Convolutional Neural Networks for P300 Detection with Application to Brain-Computer Interfaces

Hubert Cecotti and Axel Gräser

Abstract—A Brain-Computer Interface (BCI) is a specific type of human-computer interface that enables the direct communication between human and computers by analyzing brain measurements. Oddball paradigms are used in BCI to generate event-related potentials (ERPs), like the P300 wave, on targets selected by the user. A P300 speller is based on this principle, where the detection of P300 waves allows the user to write characters. The P300 speller is composed of two classification problems. The first classification is to detect the presence of a P300 in the electroencephalogram (EEG). The second one corresponds to the combination of different P300 responses for determining the right character to spell. A new method for the detection of P300 waves is presented. This model is based on a convolutional neural network (CNN). The topology of the network is adapted to the detection of P300 waves in the time domain. Seven classifiers based on the CNN are proposed: four single classifiers with different features set and three multiclassifiers. These models are tested and compared on the Data set II of the third BCI competition. The best result is obtained with a multiclassifier solution with a recognition rate of 95.5 percent, without channel selection before the classification. The proposed approach provides also a new way for analyzing brain activities due to the receptive field of the CNN models.

Index Terms—Neural network, convolution, gradient-based learning, spatial filters, brain-computer interface (BCI), electroencephalogram (EEG), P300.

1 Introduction

A Brain-Computer interface (BCI) is a direct communication pathway between a human brain and an external device. Such systems allow people to communicate through direct measurements of brain activity, without requiring any movement [1], [2], [3]. BCIs may be the only means of communication possible for people who are unable to communicate via conventional means because of severe motor disabilities like spinal cord injuries or like amyotrophic lateral sclerosis (ALS), also called Lou Gehrig's disease [2], [4]. Among noninvasive methods for monitoring brain activity, we consider in this paper electroencephalography (EEG) techniques. They have several practical qualities: Data can be easily recorded with relatively inexpensive equipment; they are the common solution for noninvasive BCIs.

A BCI is usually decomposed into four main parts that translate the neural signal processing. First, the signal is acquired via an amplifier. Then, the signal is processed and assigned to different classes, which denotes the different stimuli. Finally, the classes are sent to the output device components and the operating protocol links all the components. The signal classification component is composed of the brain signal features extraction and the translation of these signals into device commands. The EEG classification strategy depends on the stimulus and, thereby, the response to detect: event-related potentials, steady-state evoked potentials, motor imagery, or slow cortical potentials. The expected EEG drives the classification to some specific feature extraction methods.

Pattern recognition techniques are used for the classification and the detection of specific brain signals. Most of the effective solutions use machine learning models [5], [6], [7], [8]. Although neuroscience provides knowledge and guidelines about how to process and detect the expected signals, machine learning techniques allow modeling the signal variability over time and over subjects. Neural networks [9], [10], [11], [12], [13], [14], support vector machines (SVMs) [15], [16], and hidden Markov models [17], [18] have already been applied to BCI and EEG classification. Neural networks using backpropagation were used for the first time for readiness potential pattern recognition in [19], proving that neural networks can be used for classifying EEG and for tailoring a brain machine interface.

Most of the current techniques in the BCI community are based on SVMs. Gradient-based learning methods such as convolutional neural networks have been successfully used in character recognition and achieve the best recognition results in database such as MNIST [20], [21]. They are also used in speech processing [22]. In the case of character recognition, such models could offer a tolerance to geometric deformations, to some extent. The evaluation of their performance for the classification of an EEG signal, which possesses a high variability, is one topic of this study. Also, one interesting property of CNN models is the semantic of the weights once the network is trained. The receptive field/convolution kernel can be easily interpreted and can provide

• *The authors are with the Institute of Automation, University of Bremen, Otto-Hahn-Allee, NW1 28359 Bremen, Germany. E-mail: hcecotti@orange.fr, ag@iat.uni-bremen.de.*

Manuscript received 28 Nov. 2008; revised 25 Feb. 2009; accepted 3 June 2009; published online 16 June 2010.
Recommended for acceptance by L. Bottou.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2008-11-0821.
Digital Object Identifier no. 10.1109/TPAMI.2010.125.

a diagnostic about the type of high-level features to detect. We propose using CNN models and their combination, for the first time, for the detection of P300 waves.

