Conversion of EEG activity into cursor movement by a brain-computer interface (BCI).

Conversion of EEG Activity Into Cursor Movement by a Brain–Computer Interface (BCI)

Georg E. Fabiani, Dennis J. McFarland, Jonathan R. Wolpaw, and Gert Pfurtscheller, *Member, IEEE*

***Abstract*—The Wadsworth electroencephalogram (EEG)-based brain-computer interface (BCI) uses amplitude in mu or beta frequency bands over sensorimotor cortex to control cursor movement. Trained users can move the cursor in one or two dimensions. The primary goal of this research is to provide a new communication and control option for people with severe motor disabilities. Currently, cursor movements in each dimension are determined 10 times/s by an empirically derived linear function of one or two EEG features (i.e., spectral bands from different electrode locations).**

This study used offline analysis of data collected during system operation to explore methods for improving the accuracy of cursor movement. The data were gathered while users selected among three possible targets by controlling vertical [i.e., one-dimensional (1-D)] cursor movement. The three methods analyzed differ in the dimensionality of the cursor movement [1-D versus two-dimensional (2-D)] and in the type of the underlying function (linear versus nonlinear).

We addressed two questions: Which method is best for classification (i.e., to determine from the EEG which target the user wants to hit)? How does the number of EEG features affect the performance of each method? All methods reached their optimal performance with 10–20 features. In offline simulation, the 2-D linear method and the 1-D nonlinear method improved performance significantly over the 1-D linear method. The 1-D linear method did not do so. These offline results suggest that the 1-D nonlinear or the 2-D linear cursor function will improve online operation of the BCI system.

***Index Terms*—Augmentative communication, brain–computer interface (BCI), electroencephalography, feedback.**

I. Introduction

RECENT studies show that people can communicate by controlling certain components of their electroencephalographam (EEG). A system that makes this possible is called a brain–computer interface (BCI). BCI communication could substantially improve quality of life for people with no or very little voluntary muscle control [9], [22]. Current approaches to EEG-based communication [21] can be divided into two groups, those that use time-domain EEG components (e.g., [2] and [5]) and those that use frequency-domain components (e.g., [10] and [14]–[16]). Frequency-domain methods use spectral analysis and focus on specific frequencies at specific scalp locations. The BCI developed at the Wadsworth Center uses mu (8–12 Hz) and/or beta (18–25 Hz) rhythms recorded over sensorimotor cortex to control a cursor on a video screen [10]. In the simplest case, the amplitude of a single spectral band at a single location on the scalp is used to determine one-dimensional (1-D) cursor movements. The user learns to control this amplitude. Cursor movements are calculated 10 times/s by an empirically derived linear equation. In offline analysis, data from the most recent sessions are used to determine the best location and frequency band for cursor control for the next sessions.

In the application discussed in this study, the user's task is to select among three boxes arranged vertically on the screen by controlling vertical cursor movement [23]. The outcome of a trial is either a hit or a miss, depending on whether the user selects the correct box (i.e., the target) or selects a different box. Performance is measured by percent correct, i.e., percent of the trials that are hits.

At present, one or two EEG features and the parameters of the linear equation that translates them into cursor movement are chosen by the operator on the basis of the user's past performance. Previous research showed that additional locations can improve performance [7]. This study uses up to 30 EEG features, selected by sequential forward selection [3] in offline analysis of the data from five trained users. It evaluates with offline analysis three different methods for classifying each trial as one of the three possible selections. For each method, a cursor function for possible online application is discussed. The methods differ in the dimensionality of the cursor movements and whether the cursor function is linear or nonlinear. The first method is 1-D linear, the second method is two-dimensional (2-D) linear and the third method is 1-D nonlinear. The goal was to determine the best method and the optimal number of features. A preliminary version of this work has been presented in abstract form [4].

II. Current BCI System

EEG is electrical activity produced by the brain and recorded from the scalp or from the cortical surface [13]. It is produced by the neurons and synapses of the central nervous system (CNS) in the course of their operations. The Wadsworth BCI measures particular features of EEG activity (i.e., mu and/or beta rhythms over sensorimotor cortex) and converts them into 1- or 2-D cursor movements. These cursor movements are used to provide feedback to the user. This feedback helps the user to learn to control his or her EEG activity and thereby control cursor movement.

Manuscript received July 18, 2001; revised February 25, 2004. This work was supported in part by the National Center for Medical Rehabilitation Research, National Institute of Child Health & Human Development, National Insitutes of Health under Grant HD30146; in part by the National Institutes of Health under Grant HD30146; in part by the National Institute for Biomedical Imaging and Bioengineering, National Institute of Neurological Disorders and Stroke, National Institutes of Health under Grant EB00856; in part by the Lorenz-Böhler Foundation under Grant 2/03; and in part by the "Land Steiermark."

G. E. Fabiani is with binuscan, MC-98013 Monaco, Principality of Monaco (e-mail: georg_fabiani@binuscan.com).

D. J. McFarland and J. R. Wolpaw are with the Laboratory of Nervous Systems Disorders, Wadsworth Center, New York State Department of Health and State University of New York, Albany, NY 12201 USA.

G. Pfurtscheller is with the Institute of Human-Computer Interfaces, Ludwig Boltzmann Institute for Medical Informatics and Neuroinformatics, University of Technology Graz, Graz 8010, Austria.

Digital Object Identifier 10.1109/TNSRE.2004.834627

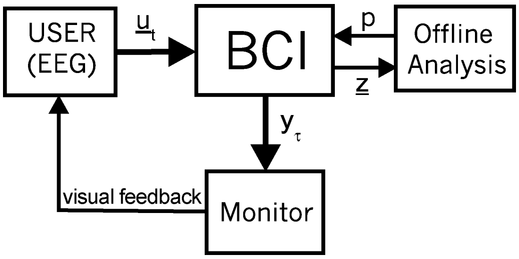

Fig. 1. EEG of the user $\underline{u}_t$ is translated into cursor movement by the BCI and displayed back to the user on a video monitor y_τ. Adjustments p to the BCI are made as a result of the offline analysis of stored data z.

