LOW-COMPLEXITY, MULTI-CHANNEL, LOSSLESS AND NEAR-LOSSLESS EEG COMPRESSION

Ignacio Capurro, Federico Lecumberry, Álvaro Martín,
Ignacio Ramírez, Eugenio Rovira, Gadiel Seroussi

Universidad de la República
Facultad de Ingeniería
Julio Herrera y Reissig 565, Montevideo, Uruguay

ABSTRACT

Current EEG applications imply the need for low-latency, low-power, high-fidelity data transmission and storage algorithms. This work proposes a compression algorithm meeting these requirements through the use of modern information theory and signal processing tools (such as universal coding, universal prediction, and fast online implementations of multivariate recursive least squares), combined with simple methods to exploit spatial as well as temporal redundancies typically present in EEG signals. The resulting compression algorithm requires $O(1)$ operations per scalar sample and surpasses the current state of the art in near-lossless and lossless EEG compression ratios.

***Index Terms*—** EEG compression, lossless compression, near-lossless compression, low-complexity

1. INTRODUCTION

Data compression is always of paramount importance when dealing with data sources which produce large amounts of data, due to the potentially large savings in storage and/or transmission costs. Electroencephalography (EEG) is no exception, with modern equipments that produce signals of up to 256 channels of up to 32 bits per sample (bps) each, sampled at frequencies up to 1000hz.

Some important applications of EEG include real-time monitoring of patients and individuals in their everyday activities (not confined to a bed or chair), and Brain-Computer Interfaces (BCI), where users can interact with computers or prosthetic body parts by activating certain regions of the brain. In such contexts, wireless and self-powered EEG acquisition devices are desirable, which imposes severe power consumption restrictions that call for efficient bandwidth use and simple embedded logic.

The requirements imposed by these applications, namely, low-power, and efficient on-line transmission of the EEG data, naturally lead to a requirement of low-complexity, and low-latency, compression algorithms. Also, as it is typical in medical applications, data acquired for clinical purposes (e.g., real-time monitoring of epileptic patients) is often required to be transmitted and/or stored without or at worst with very small distortion with respect to the data that was acquired by the sensors. This in turn leads to a need for lossless or near-lossless algorithms, where every decoded sample is guaranteed to differ by up to a preestablished bound from the original sample.

1.1. Main contribution

To satisfy the above requirements, the present work proposes a sequential, low-latency, low-complexity, lossless/near-lossless EEG compression algorithm, which combines several state-of-the art tools and developments in signal processing and data compression. The algorithm uses a novel method to exploit in combination the potential redundancy appearing across samples at different times (temporal redundancy) and the redundancy among samples obtained from different channels during the same sampling period (spatial redundancy).

The resulting algorithm surpasses current state of the art in lossless and near-lossless EEG compression, at a fraction of the computational cost required by competing algorithms such as [1–5]. The design relies on well-established theoretical tools from universal compression [6, 7] and prediction [8], and the results are backed by extensive experimentation.

1.2. Background on EEG compression

Recent years have seen significant activity in EEG compression and, in particular, in the lossless, multi-channel setting. As is commonly the case in lossless compression, most methods (both single and multichannel) are based on a predictive stage for removing temporal and/or spatial correlations. This produces a predicted signal, which is then subtracted from the original signal to obtain a prediction error that is encoded losslessly [2, 3, 9, 10]. Among prediction methods, typical choices include linear predictors [2], and neural networks [9]. After correlation is (hopefully) removed by the

This work was funded by CSIC, Universidad de la República.

prediction stage, residuals are encoded according to some statistical model. Typical choices include arithmetic and Huffman encoding. A non-predictive example, which uses a lossless variant of the typical transform-based methods used in lossy compression for temporal decorrelation is given in [1].

When moving to the multi-channel case, a spatial decorrelation stage is included to account for inter-channel redundancy. In this case, most lossless and near-lossless algorithms found in the literature resort to transform-based approaches to remove spatial redundancy. This includes the case just mentioned, [1], and the works [4, 5], where a lossy transform is used for simultaneous spatio-temporal decorrelation, and the residuals are encoded losslessly or near-losslessly. In [4] this is done using a wavelet transform, whereas [5] uses a PCA-like decomposition. As an example of a non-transform based method, the work [11] decorrelates a given channel by subtracting, from each of its samples, a fixed linear combination of the samples from neighboring electrodes corresponding to the same time slot.

