposed method to efficiently classify nonaveraged, single-trial EEG signals in a massively parallel way. To show the effectiveness of the method and study the relationship among generalization accuracy of trained pattern classifiers, hippocampal EEG frequency bandwidths, and duration of EEG signals, we performed computer simulations on fifteen different EEG classification problems ("EEG problems" for short), which were created using different frequency bandwidths and durations. In these simulations, all fifteen EEG problems had the same number of training and test data; the only difference among them was the number of input features.

A. Parameter Choices

In the simulations presented below, all the network modules in the M^3 networks were chosen to be MLPs with one hidden layer. To compare the performance of the proposed method with that of an existing approach based on conventional MLPs, the original EEG problems were also learned by single MLPs. All the MLPs used in both the proposed method and the existing approach were trained using the same back-propagation algorithm [22]. The momentums of the back-propagation algorithm were all set to 0.9. The learning rates were set to 0.1, 0.12, or 0.15. To guard against bias caused by different local minimums, every simulation was performed three times with different initial weights generated randomly. In the process of training the network modules for the M^3 networks, learning is stopped when the sum-of-squares error between the desired and actual outputs was less than 0.05 or when the number of epochs reached 5000. All of the simulations were performed on a Fujitsu VPP700E vector parallel computer.

B. Decomposition of EEG Problems

Each of the fifteen EEG problems was divided into $\binom{4}{2} = 6$ two-class subproblems, namely $\mathcal{T}_{1,2}$, $\mathcal{T}_{1,3}$, $\mathcal{T}_{1,4}$, $\mathcal{T}_{2,3}$, $\mathcal{T}_{2,4}$, and $\mathcal{T}_{3,4}$. The number of positive and negative training data belonging to each of the six two-class subproblems is shown in Table III. From this table, one can see that the number of training data for the smallest two-class subproblem $\mathcal{T}_{2,4}$ is 157, the number of training data for the largest two-class subproblem $\mathcal{T}_{1,3}$ is 1334, and all the two-class subproblems are *imbalanced* problems.

TABLE III
NUMBER OF TRAINING DATA USED FOR EACH OF THE SIX TWO-CLASS SUBPROBLEMS

Task	No. positive data	No. negative data	Total
$\mathcal{T}_{12}$	1027	136	1163
$\mathcal{T}_{13}$	1027	307	1334
$\mathcal{T}_{14}$	1027	21	1048
$\mathcal{T}_{23}$	136	307	443
$\mathcal{T}_{24}$	136	21	157
$\mathcal{T}_{34}$	307	21	348

TABLE IV
TWO WAYS OF PARTITIONING 15 DIFFERENT EEG PROBLEMS

Case	N_1	N_2	N_3	N_4	Max	Total
a	49	6	15	1	44	1189
b	30	4	9	1	70	469

TABLE V
SIX TRAINING DATA SETS EXTRACTED FROM DIFFERENT PERIODS WITHIN THE SAME 5–12 HZ BANDWIDTH AND CORRESPONDING GENERALIZATION PERFORMANCE OF THE M^3 NETWORKS. CORRECT RATES ARE MEAN (TOP ROW) AND STANDARD DERIVATIONS (BOTTOM ROW) FOR THREE SIMULATIONS

Data set	$\mathcal{A}_1$	$\mathcal{A}_2$	$\mathcal{A}_3$	$\mathcal{A}_4$	$\mathcal{A}_5$	$\mathcal{A}_6$
Duration (sec)	0~0.225	0.3~0.6	0~0.5	0~0.6	0~0.9	-1.5~4.5
No. features	75	100	165	200	300	2000
Correct rate	54.09	70.28	68.92	71.59	75.21	80.92
on test data (%)	1.18	0.46	0.66	0.15	0.07	0.37

To speed-up learning, each of the six two-class subproblems was further divided into many relatively smaller and more balanced two-class subproblems. Two different ways were used for ramdomly dividing the training data sets of larger two-class subproblems. The number of subsets for each of four classes (N_1, N_2, N_3, and N_4), the maximum number of training data for the two-class subproblems ("Max"), and the total number of the two-class subproblems ("Total") are summarized in Table IV.

C. Realization of Complete Learning

In this section, we demonstrate how to realize complete learning of complex EEG classification problems using the proposed method. We describe the learning process in detail using the EEG problem $\mathcal{A}_5$ shown in Table V, which has 300 features extracted from the duration of $0 \sim 0.9$ s. The experimental results in Table VI indicate that it is difficult to adequately learn this problem using single MLPs with a variable number of hidden units, initial weight values, and learning rates.