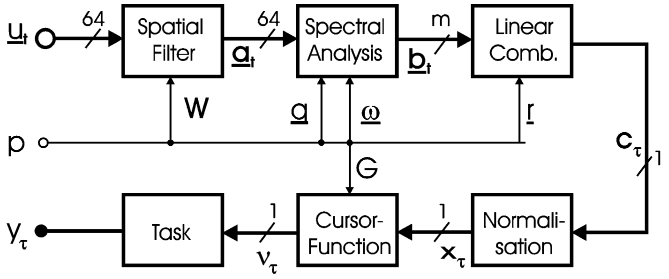

Fig. 2. Spatial filter improves signal-to noise ratio of the signal $\underline{u}_t$ of the 64 electrodes using a weight matrix W as parameter. Spectral analysis is performed on a selection $\underline{q}$ of the spatially filtered signals $\underline{a}_t$. Thereby, $\underline{w}$ selects the frequency bands. With the weights $\underline{r}$ these m spectrally analyzed signals $\underline{b}_\tau$ are linearly combined. The resulting scalar c_τ is normalized (x_τ) and then mapped to a cursor position ν_τ using the scalar G as an input. Finally, this cursor position is represented in the context of a task (e.g., three boxes, highlighted target) on the screen (y_τ).

A. BCI and Its Environment

Fig. 1 summarizes the current BCI operation. The BCI has two inputs and two outputs.

1) *Input $\underline{u}_t$*:

This is a 64-dimensional time discrete signal $\underline{u}_t = \{u_{1,t}, u_{2,t}, \ldots, u_{64,t}\}^T$. Each scalar $u_{i,t}$ $i \in \{1, 2, \ldots, 64\}$ is the EEG gathered from a different location on the scalp. The EEG is sampled at 128 Hz. Therefore, each sample corresponds to a time period of 1/128 s. The subscript t indexes these samples. Therefore, $\underline{u}_1$ is the sample of the first time period and $\underline{u}_2$ is the sample of the second time period.

2) *Input p*:

This is a user profile, containing parameters that adjust the BCI to the user.

3) *Output y_τ*:

This is the information displayed on a screen to provide feedback to the user. For this study, the screen showed a cursor and three boxes (i.e., the possible selections) arranged vertically, and one of the boxes was highlighted to indicate that it was the target (i.e., the correct selection). The index τ indices the output at a certain time. The time between two outputs is 100 ms.

4) *Output $\underline{z}$*:

Contains data of the current session for later offline analysis.

B. BCI Components

The Box "BCI" in Fig. 1 has six components. These are shown in Fig. 2.

1) Spatial Filter: A spatial filter improves the signal-to-noise ratio by subtracting a weighted subset of the 63 remaining electrodes from the electrode of interest. The output $\underline{a}_t$ is a vector containing 64 different linear combinations of the input $\underline{u}_t = \{u_{1,t}, \ldots, u_{64,t}\}^T$. Each scalar $a_{i,t} = \sum_{j=1}^{64} \{w_{i,j} * u_{j,t}\}$ $i \in \{1, \ldots, 64\}$ is the spatially filtered signal from the electrode i at the time period t. The type of the spatial filter defines the weight matrix $w_{64\times 64}$. In practice, a Laplacian Reference filter or a common average reference (CAR) filter [11] is used.

2) Spectral Analysis: A 200-ms time window from the spatially filtered signals is transformed from the time to the frequency domain. The time window $TW(i, t, k) = \{a_{i,t-k+1}, a_{i,t-k+2}, \ldots, a_{i,t}\}$ comprises the last k signals at time index t of the ith component of a. The parameter k specifies the size of the time window and depends on the sampling frequency. The system uses the Burg algorithm for estimating autoregressive (AR) coefficients of order 10. Any spectral band can then be computed from these AR coefficients [6]. Let $F(TW(i, t, k), \omega)$ be the spectrum of a time window TW. The output $\underline{b}_\tau$ is a vector, containing a selection of spectral bands from the time window

$$\underline{b}_\tau = \{\mathcal{F}(TW(q_1, t, k), \omega_1), \ldots, \mathcal{F}(TW(q_m, t, k), \omega_m)\}^T. \tag{1}$$

The vector $\underline{q} = \{q_1, \ldots, q_m\}^T$ selects the scalars of $\underline{a}$ (i.e., electrodes) to be analyzed and $\omega = \{\omega_1, \ldots, \omega_m\}$ denotes the selected spectral bands. The index τ indices the time for $\underline{b}$. For continuous feedback, the time interval between $\underline{b}_\tau$ and $\underline{b}_{\tau+1}$ is 100 ms so that the 200-ms time windows are overlapping.

3) Linear Combination: The features of the vector $\underline{b}_\tau = \{b_{1,\tau}, \ldots, b_{m,\tau}\}$ (i.e., the amplitudes of different frequency bands at different locations) are linearly combined. The coefficients of this linear combination are $\{\alpha_1, \ldots, \alpha_m\}^T$. The output is defined by

$$c_\tau = \sum_{i=1}^{m} \alpha_i b_{i,\tau}. \tag{2}$$

Currently, the EEG components and the coefficients for the linear combination are determined by offline inspection of the data.

4) Normalization: The EEG is a nonstationary signal. The EEG-amplitudes display considerable apparently spontaneous variation, both across trials and across sessions. A moving average filter is used to adjust for trial-to-trial variations not predicted by target. The data from recent trials are used to predict the mean and standard deviation of the next trial [19]. The data of the next trial is then normalized by these estimates according to

$$x_\tau = \frac{c_\tau - \mu}{\sigma}. \tag{3}$$

It is assumed that this normalized signal is stationary.

5) Cursor Function: A linear function [10] translates EEG x_τ into cursor movement $\Delta\nu(x_\tau)$

$$\Delta\nu(x_\tau) = G * x_\tau \tag{4}$$

where G is a scaling factor. The cursor position at any time is defined by the sum of all movements performed in the trial

$$\nu(X) = \sum_{i=1}^{\tau} \Delta\nu(x_\tau) \tag{5}$$

where X is defined by $X = \{x_1, \ldots, x_\tau\}$.

6) Task: The data analyzed in this paper were gathered while users performed a 1-D, three-choice task [23], referred to as the three-box task. In this task, the user is presented with three boxes placed vertically on the screen: a top, a middle, and a bottom box. At the beginning of the task, the cursor appears in the center of the middle box and one box is highlighted to indicate that it is the target. One second later, the cursor begins to move controlled by the users EEG and continues to move until it selects one of the boxes. The selection is made either by remaining in a box for a specified dwell time (e.g., 2 s) or, for the top and bottom box, by reaching the top or bottom of the screen. If the box selected is the target, the trial is a hit; if the box selected is not the target, the trial is a miss.