In general, the focus of modern multi-channel lossless EEG compression algorithms lies in the decorrelation stage. In relation to the objectives posed in this work, we note that some of the aforementioned algorithms require a number of operations that is superlinear in the number of channels and, in some cases [4, 5], also superlinear in the number of time samples per channel. The adaptive transform methods also have the drawback of having to perform more than one pass over the data to be compressed, thus being unsuitable for real-time transmission of EEG data.

In contrast, in Section 2, we propose a compression algorithm whose execution time is linear in both the number of channels and time samples requiring an amount of memory that is linear in the number of channels. Moreover, since it uses a causal predictive strategy both for temporal and spatial decorrelation, it is sequential and exhibits low latency (the samples are encoded as soon as they are received). In Section 3, we provide experimental evidence on its performance, along with a detailed discussion and conclusions based on such evidence.

2. STATISTICAL MODELING AND ENCODING

We consider a discrete time m-channel EEG signal, $m > 1$. We denote by $x_i(n)$ the $i-$th channel sample at time instant n, $n \in \mathbb{N}$. We assume that all signal samples are quantized to integer values in a finite interval $\mathcal{X}$.

We follow a predictive coding scheme, in which a *prediction* $\hat{x}_i(n)$ is sequentially calculated for each sample $x_i(n)$, and this sample is described by encoding the *prediction error*, $\epsilon_i(n) \triangleq x_i(n) - \hat{x}_i(n)$. The sequence of sample descriptions is *causal*, i.e., the order in which the samples are described, and the definition of the prediction $\hat{x}_i(n)$, are such that the latter depends solely on samples that are described before sample $x_i(n)$. Thus, a decoder can sequentially calculate $\hat{x}_i(n)$, decode $\epsilon_i(n)$, and add these values to reconstruct $x_i(n)$. A near-lossless encoding is readily derived from this scheme by quantizing the prediction error $\epsilon_i(n)$ to a value $\tilde{\epsilon}_i(n)$ that satisfies $|\epsilon_i(n) - \tilde{\epsilon}_i(n)| \leq \delta$, for some preset parameter δ. After adding $\tilde{\epsilon}_i(n)$ to $\hat{x}_i(n)$, the decoder obtains a sample approximation, $\tilde{x}_i(n)$, whose distance to $x_i(n)$ is at most δ. In this case, the prediction $\hat{x}_i(n)$ may depend on the approximation of previously described samples, but not on the exact sample values, which are not available to the decoder.

The aim of the prediction step is to produce a sequence of prediction errors that, according to some preestablished probabilistic model, exhibit typically a low empirical entropy, which is then exploited in a coding step to encode the data economically. In our encoder we use an adaptive linear predictor, in line with the classical use of autoregressive models as a statistical modeling tool for EEG data [12]. We model prediction errors by a *two-sided geometric distribution* (TSGD), for which we empirically observe a good fitting to the data, and can be efficiently encoded with adaptive Golomb codes [6, 13].

In an (independent channel) *autoregressive model* (AR) of *order* p, $p \geq 0$, every sample $x_i(n)$, $n > p$, is the result of adding independent and identically distributed noise to a prediction

$$\hat{x}_i^p(n) = \sum_{k=1}^{p} a_{i,k} x_i(n-k)\,, \quad 1 \leq i \leq m\,, \tag{1}$$

where the real coefficients $a_{i,k}$ are *model parameters*, which determine, for each channel i, the dependance of $x_i(n)$ on previous samples of the same channel. The prediction in a *multivariate autoregressive model* (MVAR) is

$$\hat{x}_i^p(n) = \sum_{j=1}^{m} \sum_{k=1}^{p} a_{i,j,k} x_j(n-k)\,, \quad 1 \leq i \leq m\,, \tag{2}$$

where now the model parameters $a_{i,j,k}$ define, for each i, a linear combination of past samples from all channels and, thus, it may potentially capture both time and space signal correlation. Indeed, for EEG data, we experimentally observe that the *minimum mean square prediction error* (MSE), where model parameters are obtained as the solution to a least squares minimization, are in general significantly smaller for an MVAR model than an AR model. The average MSE, for an MVAR model of order 3, over all files and channels of the database DB1a (detailed in Section 3), for example, is 46% of the average MSE for an AR model of the same order.