At the beginning of learning, 1189 three-layer perceptrons with 300 inputs, one hidden unit, and one output unit were selected as initial network modules to learn the corresponding 1189 two-class subproblems. When training these 1189 individual network modules up to 5000 epochs, the numbers of converged and unconverged network modules were 939 and 250, respectively. Following the minimization and maximization principles, all 1189 individually trained network modules were integrated into a M^3 network, as depicted in Fig. 2. Presenting all the 1491 training and 636 test data to the M^3 network, we obtained the performance of the M^3 network and finished the first round of learning. Here, the term "round of learning" refers to the whole process of training all the network modules for a given problem, integrating all the trained network modules into a M^3 network, and evaluating the learning and generalization accuracy of a M^3 network. The number of incorrect outputs produced by the M^3 network is shown in Table VII. From this table, one can see that there are 84 training data that were not learned successfully after the first round of learning and that there were 164 wrong outputs from the M^3 network with the test data.

TABLE VI
PERFORMANCE OF SINGLE MLPS WITH DIFFERENT NUMBERS OF HIDDEN UNITS ON EEG PROBLEM $\mathcal{A}_5$. VALUES ARE MEAN (TOP ROW) AND STANDARD DERIVATIONS (BOTTOM RAW) FOR THREE SIMULATIONS

No. hidden units	No. epochs	CPU time (sec)	Correct rate (%) Training	Test
40	**50000**	**50487**	99.20	64.57
	0	0	0.66	1.89
60	50000	69778	99.46	**66.88**
	0	0	0.39	1.53
80	50000	89669	99.22	64.78
	0	0	0.87	1.92
100	50000	108911	**99.84**	63.47
	0	0	0.11	2.01
120	44701	115299	98.84	62.32
	7493	19289	1.57	1.35
140	42362	**122983**	98.55	62.84
	12215	36290	1.30	2.00
160	36170	121886	99.82	64.15
	10112	34081	0.21	1.45

There are four different ways to solve the 250 unconverged subproblems remaining after the first round of learning: 1) increase the number of hidden units for each of the network modules; 2) change the parameters of the learning algorithm, such as the learning rate and initial weight values; 3) increase the number of epochs; and 4) further divide each of the unconverged two-class subproblems into several relatively smaller and simpler two-class subproblems. In the present study, we used the first method, increasing by two the number of hidden units for the 250 unconverged network modules. In the second round of training, all the 250 re-trained network modules and the converged 939 network modules obtained in the first round of learning were integrated into a new M^3 network, the topology of which was identical to the former M^3 network. The performance of the new M^3 network is shown in the second row of Table VII. These results indicate that both the learning and generalization accuracy of the new M^3 network were improved after the second round of learning.

After the second round of learning, unconverged network modules still remained. To implement complete learning, the number of hidden units was increased to six for the remaining 47 unconverged network modules. In the third round of training, all the 47 re-trained network modules, the 203 converged network modules obtained in the second round of learning, and the 939 converged network modules obtained in the first round of learning were integrated into a M^3 network. The performance of the newly integrated M^3 network is shown in the third row of Table VII. From this table, we see that all the 1491 training data were adequately learned by the 1189 network modules through three rounds of learning. We also see that better generalization accuracy (75.16%) was obtained.