Participants were five adults: four without disabilities (one woman and three men, ages 33–58) and a 24-year-old man with cerebral palsy who is confined to a wheel-chair and communicates with a touch-talker. All gave informed consent for the study, which had been reviewed and approved by the New York State Department of Health Institutional Review Board. After an initial evaluation defined the frequencies and scalp locations of each subject's spontaneous mu and beta rhythm activity, he or she learned EEG-based cursor control with a two target task in ten 30-min sessions (2–3/week) and then participated for 10–69 additional sessions devoted to a variety of studies. Over the course of each subject's participation, offline data evaluations led to adjustments in the electrode locations, frequency bands, and spatial filter used by the online algorithm that controlled cursor movement.

III. Methods

In offline analysis, we explored three different methods for classifying each trial, that is, deciding from the user's EEG which box he or she was trying to select (i.e., which box was the target). For each method, we determined the accuracy of classification and the dependence of this accuracy on the number of EEG features used. We used two linear methods, referred to as K1 and K2, and one nonlinear method referred to as K3. To evaluate the overall quality of each method, we used the data gathered from five users performing the three-box task (i.e., Section II-B6). The data of each user comprised 15 daily sessions. Each session included an average of 200 trials, each with an average trial duration of 2.2 s. We used ten subsets of this data (i.e., training sets) for training (i.e., to determine the parameters of each method) and ten subsets (i.e., test sets) for testing (i.e., to evaluate performance of each method). The ith training set contained sessions $\{i, i+1, \ldots, i+4\}$ and the ith test set contained the following session $(i+5)$. We used the first training set to initially select the best 30 EEG features (for each classifier separately) by sequential forward selection [3]. Then, we trained each classifier on the first training set and tested it on the first test set. We performed this procedure of training and testing ten times (i.e., for $\{1^{\text{st}}, \ldots, 10^{\text{th}}\}$ training and test set) using the same, initially selected, EEG features. The aim of this procedure was to simulate the online task as closely as possible.

A. Log-Transformation

As described in Section II-B, each scalar $b_{i,\tau}$ with $i \in \{1, \ldots, m\}$ (i.e., each feature) is the amplitude of a single spectral band at one scalp location. The logarithm of this signal is roughly normally distributed. For this reason, we took the logarithm of each component of the vector $\underline{b}_\tau$. Then, we normalized each scalar $\log(b_{i,\tau})$ to zero mean and standard deviation one (3). We defined the vector $\underline{x}_\tau = \{x_{1,\tau}, \ldots, x_{m,\tau}\}^T$, which contains the values $b_{i,\tau}$ after taking the logarithm and the normalization.

B. Trial Means

The data used offline have different trial lengths caused by the termination criterion of the three-box task (Section II-B6). This trial-length should not influence the classifiers because the trial length was determined by the algorithm used online. For this reason, we used trial means for training and testing. We defined $\underline{z}(X)$ as the trial mean

$$\underline{z}(X) = \frac{1}{\tau} \sum_{i=1}^{\tau} (\underline{x}_i). \tag{6}$$

C. Classification (i.e., Determination of Which Box Is the Target)

The discriminant function $y_1(\underline{z})$ is a linear mapping of the trial mean $\underline{z}(X)$ to one dimension

$$y_1(\underline{z}) = \underline{w}^T * \underline{z} + w0. \tag{7}$$

The parameters $\underline{w}$ and $w0$ were obtained by the pseudoinverse method [1], [8] which minimizes the sum squared error (SSE) between the predicted values (i.e., y_1) and the target values (i.e., 1/0/-1 for top/middle/bottom target). The classification of a trial was then determined by whether the discriminant function y_1 was higher than the threshold $\text{th}_{\text{Top/Middle}}$, lower than the threshold $\text{th}_{\text{Middle/Bottom}}$ or between these two thresholds. The thresholds were determined so that trials with top, middle, and bottom targets had the same probability of correct classification. The classifier K1 was defined as

$$K1(\underline{z}) = \begin{cases} \text{Top}, & \text{if } y_1(\underline{z}) > \text{th}_{\text{Top/Middle}} \\ \text{Bottom}, & \text{if } y_1(\underline{z}) < \text{th}_{\text{Middle/Bottom}} \\ \text{Middle}, & \text{else} \end{cases}. \tag{8}$$

The classifiers K2 and K3 used two linearly extracted features g1(z) and g2(z) of the feature space z(X). The extracted feature space g(z) was determined by two linear discriminant functions YTop/Middle(z) and YMiddle/Bottom(z). The weights of these discriminant functions were determined by performing linear discriminant analysis (LDA) [1] between a pair of classes (i.e., Top/Middle, Middle/Bottom) of the training data. Thus, g(z) was defined as

$$\underline{g}(\underline{z}) = \left\{ \begin{array}{c} Y_{\text{Top/Middle}}(\underline{z}) \\ Y_{\text{Middle/Bottom}}(\underline{z}) \end{array} \right\}. \tag{9}$$

The discriminant functions $y_{\text{Top/Middle}}(\underline{g})$, $y_{\text{Top/Bottom}}(\underline{g})$, and $y_{\text{Middle/Bottom}}(\underline{g})$ discriminated between each pair of classes in the previously defined 2-D extracted feature space