Modern electroencephalographs, however, can record several tens of channels and, therefore, the number of model parameters in (2) may be very large. As a consequence, since these parameters are generally unknown a priori, MVAR models may suffer from a high statistical *model cost* [7], which may offset in practice the potential code length savings that could stem from accurate prediction. Nevertheless, a close inspection of EEG databases reveals that, for a fixed model order p, we obtain an MSE almost as small as that

obtained with an MVAR model if the prediction $\hat{x}_i^p(n)$ is a linear combination of the p most recent past samples from two channels, i, ℓ, where ℓ is a channel whose recording electrode is physically close to that of channel i. Moreover, the MSE for channel i is considerably lowered if, besides past samples from channels i, ℓ, we also use the sample at time instant n of channel ℓ to predict $x_i(n)$ (assuming causality is maintained, as will be discussed below), i.e.,

$$\hat{x}_i^p(n) = \sum_{k=1}^{p} a_{i,k} x_i(n-k) + \sum_{k=0}^{p} b_{i,k} x_\ell(n-k)\,. \quad (3)$$

The average MSE obtained with (3) and $p = 3$ for database DB1a, for example, is 60% of the average MSE for an MVAR model of order 3.

The predictor in our encoder relies on this observation. However, since, as mentioned, the sequence of sample descriptions must be causal with respect to the predictor, not all predictions $\hat{x}_i(n)$ can depend on a sample at time n. We next define an order of description that satisfies this constraint and seeks to minimize, for each channel except the one whose sample is described first, the distance between its electrode and the closest electrode of channels whose samples are described before.

Let T be a *minimum spanning tree* of the complete graph whose set of vertices is the set of channels, $\{1 \dots m\}$, and each edge (i, j) is weighted with the distance between electrodes of channels i, j. In other words, the sum of the distances between electrodes of channels i, j, over all edges (i, j) of T, is minimum among all trees with vertices $\{1 \dots m\}$. We distinguish an arbitrary channel r as the *root*, and we let the edges of T be oriented so that there exists a (necessarily unique) directed path from r to every other vertex of T. Since a tree has no cycles, the edges of T induce a causal sequence of sample descriptions, for example, by arranging the edges of T, $e_1 \dots e_{m-1}$, in a breadth-first traversal order. After describing a root channel sample, all other samples are described in the order in which their channel appear as the destination of an edge in the sequence $e_1 \dots e_{m-1}$. Notice that since T depends on the acquisition system but not on the signal samples, this description order may be determined off-line. The sample $x_r(n)$ is predicted based on samples of time up to $n-1$ of the channels r, i, where (r, i) is the edge e_1; all other predictions, $\hat{x}_i(n)$, $i \neq r$, depend on the sample at time n of channel ℓ and past samples of channels ℓ, i, where (ℓ, i) is an edge of T.

Algorithm 1 summarizes the proposed encoding. We let $\mathbf{x}_i(n) = x_i(1) \dots x_i(n)$ denote the sequence of the first n samples from channel i, and we let f_i be an integer valued prediction function to be defined.

As mentioned, we use adaptive Golomb codes for the encoding of prediction errors in steps 4 and 8. To this end, an independent set of prediction error statistics is maintained for each channel. The statistics collected up to time $n-1$ determine the *order* of a Golomb code [13], which is combined

```
1  for n = 1, 2, ... do
2      Let (r, i) be edge e_1 of T
3      x̂_r(n) = f_r(x_r(n−1), x_i(n−1))
4      Encode ε_r(n)
5      for k = 1 ... m − 1 do
6          Let (ℓ, i) be edge e_k of T
7          x̂_i(n) = f_i(x_i(n−1), x_ℓ(n))
8          Encode ε_i(n)
9      end
10 end
```

Algorithm 1: Coding algorithm.

with a Rice mapping [14] from integers to nonnegative integers to encode the prediction error at time n as in [15].

To complete the description of our encoder, we next define the prediction functions, f_i, $1 \leq i \leq m$, which are used in steps 3 and 7 of Algorithm 1. For a model order p, we let $\mathbf{a}_i^p(n) = \{a_{i,k}(n),\, b_{i,k}(n)\}$ denote the set of coefficient values, $a_{i,k}$, $b_{i,k}$, that, when substituted into (3), minimize the total weighted squared prediction error up to time n

$$E_i^p(n) = \sum_{j=1}^{n} \lambda^{n-j} \Big(x_i(j) - \hat{x}_i^p(j) \Big)^2 , \quad (4)$$