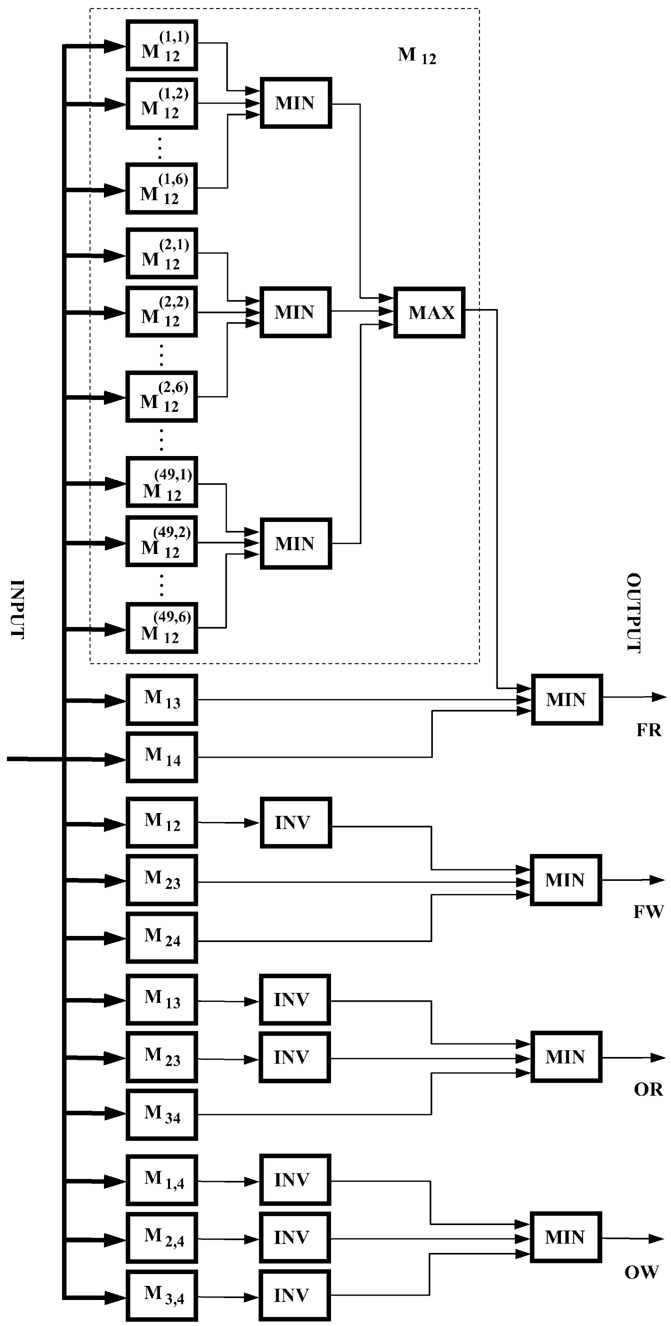

Fig. 2. Schematic representation of min-max modular network used for solving the four-class EEG classification problems; thin lines and arrows represent scalar inputs or outputs and thick lines and arrows represent vector inputs. Due to space requirements, note that only module $\mathbf{M}_{1,2}$ is plotted in detail, and the other modules are roughly illustrated.

To examine the effect of the initial network module size on learning speed and generalization performance, 1189 three-layer perceptrons with five hidden units were selected as initial network modules. When these network modules were trained up to 5000 epochs, all the network modules converged successfully.

From the above simulation results on the EEG problem $\mathcal{A}_5$, we observed that the proposed method offers several useful features. 1) Complete learning can be easily realized, provided that there are no contradictory training data. This useful feature has been successfully applied to error detection of an annotated corpus in natural language processing [23]. 2) Separation of the module training process and the module combination process results in a very flexible mechanism of rearranging the networks. 3) Designing and training networks can be performed *automatically*.

TABLE VII
THE PROCESS OF COMPLETELY TRAINING 1189 NETWORK MODULES ON EEG PROBLEM $\mathcal{A}_5$ AND PERFORMANCE OF THE CORRESPONDING M^3 NETWORKS

No. round	No. hidden units	No. tasks	No. network modules		No. wrong outputs	
			Converged	Unconverged	Training	Test
1	1	1189	939	250	84	164
2	2	250	203	47	23	160
3	6	47	47	0	0	**158**

TABLE VIII
PERFORMANCE OF THE M^3 NETWORKS ON EEG PROBLEM $\mathcal{A}_6$, IN WHICH THE NUMBER OF HIDDEN UNITS FOR EACH OF THE 1189 NETWORK MODULES WAS SELECTED AS 2, 4, 6, 8, AND 10, RESPECTIVELY. VALUES ARE MEAN (TOP ROW) AND STANDARD DERIVATIONS (BOTTOM ROW) FOR THREE SIMULATIONS

No. hidden units	No. epochs		CPU time (sec)		Correct rate (%)	
	Max	Total	Max	Total	Training	Test
2	5000	220925	41	3164	98.38	80.35
	0	9437	0	84	0.54	0.22
4	4342	56896	45	**2003**	99.40	80.77
	930	3625	9	29	0.47	0.53
6	581	**45272**	**10**	2007	**100**	80.45
	179	109	2	26	0	0.20
8	2767	49878	21	2134	100	**80.92**
	1930	1935	17	32	0	0.37
10	1166	50697	21	2253	100	80.77
	884	2260	14	25	0	0.41

D. Learning Results

The learning results of the six EEG problems defined in Table V are summarized in Tables V, VIII, and X. The performance of the M^3 networks trained on the EEG problem $\mathcal{A}_6$ is shown in Table VIII; "Max" refers to the maximum number of epochs or the maximum CPU time required for training any network module. From Table VIII, one can see that complete training was realized when the number of hidden units for each of the network modules was greater than four, and the best generalization performance (80.92%) was obtained when each of the network modules had eight hidden units. One can also see that only 10 s were required for completely learning the EEG problem in parallel. The detail classification results obtained by a M^3 neural network on EEG problem $\mathcal{A}_6$ are shown in Table IX, where each of the network modules in the M^3 neural network has 6 hidden units.