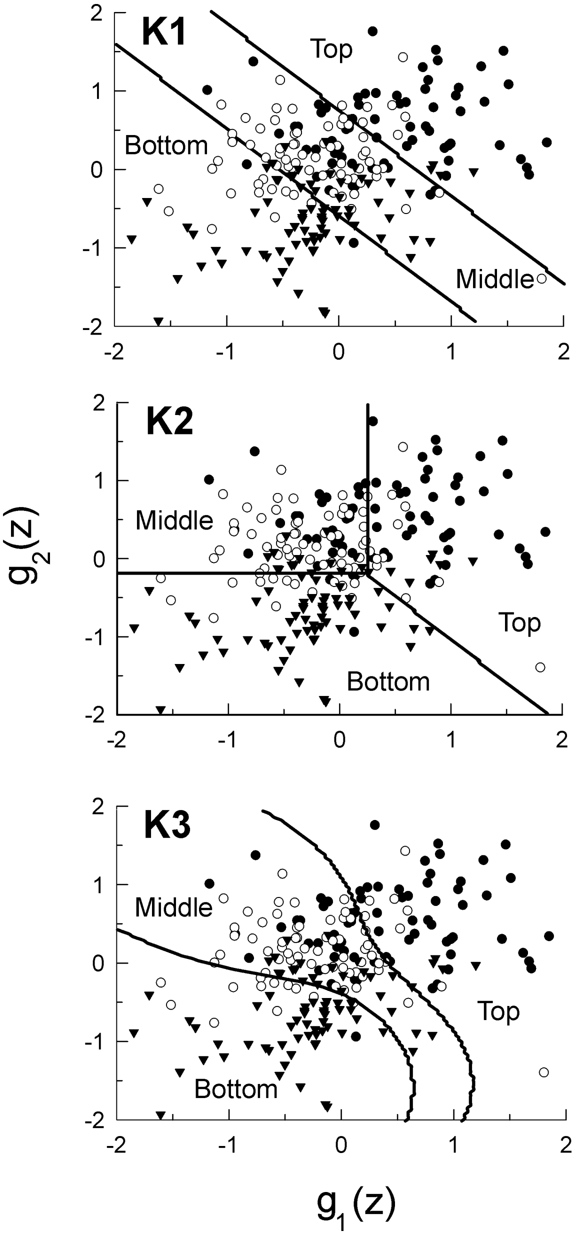

Fig. 3. Each symbol is a single trial. The target is the top box for full circles, the middle box for open circles, and bottom box for full triangles. For each graph (i.e., K1, K2, or K3), the three areas defined by the lines show the results for the classification of one session by the classifier K1, K2, or K3.

(g). The classifier K2 used these functions to divide the 2-D extracted feature space $\underline{g}$ into three areas, each associated with one target (Fig. 3 shows an example)

$$K2(\underline{g}) = \begin{cases} \text{Top}, & \text{if } y_{\text{Top/Middle}}(\underline{g}) > 0 \\ & \wedge y_{\text{Top/Bottom}}(\underline{g}) > 0 \\ \text{Bottom}, & \text{if } y_{\text{Top/Bottom}}(\underline{g}) < 0 \\ & \wedge y_{\text{Middle/Bottom}}(\underline{g}) < 0 \\ \text{Middle}, & \text{else.} \end{cases} \tag{10}$$

It is also possible to use these three discriminant functions for K2 in the feature space $\underline{z}$ instead of $\underline{g}$. In this case, the classification results would be similar, but 2-D feedback would not be possible (Section III-E).

The classifier K3 used a statistical method on the 2-D extracted feature space $\underline{g}$. We assumed multivariate normal distribution for the three classes. Using Bayes theorem [1] and equal prior probabilities, we calculated posterior probabilities P for all three classes. Thus, the nonlinear discriminant function $y_3(\underline{g})$ minimized the SSE between the prediction (i.e., y_3) and the target values of the training data. It was defined by

$$y_3(\underline{g}) = P(\text{Top}|\underline{g}) - P(\text{Bottom}|\underline{g}). \tag{11}$$

As for K1, the thresholds were determined so that trials with each of the three targets had the same probability of correct classification. The classifier K3 was defined as

$$K3(\underline{g}) = \begin{cases} \text{Top}, & \text{if } y_3(\underline{g}) > \text{th}_{\text{Top/Middle}} \\ \text{Bottom}, & \text{if } y_3(\underline{g}) < \text{th}_{\text{Middle/Bottom}} \\ \text{Middle}, & \text{else.} \end{cases} \tag{12}$$

D. Feature Selection

All the methods previously described can process multivariate input. For this study, 15 relevant scalp locations over sensorimotor cortex were evaluated (i.e., FC_5, FC_3, FC_4, FC_6, C_3, C_1, C_z, C_2, C_4, C_6, CP_3, CP_1, CP_z, CP_2, and CP_4 [20]). Each of these 15 locations was spatially filtered with a large laplacian filter. Then seven different 3-Hz wide frequency bands, all in the range 8–27 Hz, were analyzed for each of these spatially filtered locations. Thus, the feature space consisted of 105 features. Many features contained minimal or redundant information and were, therefore, not useful.

In this study, sequential forward selection [3] was used to find for each method (i.e., K1, K2, and K3) separately the most important 30 features ordered by their importance. Each feature was found by minimizing the SSE between the prediction (i.e., y_1, $\underline{y}_2$, or y_3) and the target values of the first training set (i.e., sessions $\{1, \ldots, 5\}$).

The sequential forward selection algorithm began by finding the best single feature as the first feature in the subset. Next, all remaining features were evaluated (i.e., SSE was calculated) in combination with the subset. The best of these features was then added to the subset. This was continued until the subset contained 30 features.

E. Feedback

For each classifier, a cursor position (i.e., ν_1, ν_2, ν_3) was obtained by multiplying the time index within a trial (i.e., τ) with the discriminant function (i.e., y_1 for K1, y_3 for K3) or the extracted features (i.e., $\underline{g}$ for K2). The resulting cursor position was 1-D for K1 and K3 and 2-D for K2.

The cursor movements of ν_1 were derived by a linear function (i.e., $\Delta\nu_1(x_i) = \underline{w}^T * \underline{x}_i + w0$) and the cursor position was the sum of these movements

$$\nu_1(X) = \tau * y_1(\underline{z}(X)) = \sum_{i=1}^{\tau} \Delta\nu_1(x_i). \tag{13}$$

For K2, the cursor position was obtained by $\nu_2(X) = \tau * \underline{g}(\underline{z}(X))$. Each dimension of the cursor movements $\Delta\nu_2$ was, therefore, expressed by a linear function similar to $\Delta\nu_1$. The 2-D cursor position ν_2 was then defined by the sum of these movements.

The cursor position of the classifier K3 was obtained by multiplying the time index by the discriminant function [i.e., $\nu_3(X) = \tau * y_3(\underline{z}(X))$].