where λ, $0 < \lambda \leq 1$, is an exponential decay factor parameter. A *sequential linear predictor* of order p uses the coefficients $\mathbf{a}_i^p(n-1)$ to predict the sample value at time n as

$$\dot{x}_i^p(n) = \sum_{k=1}^{p} a_{i,k}(n\text{–}1) x_i(n\text{–}k) + \sum_{k=0}^{p} b_{i,k}(n\text{–}1) x_\ell(n\text{–}k)\,, \quad (5)$$

and, after having observed $x_i(n)$, updates the set of coefficients from $\mathbf{a}_i^p(n-1)$ to $\mathbf{a}_i^p(n)$ and proceeds to the next sequential prediction. This determines a total weighted *sequential prediction error* defined as

$$\mathcal{E}_i^p(n) = \sum_{j=1}^{n} \lambda^{n-j} \Big(x_i(j) - \dot{x}_i^p(j) \Big)^2 . \quad (6)$$

Notice that each prediction $\dot{x}_i^p(j)$ in (6) is calculated with a set of model parameters, $\mathbf{a}_i^p(j-1)$, which only depends on samples that are described before $x_i(j)$ in Algorithm 1. This model parameters vary, in general, with j (cf. (4)).

A suitable model order for EEG modeling may vary depending on signal characteristics [16]. However, numerous lattice algorithms have been proposed to efficiently calculate $\mathbf{a}_i^p(n)$ from $\mathbf{a}_i^p(n-1)$ simultaneously *for all* model orders p, up to a predefined maximum order P (see, e.g., [17] and references therein). Therefore, following [8], we do not fix nor estimate any specific model order but we instead average the predictions of all sequential linear predictors of order p, $0 \leq p \leq P$, exponentially weighted by their prediction performance up to time $n-1$. Specifically, for $i \neq r$, we define

$$f_i(\mathbf{x}_i(n-1), \mathbf{x}_\ell(n)) = \left\lfloor \frac{1}{M} \sum_{p=0}^{P} \mu_p(n) \dot{x}_i^p(n) \right\rfloor , \quad (7)$$

where $\lfloor \cdot \rceil$ denotes rounding to the nearest integer within the quantization interval $\mathcal{X}$, $\mu_p(n) = \exp\{-\frac{1}{c}\mathcal{E}_i^p(n-1)\}$, c is a constant that depends on $\mathcal{X}$ [8], and M is a normalization factor that makes $\mu_p(n)$ sum up to unity with p. The per-sample normalized prediction error of this predictor is asymptotically as small as the minimum normalized sequential prediction error among all linear predictors of order up to P [8]. The definition of f_r is analogous, removing the terms in $k = 0$ from (3) and (5), and letting the summation index p in (7) take values in the range $1 \leq p \leq P+1$.

In a near-lossless setting, steps 4 and 8 encode a quantized version, $\tilde{\epsilon}_i(n)$, of each prediction error, $\epsilon_i(n)$, defined as

$$\tilde{\epsilon}_i(n) = \text{sign}(\epsilon_i(n)) \left\lfloor \frac{|\epsilon_i(n)| + \delta}{2\delta + 1} \right\rfloor, \qquad (8)$$

which guarantees that the reconstructed value, $\tilde{x}_i(n) \triangleq \hat{x}_i(n) + \tilde{\epsilon}_i(n)(2\delta+1)$, differs by up to δ from $x_i(n)$. All model parameters and predictions are calculated with $\tilde{x}_i(n)$ in lieu of $x_i(n)$.

The overall encoding and decoding time complexity of the algorithm is linear in the number of encoded samples. Indeed, a Golomb encoding over a finite alphabet requires $O(1)$ operations and, since the set of predictions $\dot{x}_i^p(n)$, $0 \leq p \leq P$, can be recursively calculated executing $O(1)$ scalar operations per sample [17], the sequential computation of f_i requires $O(1)$ operations per sample. Regarding memory requirements, since each predictor and Golomb encoder requires a constant number of samples and statistics, the overall memory complexity of a fixed arithmetic precision implementation of the proposed encoder is $O(m)$.