The generalization performance of the M^3 networks trained on all the six EEG problems is also shown in Table V; each of the six EEG problems was learned three times using network modules that had different numbers of hidden units. From these learning results, we conclude: 1) The EEG signals extracted from the period between $0 \sim 0.225$ s contribute little to discrimination because the generalization performance of the M^3 network trained on the EEG problem $\mathcal{A}_1$ is just 54.09%. 2) The EEG signals extracted from the period between $0.3 \sim 0.6$ s play an important role in discrimination. The reason is that the generalization performance of the M^3 network trained on the EEG problem $\mathcal{A}_2$ is better than that of the M^3 network trained on the EEG problem $\mathcal{A}_3$, even through the number of features in $\mathcal{A}_2$ is less than that in $\mathcal{A}_3$.

TABLE IX
THE DETAIL CLASSIFICATION RESULTS OBTAINED BY A M^3 NETWORK ON EEG PROBLEM $\mathcal{A}_6$

Class	No. test data	As "FR"	As "FW"	As "OR"	As "OW"	Unknown	No. incorrect outputs	Correct rate (%)
FR	430	386	28	15	0	1	44	89.77
FW	68	43	20	5	0	0	48	29.41
OR	128	18	0	108	0	2	20	84.38
OW	10	8	1	0	0	1	10	0.00

TABLE X
PERFORMANCE OF SINGLE MLPS WITH DIFFERENT NUMBERS OF HIDDEN UNITS ON EEG PROBLEM $\mathcal{A}_6$. VALUES ARE MEAN (TOP ROW) AND STANDARD DERIVATIONS (BOTTOM RAW) FOR THREE SIMULATIONS

No. hidden units	No. epochs	CPU time (sec)	Correct rate (%)	
			Training	Test
40	**1663**	14171	**100.0**	78.36
	164	1396	0.0	1.37
60	1399	14682	100.0	78.93
	136	1425	0.0	1.32
80	1156	**13902**	100.0	**80.71**
	173	2080	0.0	0.66
100	**1139**	15445	100.0	80.50
	47	636	0.0	0.68

TABLE XI
PERFORMANCE OF THE M^3 NETWORKS ON NINE DIFFERENT EEG PROBLEMS WITHIN THE SAME 3.03–17.11 HZ BANDWIDTH. VALUES ARE MEAN (TOP ROW) AND STANDARD DERIVATIONS (BOTTOM ROW) FOR THREE SIMULATIONS IN WHICH THE NUMBER OF HIDDEN UNITS FOR EACH OF THE 469 NETWORK MODULES WAS SELECTED AS 5, 7, AND 10, RESPECTIVELY.

Data set	Duration (sec)	No. features	No. epochs		CPU time (sec)		Correct rate (%)	
			Max	Total	Max	Total	Training	Test
$\mathcal{B}_1$	0~0.5	1100	5000	497924	**452**	**43954**	99.64	72.69
			0	31675	15	1721	0.08	0.20
$\mathcal{B}_2$	0~1.0	2200	5000	206791	1645	66434	99.69	75.58
			0	28242	24	7982	0.21	0.49
$\mathcal{B}_3$	0~1.5	3300	5000	110405	3466	75545	99.84	75.58
			0	7472	39	4176	0.11	0.41
$\mathcal{B}_4$	0~2.0	5500	3176	75775	3889	85664	99.96	78.35
			1716	789	2096	7775	0.06	0.07
$\mathcal{B}_5$	0~2.5	6600	3725	52559	6962	96966	**100.00**	81.81
			1803	6340	3352	11204	0	0.27
$\mathcal{B}_6$	0~3.0	7700	3516	48799	9470	129855	100.00	82.18
			2098	8508	5638	22097	0	0.53
$\mathcal{B}_7$	0~3.5	8800	1266	34524	4582	123018	100.00	**82.55**
			668	4178	2400	14208	0	0.44
$\mathcal{B}_8$	0~4.0	8800	**471**	27430	2250	127970	100.00	82.55
			126	2862	608	12750	0	0.71
$\mathcal{B}_9$	0~4.5	9900	518	**24508**	3104	143314	100.00	82.50
			239	2413	1439	13597	0	0.20