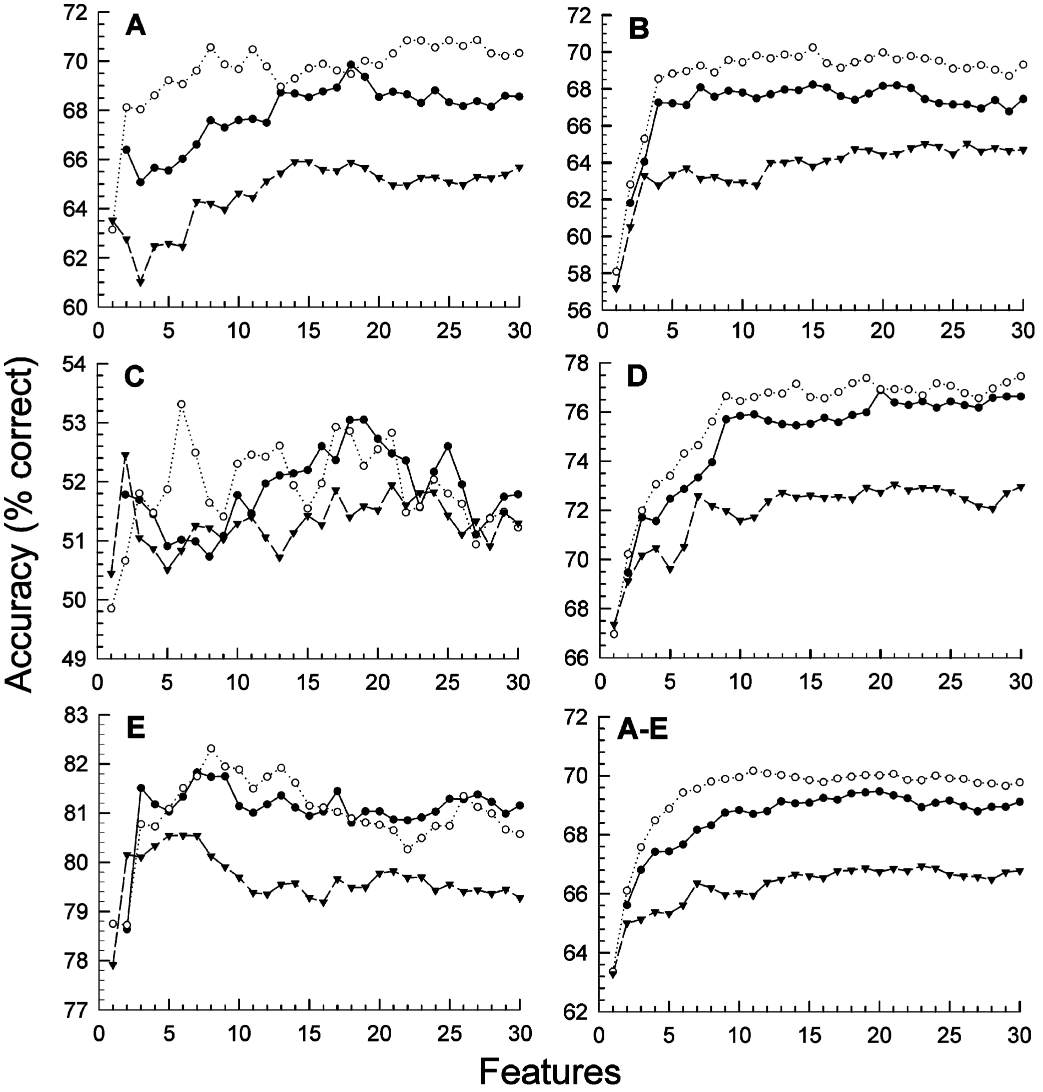

Fig. 4. Discriminant function y_3 as a function of the extracted features $g_1(\underline{z})$ and $g_2(\underline{z})$.

IV. Results

We assessed the performance of the classification methods K1, K2, and K3, described in the previous section. For each method, the percentage of trials correctly classified was determined as a function of the number of features used.

Fig. 4 summarizes the results. The results show that more features can substantially improve performance for all methods. The improvement was strongest for K2 and K3. For more than three features, the performance of K2 (or K3) was significantly ($p < 0.01$, paired t-test) higher than the performance of K1. Performance increased with features up to a point. With too many features, the performance deteriorated again. For K1, the performance deteriorated from a maximum of 66.8% with 23 features to 65.4% with 80. For K2, it deteriorated from a maximum of 70.2% with 11 features to 68.0% with 80. For K3, it deteriorated from a maximum of 69.5% with 20 features to 67.7% with 80. These results suggest that the classifiers give their maximum performance with 10–20 features (see Fig. 4).

The actual online performance averaged over subjects A–E was 69.3%. Data used for online operation was sampled with a different A/D converter so that offline analysis has been done with data that were not identical. To eliminate this influence in the results, we performed an offline simulation of the linear cursor function that was used online. In offline analysis, accuracy averaged over subjects A–E was 65.6%. Using more than three features, the performance of the classifier K2 (or K3) was significantly ($p < 0.01$, paired t-test) higher than the performance of the simulation. In contrast, there was no significant ($p < 0.01$, paired t-test) difference between the performance of K1 and the performance of the simulation.

We tested the 1-D cursor functions ν_1 and ν_3 (i.e., linear and nonlinear). For all cursor movements of all subjects, we analyzed r^2 [12] which reflects the user's EEG-control. This was 0.130 for $\Delta\nu_1$ and 0.133 for $\Delta\nu_3$. Fig. 6 shows a histogram of these cursor movements. The percentage of the correct movements toward the bottom target was 71.8% for $\Delta\nu_1$ and 74.3% for $\Delta\nu_3$.

V. Discussion

This study focused on the classification of trials from a three-box task. We addressed two questions: What methods are best for classification? How does the number of features affect the performance of each method?

We evaluated three different methods: the classifiers K1, K2, and K3. Each classifier is associated with a cursor function: 1-D linear for K1, 2-D linear for K2, and 1-D nonlinear for K3.

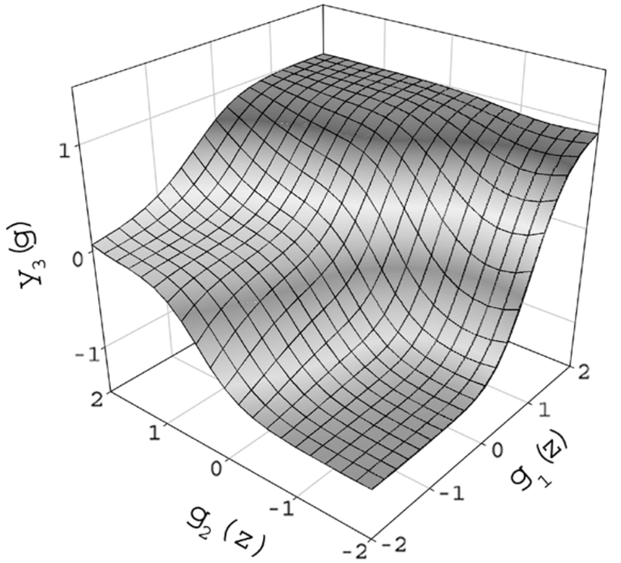

Fig. 5. Performance versus number of features for each of the five users (A, B, C, D, E) and for all together (A–E). Classifier K1: full triangles; classifier K2: open circle; and classifier K3: full circle. For each user, the classifiers were tested on ten sessions and each session consisted on average of 200 trials. The performance of each classifier represents its accuracy averaged over 2000 trials.