The paper is organized as follows: The P300 wave, the oddball paradigm, and the database are presented in Section 2. The neural network is described in Section 3. Section 4 describes the different classifiers. Finally, the results and their discussion are detailed in Sections 5 and 6.

2 Background

2.1 The P300 Speller

The P300 wave is an event-related potential (ERP) which can be recorded via EEG. The wave corresponds to a positive deflection in voltage at a latency of about 300 ms in the EEG. In other words, it means that after an event like a flashing light, a deflection in the signal should occur after 300 ms. The signal is typically measured most strongly by the electrodes covering the parietal lobe. However, Krusienski et al. showed that occipital sites are more important [23]. Furthermore, the presence, magnitude, topography, and time of this signal are often used as metrics of cognitive function in decision making processes. If a P300 wave is detected 300 ms after a flashing light in a specific location, it means that the user was paying attention to this same location. The detection of a P300 wave is equivalent to the detection of where the user was looking 300 ms before its detection. In a P300 speller, the main goal is to detect the P300 peaks in the EEG accurately and instantly. The accuracy of this detection will ensure a high information transfer rate between the user and the machine. Farwell and Donchin introduced the first P300-BCI in 1988 [24], [25].

2.2 P300 Detection

There exist two types of classifications for the problem of P300-based BCI as illustrated in Fig. 1. These classification steps are sequential:

1. The detection of P300 responses. It corresponds to a binary classification: One class represents signals that correspond to a P300 wave, the second class is the opposite. For this classification problem, the creation of the ground truth can be quite challenging. Although the paradigm during the experiment allows knowing when a P300 response is expected, this response depends on the user. Indeed, although a P300 response can be expected at one particular moment, it is possible that the user does not produce a P300 response at the right moment as many artifacts can occur. The production of a P300 wave is not a phenomenon of consciousness; it is produced due to the flashing lights.
2. The character recognition. The outputs of the previous classification are then combined to classify the main classes of the application (characters, symbols, actions, ...). Whereas the ground truth of the first classification step remains uncertain, the ground truth of the character recognition problem can be created easily as the character to spell is clearly given to the subject. In the oddball paradigm, a character is defined by a couple (x, y). The flashing lights are on each row and column and not on each character. The character is supposed to correspond to the intersection of the accumulation of several P300 waves. The best accumulation of P300 waves for the vertical flashing lights determines the column of the desired character. The principle is the same for the horizontal flashing lights and the rows.

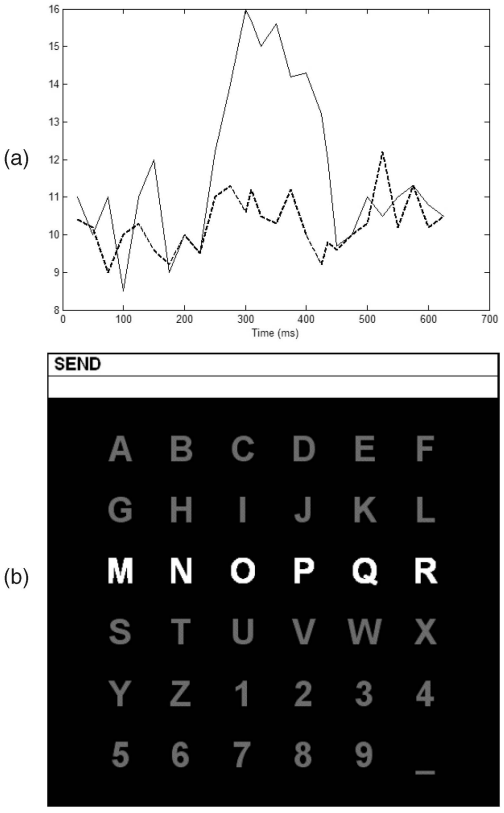


Fig. 1. The two classification problems. (a) P300 detection. (b) Character recognition.