The nine EEG problems and the corresponding learning results are shown in Table XI. Each of the nine EEG problems was learned by three different M^3 networks, in which the number of hidden units for the network modules was set to 5, 7, and 10, respectively. Examining the learning results shown in Table XI, we can make the following observations on the relationship among generalization accuracy of trained classifiers, two different hippocampal EEG bandwidths, and different numbers of features extracted from various duration of EEG signals. 1) The greater the number of features, i.e., the longer the EEG signal duration, the fewer the number of average epochs required for training each of the network modules. In other words, EEG problems that have many features converge more quickly than those that have few. 2) The generalization performance of the M^3 networks improves as the duration of the EEG signals increases in the range between 0.5 s and 3.5 s. However, the generalization performance degrades when the duration is longer than 3.5 s. This trend indicates that there is an *optimal* duration for good generalization performance, and the longest duration examined is by no means the optimal one. 3) The generalization performance of the M^3 networks improves slightly as the hippocampal EEG frequency bandwidths are increased from 5–12 Hz to 3.03–17.11 Hz.

VI. COMPARISON STUDIES

In this section, we compare experimentally, through two EEG problems, the proposed method with the existing approach that is based on conventional MLPs.

A. Training Time

To compare the performance of our method with the existing approach, two EEG problems $\mathcal{A}_5$ and $\mathcal{A}_6$ were learned by single MLPs with various numbers of hidden units, initial weight values, and learning rates. The simulation results from these single MLPs are shown in Tables VI and X. From Table VI, one can see that a very long computing time was required for training each of the MLPs on EEG problem $\mathcal{A}_5$. For these MLPs that had 100 or fewer hidden units, no complete learning could be obtained, even if twelve episodes of 50 000 epochs each were performed. When the number of hidden units for single MLPs was increased to 120, some of these MLPs could realize complete learning. However, the CPU time required for training each of these MLPs was over 32 h and their generalization performance was worse than that of the smaller MLPs. The simulation results in Table X indicate that all of the MLPs could learn EEG problem $\mathcal{A}_6$ completely.

For the reader's convenience, the simulation results of two MLPs and two M^3 networks on EEG problems $\mathcal{A}_5$ and $\mathcal{A}_6$ are summarized in Table XII. Provided that learning with our method was performed in parallel, the results show that our new method is about 6344 and 662 times faster on learning EEG problems $\mathcal{A}_5$ and $\mathcal{A}_6$, respectively, than the existing approach based on conventional MLPs. The results in Table XII also indicate that, even though all the network modules were trained in serial, our method is still much faster than the existing approach.

B. Generalization

Generalization is one of the most important measures for judging the efficiency of neural network models. This property refers to the ability of a trained neural network to generate reasonable outputs for novel inputs that did not occur during learning. A popular method for examining generalization performance is to measure the performance of trained networks on test data that were not presented during training, i.e., the generalization performance of a trained network is judged by the correct recognition rates on test data.

To compare the generalization performance of our method with that of the existing approach, a number of experiments was performed

TABLE XII
PERFORMANCE COMPARISON ON EEG PROBLEMS $\mathcal{A}_5$ AND $\mathcal{A}_6$ OF THE M^3 NETWORKS WITH SINGLE MLPS.

Method	No. features	No. modules	No. hidden units	CPU time (s.)		Correct rate (%)		Speedup	
				Max	Total	Training	Test	Parallel	Serial
MLP	300	1	60	69778	69778	99.46	66.88	-	-
M^3	300	1189	5	**11**	**986**	**100.0**	**74.68**	**6344**	**71**
MLP	2000	1	80	13902	13902	100.0	80.71	-	-
M^3	2000	1189	8	**21**	**2134**	**100.0**	**80.92**	**662**	**7**

on EEG problems $\mathcal{A}_5$ and $\mathcal{A}_6$. The simulation results are shown in Tables VI, VIII, X, and XII. From Table XII, one can see that the generalization performance (74.68%) of our method is far superior to that (66.88%) of the existing approach on EEG problem $\mathcal{A}_5$. For EEG problem $\mathcal{A}_6$, the generalization performance (80.92%) of our method is slightly better than that (80.71%) of the existing approach.