We used the sequential forward selection algorithm [3] for feature selection. While this method does not guarantee the best selection of features, it has modest computational costs. The results showed that more features can substantially improve performance. This improvement was strongest for methods K2 and K3. Comparing the performance of K2 (or K3) with K1, the results showed that the performance of K2 (or K3) was significantly higher than that of K1 for more than three features. Fig. 3 demonstrates this with the example of one session. The decision boundaries of the classifier K1 are two parallel hyperplanes. In the graph, it can be seen that this is a strong restriction (i.e., less trials were correctly classified). K2 and K3 have more degrees of freedom. K2 uses three hyperplanes that are not necessarily parallel and K3 uses two nonlinear decision boundaries. Fig. 5 visualizes the discriminant function y_3 as a function of the extracted features g_1 and g_2. It can be seen that the nonlinear decision boundaries in Fig. 3 were obtained by applying (12) on the graph, i.e., cutting the graph horizontally at the level of the thresholds.

The results in Fig. 4 showed that the performance difference between K2 (or K3) and K1 increased with the addition of features. Since it has been shown that K2 (or K3) is significantly better than K1, we conclude that the optimal decision boundaries are not parallel (K2) or are nonlinear (K3) for more than three features and that any linear 1-D discriminant function would perform worse than K2 (or K3). However, as demonstrated in Fig. 3, the classes show considerable overlap, which makes discrimination between K2 and K3 difficult.

The classifiers K2 and K3 operate in the 2-D extracted feature space (9). Each dimension of this extracted feature space was obtained by a linear discriminant function. The number of weights of these functions was equal to the number of features plus one. Therefore, the demand for data is a linear function of the number of features. Similar, the number of weights used for the classifier K1 is a linear function of the number of features (7). For some users, optimal or near optimal performance could be obtained with less than 10–20 features. However, using 10–20 features, optimal or near optimal performance could be obtained for each user and averaged over users, performance became a maximum.

The classifiers K2 and K3, therefore, improved offline performance because their decision boundaries have more degrees of freedom compared to a 1-D linear method (like K1 or the method that is currently used online).

The statistical approach based on Bayes theorem discussed in this study (11) is not the only way to obtain 1-D nonlinear discriminant functions (e.g., others include multilayer perception (MLP) [1] or radial basis functions [1]) but it is computationally simple and effective.

A. Online Performance Versus Offline Simulation

The data used in this study were collected during online operation of the Wadsworth BCI system. In online operation, the parameters used to convert EEG into cursor movements are continually adjusted according to the most recent data and the user continually adapts to these adjustments. In contrast, in offline simulation, while the parameters are continually adjusted according to the most recent data, the performance of the user is simulated by the subsequent data, which is not adapted to these adjustments. Thus, evaluating a method in offline simulation provides an imperfect estimate of its online performance. The methods evaluated here must be tested online to determine their actual online performance. Since the user will probably adapt to the new parameters, it is likely that the offline test is too conservative, i.e., that the online results will be better.

B. Implementation in the BCI

In online operation of the current BCI, the trial end is determined by the termination criterion (i.e., Section II-B6) and the trial outcome is determined according to the cursor position at the end of the trial. In this study, the methods tested were used to determine the trial outcome according to trial means. Feedback or cursor movements were not necessary for offline classification since there was no user interacting. In contrast, in the current online application a cursor function is defined where the trial outcome corresponds to the terminal cursor position.

While the discussed methods do ultimately need to be tested online, they do suggest possibilities for such online research.

1) Targets: For the 1-D cursor functions ν_1 and ν_3, the cursor position at the end of the trial is equivalent to the discriminant function y_1 or y_3 multiplied by the trial length. The middle box on the screen could then be defined by the thresholds $\mathrm{th}_{\mathrm{Top/Middle}}$ and $\mathrm{th}_{\mathrm{Middle/Bottom}}$ multiplied by a fixed trial length. At the end of a trial, selection according to the cursor position would be equivalent to the classification result of K1 or K3.

For the classification of the cursor position ν_2, the classifier K2 could be used. The input $\underline{g}$ would be replaced by the cursor position ν_2 divided by the trial length. As a consequence, the screen would be divided into three areas similar to Fig. 3, each associated with one target. Then, the cursor position at the end of the trial would indicate the target selection made by the classifier K2.

2) Feedback: The cursor function for K1 is similar to the cursor function that is currently used online and should give similar feedback in online operation. However, the offline results indicate that there is no improvement to be expected.

The cursor function ν_2 performs 2-D cursor movements for the classifier K2. In a preliminary study, the Wadsworth BCI has been applied to a 2-D four-choice task [22]. The cursor position ν_2 of K2 is similar to the cursor function used in that study: both are the sum of 2-D cursor movements derived by two linear equations. The only difference is that ν_2 uses more features and the coefficients of the linear equations are derived systematically. Given

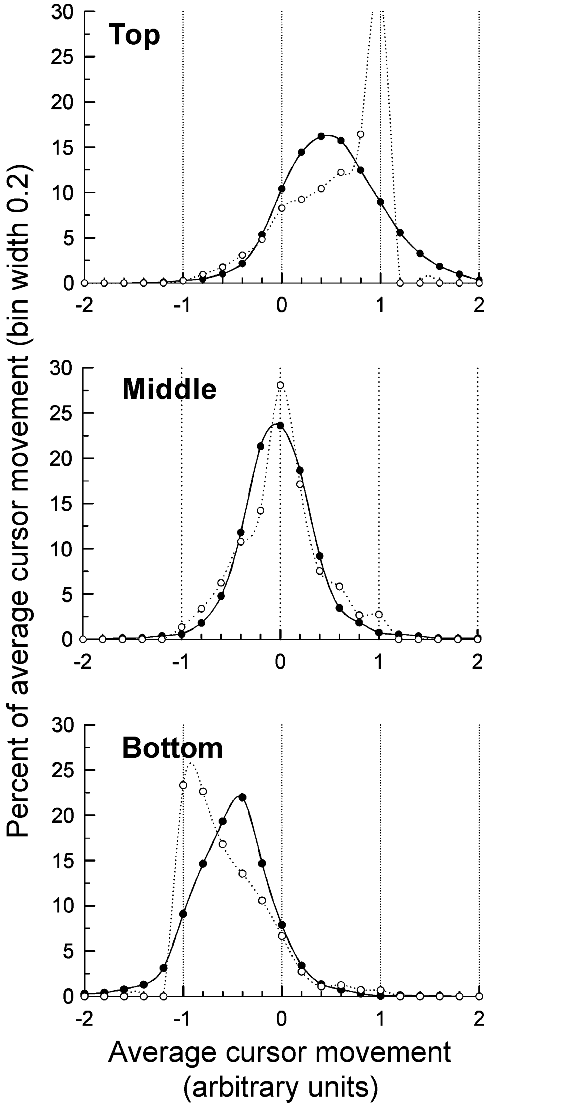

Fig. 6. Percentage of cursor movement for top/middle/bottom target. Open circle: nonlinear cursor movements $\Delta\nu_3$, full circle: linear cursor movements $\Delta\nu_1$.

this initial demonstration that a cursor can be controlled in two dimensions and the present offline results with the classifier K2, it is probable that K2 in combination with a 2-D feedback would significantly improve online performance of the three-box task.