2.3 Database

Data set II from the third BCI competition was used for testing the different models [26]. The database was initially provided by the Wadsworth Center, New York State Department of Health. This data set contains a complete record of P300 evoked potentials from two subjects. The signal was recorded in five sessions with the BCI2000 framework [27]. In these experiments, the subject was presented with a matrix of size 6×6. Each cell of the matrix contains a character: [A-Z], [1-9], and [_]. The main classification problem therefore has 36 classes. The subject's task was to sequentially focus on characters from a predefined word. The six rows and six columns of this matrix were successively and randomly intensified at a rate of 5.7 Hz. The character to select is defined by a row and a column. Thus, 2 out of 12 intensifications of rows/columns highlighted the expected character, i.e., 2 of the 12 intensifications should produce a P300 response.

During the experiment, the matrix was displayed for a 2.5 s period, and during this time, the matrix was blank: Each character had the same intensity. Then, each row and column in the matrix was randomly intensified for 100 ms. After intensification of a row/column, the matrix was blank for 75 ms. Row/column intensifications were block randomized in blocks of 12. The sets of 12 intensifications were repeated

TABLE 1
Database Size for Each Subject

	Training	Test
P300	2550	3000
no P300	12750	15000

15 times for each character epoch. All of the rows/columns were intensified 15 times. Therefore, 30 possible P300 responses should be detected for the character recognition.

Signals from the two subjects were collected from 64 ear-referenced channels. The signal was bandpass filtered from 0.160 Hz and digitized at 240 Hz [28]. The training database is composed of 85 characters, while the test database contains 100 characters. Each character epoch is supposed to contain two P300 signals, one for a row flash and one for the column flash. For the training database, the number of P300 to detect is $85 * 2 * 15$. The number of samples for both databases and for each subject is presented in Table 1.

2.4 Existing Systems

This section describes some of the best techniques that have been proposed during the third BCI competition. They also correspond to the state of the art for the P300 speller. These solutions are mostly based on multiclassifiers strategy. Some techniques use advanced signal processing methods for cleaning the data. Furthermore, it is not easy to compare the inner strength of one classifier as the inputs are often different. They vary in size in relation to the number of considered electrodes and the size of the time window describing a P300 wave. The results of the best methods will be compared with the proposed method in the last section.

- The solution proposed by Rakotomamonjy and Guigue [16] is based on an ensemble of SVMs. In this solution, the signal is extracted with a 667 ms time window after each stimulus. Then the signal is bandpass filtered with an 8-order filter with cutoff frequencies between 0.1 and 20 Hz. For each channel, the signal is defined by 14 features. The size of the input is 896 ($14 * 64$). The training database is partitioned into groups of 900 patterns. Each group is related to the spelling of five characters. Therefore, the training database is divided into 17 partitions. A linear SVM is trained on each partition and a channel selection procedure is performed. The channel selection algorithm is a recursive channel elimination based on criteria in relation to the confusion matrix of the validation test. The character recognition is achieved by summing all the scores of the SVMs. The row and column that get the highest score are considered as the coordinate of the character to detect.
- The method of Li Yandong from the Department of Automation and Department of Biomedical Engineering, Tsinghua University, China, is decomposed into three steps. First, data are preprocessing with bandpass filtering at 0.5-8 Hz. Then, eye movement artifacts are removed by using independent component analysis (ICA) for the whole data set. The classification is based on SVMs and bagging with patterns selected with a time window of 100-850 ms after a flashing light [29]. A subset of electrodes is selected prior to the classification. The final classification is achieved through the voting of multiple SVM classifiers contrary to other methods, which average the outputs.
- The technique of Zhou Zongtan from the Department of Automatic Control, National University of Defense Technology, China, is based on frequency filtering and principal component analysis (PCA) for the preprocessing steps. The feature selection uses t-statistic values at each data point in each channel. For the data, the author only keeps the extremum points of t-statistic values from each channel. The classification is performed by comparing the different features set.
- Ulrich Hoffmann from the Ecole Polytechnique Federale of Lausanne (EPFL), Switzerland, uses a gradient boosting method that is described in [30].
- Lin Zhonglin, Department of Automation, Tsinghua University, China, uses bagging with component classifier linear discriminant analysis (LDA) [29]. For each subject, the authors first create 150 training sets by drawing about 60 percent samples from the original training set. Then each of these data sets is used to train an LDA classifier. The final classification decision is based on the vote of each component classifier. For the input, the signal is bandpass filtered between 0.5 and 15 Hz and 10 channels are selected before the classification.