3. EXPERIMENTS, RESULTS AND DISCUSSION

We evaluate our algorithm for lossless ($\delta = 0$) and near-lossless ($\delta > 0$) compression of several publicly available EEG databases:

- DB1a and DB1b [18,19]: 64-channel, 160Hz, 12bps EEG of 109 subjects using the BCI2000 system. Recordings are divided in 1-minute calibration (DB1a) and 2-minute motor imagery task (DB1b).
- DB2a and DB2b [20] (BCI Competition III[1]): 118-channel, 1000Hz, 16bps EEG of 6 subjects performing motor imagery tasks (DB2a). DB2b is a 100Hz down-sampled version of DB2a.
- DB3 [21] (BCI Competition IV[2]): 59-channel, 1000Hz, 16bps EEG of 7 subjects performing motor imagery tasks.
- DB4 [22]: 31-channel, 1000Hz, 16bps EEG of 15 subjects performing image classification and recognition tasks.

For each database we compress each data file separately and we calculate the *compression ratio* (CR) as kN/n, where

[1] http://bbci.de/competition/iii/

[2] http://bbci.de/competition/iv/

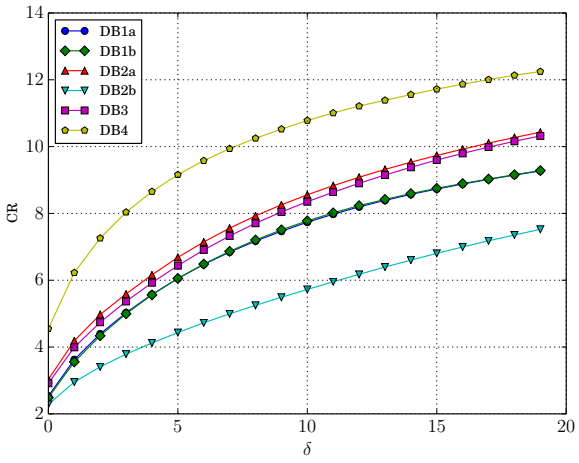

Fig. 1. CR obtained for each value of δ and all databases. The plots of DB1a and DB1b overlap.

N is the the sum of the number of scalar samples over all files of the database, k is the number of bits used in sample quantization, and n is the sum of the number of bits over all compressed files of the database. For testing we set $P = 7$, $\lambda = 0.99$, and a baseline value $c = 32$ (c is doubled whenever M falls below a small threshold, to improve numerical stability in (7)). The specific selection of the root channel r did not have any significant impact on the results of our experiments.

Table 1. CR of Alg. 1 and largest CR in [4,5] (in parenthesis).

δ	DB1a	DB1b	DB2a	DB2b	DB3	DB4
0	**2.54** (2.14)	**2.48**	**3.03**	**2.28** (1.94)	**2.92** (1.51)	**4.55**
5	**6.05** (4.78)	**6.05**	**6.68**	**4.43** (3.36)	**6.44** (2.27)	**9.16**
10	**7.74** (6.63)	**7.78**	**8.56**	**5.72** (4.15)	**8.35** (2.63)	**10.78**

The compression ratio, for each database and for different values of δ, is shown in Table 1 and plotted in Figure 1. In most of the reviewed literature, the performance of compression algorithms are tested with proprietary EEG data, which makes a direct comparison difficult. This is not the case with the recent works published in [4,5], where several compression algorithms are tested with EEG data taken from databases DB1a, DB2b, and DB3. Table 1 also shows, in parenthesis, the largest compression ratio reported in [4,5].

As Table 1 shows, the compression ratios obtained with the proposed method are significantly better than those obtained in [4,5] in all cases. Moreover, the algorithms described in [4,5] are multi-pass, and thus not suited to real-time compression of EEG data. Our algorithm is sequential and very efficient; compressing all files in databases DB1a and DB1b, which represent more than 2800 minutes of EEG recording, requires about 10 minutes in a modern PC. On the other hand, a progressive precision transmission application is suggested in [4,5], which we did not explore. Such application could be implemented by combining the proposed

near-lossless compression algorithm with a suitable compression of the signal residual.

REFERENCES

[1] Y. Wongsawat, S. Oraintara, T. Tanaka, and K.R. Rao, "Lossless multi-channel EEG compression," in *Proc. 2006 IEEE Int. Symp. Circuits and Systems*, May 2006.

[2] Z. Arnavut and H. Koak, "Lossless EEG signal compression," in *Proc. 5th Int. Conf. Soft Computing, Computing with Words and Perceptions in System Analysis, Decision and Control*, Sept 2009.

[3] K. Srinivasan, Justin Dauwels, and M. R. Reddy, "A two-dimensional approach for lossless EEG compression," *Biomedical Signal Processing and Control*, vol. 6, no. 4, pp. 387 – 394, 2011.