For the classifier K3, the cursor function is a 1-D nonlinear function of EEG-features and time. Therefore, a cursor movement can only be determined by subtracting the current cursor position from the next. The difference between the r^2 values of ν_1 and ν_3 was small. In general, the value r^2 is also sensitive to the distribution of the cursor movements (e.g., a few, very large movements toward the target would increase r^2 more than they would increase performance). Fig. 6 shows that the cursor movements $\Delta\nu_1$ were roughly normally distributed. In contrast, the distributions of the movements $\Delta\nu_3$ were skewed and had peaks at 1, 0, and -1. In an application online, these peaks might not be a disadvantage. They occur because y_3 emphasizes values near 1, 0, and -1. For top and bottom targets, these peaks were caused by cursor movements of trials that were hits; the cursor moved with maximum speed (i.e., 1 or -1) in the direction of the target. Fig. 5 shows that the average cursor movement (i.e., this is y_3) of a trial is limited to a range between 1 and -1. For the middle target, the peaks at 1 and -1 were derived from cursor movements of trials that were misses while the peak at zero was derived from trials that were hits; the cursor moved to the center and stayed until the trial was over.

Given the higher r^2 value for cursor movements, the higher number of movements toward the target and the significantly better classification results of K3 compared to K1, it is most likely that the method K3 (including feedback) will substantially improve online performance of the BCI.

C. Other BCI Projects

Linear classifiers are widely used for two class problems, (e.g., [15] and [16]). In the present study, we used linear classifiers for the classification between each pair of classes (i.e., $y_{\text{Top/Middle}}$, $y_{\text{Top/Bottom}}$, $y_{\text{Middle/Bottom}}$). Compared to nonlinear classifiers (e.g., to neural networks), linear classifiers have less degrees of freedom (i.e., weights). While this can decrease performance, fewer weights also need less data for training. This can certainly be an advantage. For the current study, we did not find any neural network based classifier (e.g., MLP [1]) in a two-class situation that performed better than a linear classifier. Other studies have used neural network based classifiers for two class problems. For example, Pfurtscheller *et al.* [15] used the Learning Vector Quantisation (LVQ) algorithm to classify trials. Feedback was given at the end of the trial. Penny *et al.*. [14] used a logistic regression model trained using a Bayesian evidence framework to control a cursor. This led to more robust cursor movements because uncertain movements could be rejected.

The three-class problem is different from the two-class problem. The results of this study showed that the three classes are not linearly dependent. Nonlinear classifiers can adapt to this nonlinearity but simple linear classifiers (e.g., K1) do not. Milan *et al.* used a statistical approach to the three class problem [17]. Each trial was classified as the class with the highest probability given the EEG. This method is similar to the classifier K3 discussed in the present study except that it does not provide continuous feedback and does not base the classification on a 1-D cursor position.

Feature selection is usually not performed for linear classifiers (e.g., [15] and [16]). The number of weights is a linear function of the number of features. For this reason, useless features have little impact on performance. For nonlinear methods, this issue can be more crucial since the number of weights is usually an exponential function of the number of features (e.g., Pfurtscheller *et al.* used the distinction sensitive learning vector quantization [(DSLVQ), an LVQ-based algorithm] [25] for weighting features by their importance [15]). In the present study and some other studies (e.g., [16] and [17]), the features used were spectral bands of spatially filtered EEG signals. Other studies used autoregressive (AR) parameters (e.g., [14] and [15]) or common spatial filters (e.g., [15]). The Wadworth BCI calculates all spectral bands from AR-parameters II-A. This indicates that the same information is contained in both representations (i.e., AR-parameters and spectral bands) so that classification using frequency bands and classification using AR-parameters are similar approaches.

VI. Conclusion

This study used offline analysis of data collected from subjects selecting among three choices in an EEG-based cursor movement task. Currently, cursor movements occurs in one dimension as a linear function of EEG features. For a three-box task, this function assumes that the EEG patterns for cursor control (i.e., move cursor up, stop cursor, move cursor down) are linearly dependent. We presented two methods for classification and feedback (i.e., 2-D linear or 1-D nonlinear) that had more degrees of freedom. An offline analysis with these methods performed significantly better than a linear 1-D cursor function. Performance improved with features up to a point and deteriorated then slightly. The number of optimal features was user dependent. Averaged over the users, maximum performance could be obtained with 10–20 features. These results suggest that online BCI performance can be improved by using more of the information available in the EEG.

References

[1] C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford, U.K.: Claredon, 1995.

[2] N. Birbaumer, "Slow cortical potentials: Their origin, meaning, and clinical use," in *In Brain and Bahavior Past, Present, and Future*. Tilburg, Germany: Tilburg Univ. Press, 1997, pp. 25–39.

[3] P. A. Devijer and J. Kittler, *Pattern Recognition: A Statistical Approach*. Englewood Cliffs, NJ: Prentice-Hall, 1982.

[4] G. E. Fabiani, D. J. McFarland, J. R. Wolpaw, and G. Pfurtscheller, "EEG-based brain computer communication: Improvement by Bayesian discriminant analysis," in *Proc. Soc. Neuroscience 29th Annu. Meeting*, vol. 25, Miami Beach, FL, 1999, p. 1413.

[5] L. A. Farwell and E. Donchin, "Talking off the top your head: Toward a mental prothesis utilizing event-related brain potentials," *Electroencephalogr. Clin. Neurophysiol.*, vol. 70, pp. 510–523, 1988.

[6] G. Florian and G. Pfurtscheller, "Dynamic spectral analysis of event-related EEG data," *Electroencephalogr. Clin. Neurophysiol.*, vol. 95, pp. 393–396, 1995.

[7] M. J. Foster, D. J. McFarland, and J. R. Wolpaw, "Improvement in EEG-based brain-computer communication by use of additional recording locations," *Rehab. Eng. Soc. North. Amer.*, vol. 15, pp. 687–689, 1995.