3 Convolutional Neural Network

The classifiers that are used for the detection of P300 responses are based on a convolutional neural network (CNN). This type of neural network is a multilayer perceptron (MLP) with a special topology and contains more than one hidden layer. This neural network is used for object recognition [31] and handwriting character recognition [21], [32]. It allows automatic features extraction within its layers and it keeps as input the raw information without specific normalization, except for scaling and centering the input vector. This kind of model has many advantages when the input data contain an inner structure like for images and where invariant features must be discovered. One interest on convolutional neural network is the possibility of including, inside the network, high-level knowledge that is directly related to the problem, contrary to kernel-based methods [20]. One other interest is to avoid hand-designed input features, which are not derived by the general problem. However, the topology of the network remains an empirical choice and depends on the application. The topology translates different successive signal/neural processing steps.

A classifier based on a CNN seems to be a good approach for EEG classification as the signal to detect contains a lot of variations over time and persons. For such a variable signal, architectures based on local kernels can be inefficient at representing functions that must be tolerant to many variations, i.e., functions that are not globally smooth [20]. The interest of the CNN is to directly classify the raw signal and to integrate the signal processing functions within the discriminant steps. Indeed, it is not always possible to know the type of features to extract. It is better to let the network

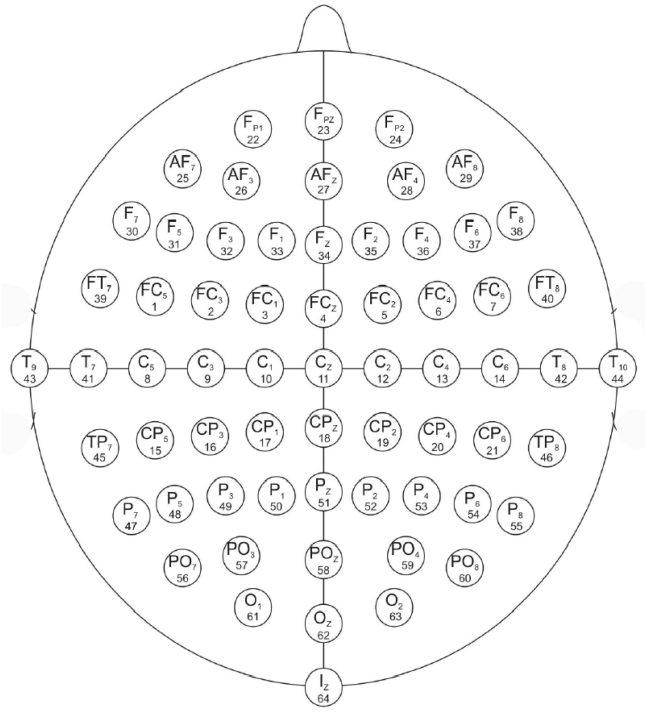


Fig. 2. Electrode map.

extract the most discriminant features by constructing high-level features throughout the propagation step.

3.1 Input Normalization

The inputs are the EEG signal values from the electrodes during TSs, $I_{i,j}$, $0 \leq i < N_{elec}$, $0 \leq j < SF * TS$. SF is the sampling frequency in hertz (Hz). The data are normalized in two steps. First, the EEG signal is subsampled to reduce the size of the data to analyze. The size is divided by two. It is now equivalent to a signal sampled at 120 Hz. Then, the signal is bandpass filtered between 0.1 and 20 Hz to keep only relevant frequencies but it is kept sampled at 120 Hz. Finally, the signal is normalized as follows:

$$I_{i,j} \leftarrow (I_{i,j} - \bar{I}_i)/(\sigma_i), \tag{1}$$

where $\bar{I}_i$ and σ_i are, respectively, the average value and the first deviation of the electrode i at the time j in TSs. The average and the standard deviation are based on each individual pattern and for each electrode. The input of the CNN is a matrix $N_{elec} \times N_t$, where N_t is the number of points that are considered for the analysis: $N_t = SF*TS$. N_t corresponds to the number of recorded samples in TSs with the sampling rate SF. When all of the electrodes are used, $N_{elec} = 64$. In the experiments, we set $N_t = 78$ that represents 650 ms. Each pattern represents a part of the signal starting after a flashing light and during 650 ms.