Examining the correct recognition rates shown in Table VIII for the test data, one can see that our new method generates stable generalization performance, i.e., the generalization performance of the M^3 networks was almost invariant with the size of network modules in the M^3 networks. A similar phenomenon can also be observed from the simulation results on the nine EEG problems (Table XI). This property is very useful for realizing efficient learning because we can easily determine the topology of network modules according to the module expansion procedure [19]. Furthermore, it is not necessary to train several network modules of different sizes in order to achieve better generalization. In contrast, the simulation results in Tables VI and X show that the number of hidden units for single MLPs strongly affects the generalization performance. For example, the correct rate obtained by the MLP with 60 hidden units was 66.88%, while the correct rate achieved by the MLP with 120 hidden units was only 62.32%. To obtain better generalization performance, one needs to train several MLPs that have different numbers of hidden units. This results in a much longer training time compared to that for training a single MLP.

C. Scaling

In the application of neural network solutions to large-scale, real-world problems, one of the biggest difficulties encountered by most neural network models is the *scaling problem*. This problem addresses the issue of how well the training time and generalization performance of a network model behave as the learning task increases in size and complexity [24]. While most network models successfully operate in small and artificial domains, few of them show promise with large-scale, real-world problems.

It has been shown theoretically that learning in traditional MLPs is NP-complete [25]. That is, training MLPs becomes intractable as the problem size becomes larger. The simulation results mentioned above also indicate that training large-scale MLPs is computationally expensive, especially for complicated problems such as the EEG problem $\mathcal{A}_5$.

In contrast, the simulation results in Tables VIII, XII, and XI indicate that our method could scale-up to large-scale problems. Specifically, the computing time required for training the network modules in the M^3 networks was reasonable and did not grow exponentially with problem size.

VII. DISCUSSION

It is widely believed that the human brain's electrical activity reflects higher cognitive functions such as attention, arousal, and even consciousness [26]. For example, the P300 component of event-related potentials (mainly interpreted as a cognitive potential) has been observed in human subjects when they are required to respond to infrequent events, while ignoring frequent ones in psychophysical tasks [1]. The P300 is an endogenous component of the event-related potential (ERP) that appears as a positive deflection with a peak latency of approximately 300 – 600 ms for simple auditory stimuli [27]. This deflection is largest at centro-parietal electrodes.

In animals (including monkeys, cats and rats) trained to perform similar "oddball" types of tasks, P300-like potentials are remarkably similar to those in human subjects. Several investigators have found that the hippocampal theta rhythm is responsible for the genesis of the P300 response [26], [28]. Hippocampal theta rhythm is a local field potential also known as rhythmical slow-wave activity in the rat. This sinusoidal-like EEG signal occurs at frequencies within the bandwidth of 5 and 12 Hz. On the contrary, human theta rhythm occurs at frequencies between 5 Hz and 8 Hz in neocortex and limbic system. The long-standing focus on the theta rhythm in brain research stems from the hope that it may be linked to higher cognitive functions such as, "attention," "motivational state," or "learning" [29]. Theta rhythm can be recorded from the hippocampal formation of mammals during voluntary motor behaviors, such as walking, running, rearing, jumping, swimming, digging, manipulating objects with the forelimbs, and orienting head and body movements [30]. In recent experiments involving human subjects, the theta rhythm recorded from the cortical surface during virtual spatial navigation has been reported [31], and its possible connection to hippocampal theta rhythm has been discussed [32]. Given the close relationship between the P300 ERP and event-related changes in theta band rhythm, event-related theta rhythm, like P300 ERP responses [33], can be used for a BCI. Although the P300 ERP has been used as a BCI, to date, the event-related theta rhythm has not been implemented in a BCI system.

In this paper, we explored whether a min-max modular neural network can classify event-related changes in hippocampal theta rhythm recorded from rats performing an auditory oddball discrimination task. As expected, the min-max modular neural network successfully discriminated four behavioral classes from single trial, hippocampal EEG signals. More interestingly, the min-max modular neural network correctly identifies that the EEG signals extracted from the 0.3 – 0.6 s period of the task play an important role in discrimination (see Section V). The period from 0.3 to 0.6 s is identical to the P300 ERP latency. Our results, thus, demonstrate that event-related theta rhythm can be used for implementing an efficient BCI system.