[4] K. Srinivasan, J. Dauwels, and M.R. Reddy, "Multichannel EEG compression: Wavelet-based image and volumetric coding approach," *IEEE Journal of Biomedical and Health Informatics*, vol. 17, no. 1, pp. 113–120, Jan 2013.

[5] J. Dauwels, K. Srinivasan, M.R. Reddy, and A. Cichocki, "Near-lossless multichannel EEG compression based on matrix and tensor decompositions," *IEEE Journal of Biomedical and Health Informatics*, vol. 17, no. 3, pp. 708–714, May 2013.

[6] N. Merhav, G. Seroussi, and M.J. Weinberger, "Coding of sources with two-sided geometric distributions and unknown parameters," *IEEE Trans. Inform. Theory*, vol. 46, no. 1, pp. 229–236, Jan 2000.

[7] J. Rissanen, "Universal coding, information, prediction, and estimation," *IEEE Trans. Inform. Theory*, vol. 30, pp. 629–636, July 1984.

[8] A.C. Singer and M. Feder, "Universal linear prediction by model order weighting," *IEEE Trans. Sig. Processing*, vol. 47, no. 10, pp. 2685–2699, Oct 1999.

[9] G. Antoniol and P. Tonella, "EEG data compression techniques," *IEEE Trans. Biomedical Engineering*, vol. 44, no. 2, pp. 105–114, Feb 1997.

[10] N. Memon, X. Kong, and J. Cinkler, "Context-based lossless and near-lossless compression of EEG signals," *IEEE Trans. Inform. Technology in Biomedicine*, vol. 3, no. 3, pp. 231–238, Sept 1999.

[11] Q. Liu, M. Sun, and R.J. Sclabassi, "Decorrelation of multichannel EEG based on Hjorth filter and graph theory," in *Proc. 6th Int. Conf. Signal Proc.*, Aug 2002, vol. 2, pp. 1516–1519 vol.2.

[12] J. Pardey, S. Roberts, and L. Tarassenko, "A review of parametric modelling techniques for EEG analysis," *Medical Engineering & Physics*, vol. 18, no. 1, pp. 2–11, Jan 1996.

[13] S. W. Golomb, "Run-length encodings," *IEEE Trans. Inform. Theory*, vol. 12, pp. 399–401, July 1966.

[14] R. F. Rice, "Some practical universal noiseless coding techniques — Parts I–III," Jet Propulsion Lab., Pasadena, CA, Tech. Reps. JPL-79-22, JPL-83-17, and JPL-91-3, Mar. 1979, Mar. 1983, Nov. 1991.

[15] M.J. Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS," *IEEE Trans. Image Processing*, vol. 9, no. 8, pp. 1309–1324, Aug 2000.

[16] F. Vaz, P. G. de Oliveira, and J. C. Principe, "A study on the best order for autoregressive EEG modelling," *International Journal of Bio-Medical Computing*, vol. 20, no. 1–2, pp. 41 – 50, 1987.

[17] G.-O. Glentis and N. Kalouptsidis, "A highly modular adaptive lattice algorithm for multichannel least squares filtering," *Signal Processing*, vol. 46, no. 1, pp. 47–55, Sept. 1995.

[18] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. Ch. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, 2000 (June 13).

[19] G. Schalk, D.J. McFarland, T. Hinterberger, N. Birbaumer, and J.R. Wolpaw, "BCI2000: a general-purpose brain-computer interface (BCI) system," *IEEE Trans. Biomedical Engineering*, vol. 51, no. 6, pp. 1034–1043, June 2004.

[20] G. Dornhege, B. Blankertz, G. Curio, and K. Muller, "Boosting bit rates in noninvasive EEG single-trial classifications by feature combination and multiclass paradigms," *IEEE Trans. Biomedical Engineering*, vol. 51, no. 6, pp. 993–1002, June 2004.

[21] B. Blankertz, G. Dornhege, M. Krauledat, K.-R. Müller, and G. Curio, "The non-invasive Berlin brain–computer interface: Fast acquisition of effective performance in untrained subjects," *NeuroImage*, vol. 37, no. 2, pp. 539 – 550, 2007.

[22] A. Delorme, G. A. Rousselet, M. J.-M Macé, and M. Fabre-Thorpe, "Interaction of top-down and bottom-up processing in the fast visual analysis of natural scenes," *Cognitive Brain Research*, vol. 19, no. 2, pp. 103 – 113, 2004.