[8] G. Golumb and W. Kahan, "Calculating the singular values and pseudo-inverse of a matrix," *SIAM Numer. Anal.*, vol. B 2, no. 2, pp. 205–224, 1965.

[9] D. J. McFarland, G. W. Neat, R. F. Read, and J. R. Wolpaw, "An EEG-based method for graded cursor control," *Psychobiol.*, vol. 21, pp. 77–81, 1993.

[10] D. J. McFarland, A. T. Lefkowicz, and J. R. Wolpaw, "Design and operation of an EEG-based brain-computer interface (BCI) with digital signal processing technology," *Behav. Res. Meth. Instrum. Comput.*, vol. 29, pp. 337–345, 1997.

[11] D. J. McFarland, L. M. McCane, S. V. David, and J. R. Wolpaw, "Spatial filter selection for EEG-based communication," *Electroencephalogr. Clin. Neurophysiol.*, vol. 103, pp. 386–394, 1997.

[12] D. C. Montgomery, *Design and Analysis of Experiments*, 4th ed. New York: Wiley, 1996.

[13] E. Niedermeyer and F. Lopez de Silva, *Electroencephalography: Basic Principles, Clinical Applications and Related Fields*, 3rd ed. Baltimore, MD: Williams & Williams, 1993.

[14] W. D. Penny, S. J. Roberts, E. A. Curran, and M. J. Stokes, "EEG-based communication: A pattern recognition approach," *IEEE Trans. Rehab. Eng.*, vol. 8, pp. 214–215, June 2000.

[15] G. Pfurtscheller, C. Neuper, C. Guger, W. Harkam, H. Ramoser, A. Schlögl, B. Obermaier, and M. Pregenzer, "Current trends in Graz brain-computer interface (BCI) research," *IEEE Trans. Rehab. Eng.*, vol. 8, pp. 216–219, June 2000.

[16] F. Babiloni, F. Cincotti, L. Lazzarini, J. Millán, J. Mourino, J. Varsta, J. Heikkonen, L. Bianchi, and M. G. Marciani, "Linear classification of low-resolution EEG patterns produced by imagined hand movements," *IEEE Trans. Rehab. Eng.*, vol. 8, pp. 186–188, June 2000.

[17] J. Milán, J. Mourino, F. Babiloni, F. Cinotti, M. Varsta, and J. Heikkonen, "Local neural classifier for EEG-based recognition of mental tasks," in *Proc. IEEE–INNS–ENNS Int. Conf. Neural Networks*, vol. 3, Como, Italy, 2000, pp. 632–636.

[18] G. Pfurtscheller, D. Flotzinger, and J. Kalcher, "Brain-computer interface–A new communication device for handicapped persons," *J. Microcomp. Appl.*, vol. 16, pp. 293–299, 1993.

[19] H. Ramoser, J. R. Wolpaw, and G. Pfurtscheller, "EEG-based communication: Evaluation of alternative signal prediction methods," *Biomed. Technol.*, vol. 42, pp. 226–233, 1997.

[20] F. Sharbrough, G. E. Chatrian, R. P. Lesser, H. Luders, M. Nuwer, and T. W. Picton, "American electroencephalographic society guidelines for standard electrode nomenclature," *J. Clin. Neurophysiol.*, vol. 8, pp. 200–202, 1991.

[21] T. M. Vaughan, J. R. Wolpaw, and E. Donchin, "EEG-based communication: Prospects and problems," *IEEE Trans. Rehab. Eng.*, vol. 4, pp. 425–430, Dec. 1996.

[22] J. R. Wolpaw and D. J. McFarland, "Multichannel EEG-based brain-computer communication," *Electroencephal. Clin. Neurophysiol.*, vol. 90, pp. 444–449, 1994.

[23] D. J. McFarland and J. R. Wolpaw, "EEG-based communication and control: Speed-accuracy relationships," *Appl. Psychophysiol. Biofeedback*, vol. 28, pp. 217–231, 2003.

[24] D. Flotzinger, G. Pfurtscheller, C. Neuper, and J. Berger, "Classification of nonaveraged EEG data by learning vector quantization and the influence of signal preprocessing," *Med. Biol. Eng. Comput.*, vol. 32, pp. 571-–576, 1994.

[25] M. Pregenzer, G. Pfurtscheller, and D. Flotzinger, "Automated feature selection with a distinction sensitive learning vector quantizer," *Neurocomp.*, vol. 11, pp. 19–29, 1996.

Georg E. Fabiani received the MSc degree in telematics (computer science) from the University of Technology Graz, Graz, Austria, in 2000.

From November 1998 to August 1999, he was a Visiting Scientist at the Wadsworth Center for Laboratories and Research, New York State Department of Health, Albany. He is currently a Software Engineer for binuscan, Monaco. His research interests include the development of image processing algorithms.

Dennis J. McFarland received the B.S. and Ph.D. degrees from the University of Kentucky, Lexington, in 1971 and 1978, respectively.

He is currently a Research Scientist with the Wadsworth Center for Laboratories and Research, New York State Department of Health, Albany. His research interests include the development of EEG-based communication and auditory psychophysics.

Jonathan R. Wolpaw received the A.B. degree from Amherst College, Amherst, MA, in 1966 and the M.D. degree from Case Western Reserve University, Cleveland, OH, in 1970. He completed a residency in neurology at the University of Vermont, Burlington, and fellowship training in neurophysiology research at the National Institutes of Health (NIH), Washington, DC.

He is currently the Chief of the Laboratory of Nervous System Disorders and a Professor at the Wadsworth Center of the New York State Departement of Health and the State University of New York, Albany. His research interests include operant conditioning of spinal reflexes as a new model for defining the plasticity underlying a simple form of learning in vertebrates, and development of EEG-based communication technology for those with severe motor disabilities. His research has been funded for many years by the NIH and many private foundations, and has received extensive recognition including a number of important awards.

Gert Pfurtscheller (M'00) received the M.S. and Ph.D. degrees in electrical engineering from the Graz University of Technology, Graz, Austria.

He is a Professor of medical informatics, Director of the Institute of Human-Computer Interfaces, Graz University of Technology, and Director of the Ludwig Boltzmann-Institute for Medical Informatics and Neuroinformatics. His research interests include functional brain topography using event-related desynchronization, the design of brain-computer communication systems, and navigation in virtual environments by a brain-computer interface. He has authored more than 300 publications in peer-reviewed journals and published four books.

Dr. Pfurtscheller is a member of the Austrian Academy of Science.