We also investigated an optimal duration that can result in better discrimination performance. Our results suggest that there is an optimal duration that supports good generalization performance. However, the longest duration we investigated is by no means the optimal (see Section V). Finally, we confirmed that generalization performance of the min-max modular neural network improves as the bandwidth increases for the selected raw hippocampal EEG data used for classification (see

Section V). These results can guide the design of a practical BCI system for human subjects. Previously, we used the min-max modular neural network for testing hypotheses related to hippocampal theta rhythm and found that this network can be used for objective data interpretation. This can, at least partly, complement human subjective data interpretation [1]. Taken together, the min-max modular neural network can be applied to diverse bio-informatics and neuro-informatics research, such as complex and large biological data mining, objective data interpretation, and efficient BCI system implementation.

ACKNOWLEDGMENT

The authors would like to thank A. Talnov for source EEG data, and K. Shionoya, and J. Brankack for support and help. They also would like to thank Prof. A. F. Infantosi and the anonymous reviewers for their helpful comments and suggestions.

REFERENCES

[1] J. Shin, B. L. Lu, A. Talnov, G. Matsumoto, and J. Brankack, "Reading auditory discrimination behavior of freely moving rats from hippocampal EEG," *Neurocomputing*, vol. 38–40, pp. 1557–1566, 2001.

[2] A. A. petrosian, D. V. Prokhorov, W. Lajara-Nanson, and R. B. Schiffer, "Recurrent neural network-based approach for early recognition of Alzheimer's disease in EEG," *Clin. Neurophysiol.*, vol. 112, pp. 1378–1387, 2001.

[3] C. W. Anderson, "Effects of variations in neural network topology and output averaging on the discrimination of mental tasks from spontaneous electroencephalogram," *J. Intell. Syst.*, vol. 7, no. 1–2, pp. 165–190, 1997.

[4] B. O. Peters, G. Pfurtscheller, and H. Flyvbjerg, "Automatic differentiation of multichannel EEG signals," *IEEE Trans. Biomed. Eng.*, vol. 48, pp. 111–116, Jan. 2001.

[5] M. A. L. Nicolelis, "Actions from thoughts," *Nature*, vol. 409, pp. 403–407, 2001.

[6] J. del R. Millan, J. Mourino, M. franze, F. Cincotti, M. Varsta, J. Heikkonen, and F. Babiloni, "A local neural classifier for the recognition of EEG patterns associated to mental tasks," *IEEE Trans. Neural Networks*, vol. 13, pp. 678–686, May 2002.

[7] A. C. Tsoi, D. So, and A. A. Sergejew, "Classification of electroencephalogram using artificial neural networks," in *Advances in Neural Information Processing Systems*. San Mateo, CA: Morgan Kaufmann, 1994, vol. 6, pp. 1151–1158.

[8] N. Masic and G. Pfurtscheller, "Neural network based classification of single-trial EEG data," *Artif. Intell. Med.*, vol. 5, pp. 503–513, 1993.

[9] B. H. Jansen and P. R. Desai, "K-complex detection using multi-layer perceptrons and recurrent networks," *Int. J. Biomed. Computing*, vol. 37, pp. 249–257, 1994.

[10] C. Robert, J. Gaudy, and A. Limoge, "Electroencephalogram processing using neural networks," *Clin. Neurophysiol.*, vol. 113, pp. 694–701, 2002.

[11] N. Hazarika, J. Z. Chen, A. C. Tsoi, and A. Sergejew, "Classification of EEG signals using the wavelet transform," *Signal Processing*, vol. 59, pp. 61–72, 1997.

[12] B. L. Lu and M. Ito, "Task decomposition based on class relations: A modular neural network architecture for pattern classification," in *Biological and Artificial Computation: From Neuroscience to Technology*, J. Mira, R. Moreno-Diaz, and J. Cabestany, Eds. Berlin, Germany: Springer-Verlag, 1997, vol. 1240, Lecture Notes in Computer Science, pp. 330–339.

[13] ——, "Task decomposition and module combination based on class relations: A modular neural network for pattern classification," *IEEE Trans. Neural Networks*, vol. 10, pp. 1244–1256, Sept. 1999.

[14] J. Brankack, T. Seidenbecher, and H. Muller-Gartner, "Task-relevant late positive component in rats: Is it related to hippocampal theta rhythm ?," *Hippocampus*, vol. 6, pp. 475–482, 1996.

[15] S. Mallat, *A Wavelet Tour of Signal Processing*, 2nd ed. New York: Academic, 2001.

[16] B. L. Lu, J. Shin, and M. Ichikawa, "Massively parallel classification of EEG signals using min-max modular neural networks," in *Artificial Neural Networks-ICANN 2001*, G. Dorffiner, H. Bischof, and K. Hornik, Eds. Berlin, Germany: Springer-Verlag, 2001, vol. 2130, Lecture Notes in Computer Science, pp. 601–608.

[17] R. Anand, K. G. Mehrotra, C. K. Mohan, and S. Ranka, "An improved algorithm for neural network classification of imbalanced training sets," *IEEE Trans. Neural Networks*, pp. 962–973, Nov. 1993.

[18] G. S. Almasi and A. Gottlieb, *Highly Parallel Computing*, 2nd ed. San Francisco, CA: Benjamin/Cummings, 1994.

[19] B. L. Lu, Q. Ma, M. Ichikawa, and H. Isahara, "Efficient part-of-speech tagging with a min-max modular neural-network model," *Appl. Intell.*, vol. 19, pp. 65–81, 2003.

[20] N. Sundararajan and P. Saratchandran, *Parallel Architectures for Artificial Neural Networks*. Los Alamitos, CA: IEEE Comput. Soc. Press, 1998.

[21] B. L. Lu and M. Ichikawa, "Emergence of learning: An approach to coping with NP-complete problems in learning," in *Proc. IJCNN'2000*, vol. IV, Como, Italy, July 24–27, 2000, pp. 159–164.

[22] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing: Exploration in the Microstructure of Cognition*. Cambridge, MA: The MIT Press, 1986, vol. 1.

[23] Q. Ma, B. L. Lu, M. Murata, M. Ichikawa, and H. Isahara, "On-line error detection of annotated corpus using modular neural networks," in *Artificial Neural Networks-ICANN 2001*, G. Dorffiner, H. Bischof, and K. Hornik, Eds. Berlin, Germany: Springer-Verlag, 2001, vol. 2130, Lecture Notes in Computer Science, pp. 1185–1192.

[24] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1999.

[25] J. S. Judd, *Neural Network Design and the Complexity of Learning*. Cambridge, MA: MIT Press, 1990.

[26] E. Basar, *Brain Function and Oscillations, II. Integrative Brain Function Neurophysiology and Cognitive Processes*. Berlin, Germany: Springer-Verlag, 1998.

[27] T. W. Picton, "The P300 wave of the human event-related potential," *J. Clin. Neurophysiol.*, vol. 9, pp. 456–479, 1992.

[28] C. Basar-Eroglu, E. Basar, and F. Schmielau, "P300 in freely moving cats with intracranial electrodes," *Int. J. Neurosci.*, vol. 60, pp. 215–226, 1991.

[29] W. Klimesch, "EEG alpha and theta oscillations reflect cognitive and memory performance: A review and analysis," *Brain Res. Rev.*, vol. 29, pp. 169–195, 1999.

[30] C. H. Vanderwolf, "Hippocampal electrical activity and voluntary movement in the rat," *Electroenceph. Clin. Neurophysiol.*, vol. 26, pp. 407–418, 1969.

[31] M. J. Kahana, R. Sekuler, J. B. Caplan, M. Kirschen, and J. R. Madsen, "Human theta oscillations exhibit task dependence during virtual maze navigation," *Nature*, vol. 399, pp. 781–784, 1999.

[32] J. O'Keefe and N. Burgess, "Theta activity, virtual navigation and the human hippocampus," *Trends Cogn. Sci.*, vol. 3, pp. 403–407, 1999.

[33] E. Donchin, K. M. Spencer, and R. Wijesinghe, "The mental prosthesis: Assessing the speed of a P300-based brain-computer interface," *IEEE Trans Rehab. Eng.*, vol. 8, pp. 174–179, June 2000.

Erratum to "Quantifying Ventricular Fibrillation: *In Silico* Research and Clinical Implications"

In [1], Dr. Alexander V. Panfilov, was misidentified as Alberto V. Pavilov. We apologize for any inconvennience this error may have caused.

REFERENCES

[1] A. V. Panfilov and P. L. M. Kerkhof, "Quantifying Ventricular Fibrillation: *In Silico* Research and Clinical Implications," *IEEE Trans. Biomed. Eng.*, pp. 195–196, Jan. 2004.

Manuscript received January 20, 2004.
Digital Object Identifier 10.1109/TBME.2004.824567