3.2 Neural Network Topology

The network topology is the key feature in the classifier. The network is composed of five layers, which are themselves composed of one or several maps. A map represents a layer entity, which has a specific semantic: Each map of the first hidden layer is a channel combination. The second hidden layer subsamples and transforms the signal in the time domain. The classifier architecture is presented in Fig. 3. The number of neurons for each map is presented between brackets; the size of the convolution kernel is between hooks. The order of the convolution is chosen in relation to what is traditionally done in BCI. First, optimal spatial filters/channel combinations are set, then the signal is processed in the time domain. The choice of the topology is also justified by the possibility of easily interpreting the trained convolution kernel, i.e., the receptive fields. In the proposed strategy, the kernels are vectors and not matrix, like in CNNs for image recognition. The reason is to not mix in one kernel features related to the space and time domain.

The network topology is described as follows:

- L_0: The input layer. $I_{i,j}$ with $0 \leq i < N_{elec}$ and $0 \leq j < N_t$.

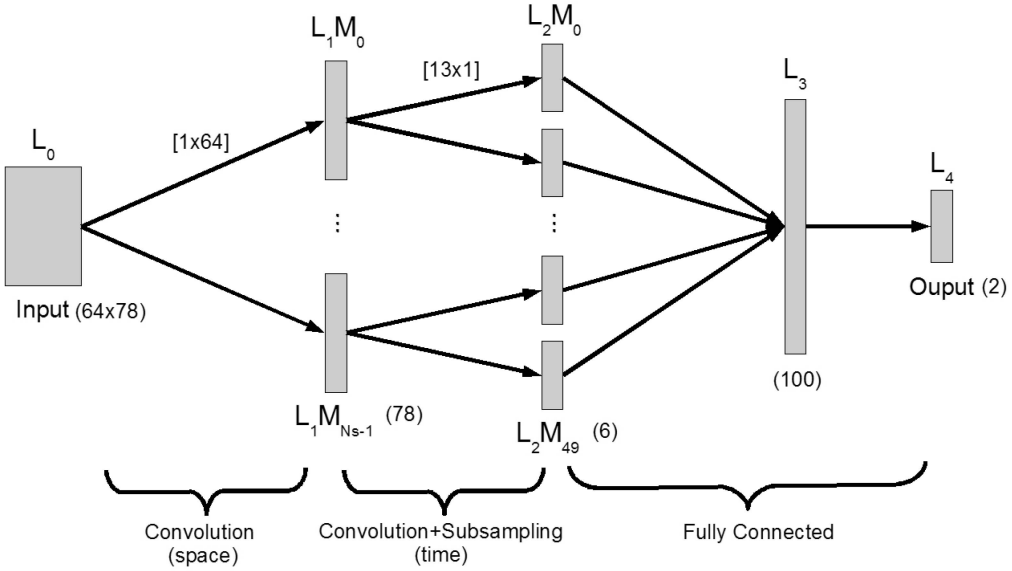


Fig. 3. Neural network architecture.

- L_1: The first hidden layer is composed of N_s maps. We define L_1M_m, the map number m. Each map of L_1 has the size N_t.
- L_2: The second hidden layer is composed of $5N_s$ maps. Each map of L_2 has six neurons.
- L_3: The third hidden layer is composed of one map of 100 neurons. This map is fully connected to the different maps of L_2.
- L_4: The output layer. This layer has only one map of two neurons, which represents the two classes of the problem (P300 and no P300). This layer is fully connected to L_3.

3.3 Learning

A neuron in the network is defined by $n(l, m, j)$, where l, m, and j are the layer, the map, and its position in the map, respectively. Its current value is $x^l_m(j)$, or $x^l(j)$ when there is only one map in the layer:

$$x^l_m(j) = f\big(\sigma^l_m(j)\big), \tag{2}$$

where f depends on the layer.

- This sigmoid function is almost linear between -1 and 1, $f(1) = 1$ and $f(-1) = -1$. The constants are set according to the recommendations described in [33]. It is used for the first two hidden layers, which represent convolution of the input signal:

$$f(\sigma) = 1.7159\ tanh\left(\frac{2}{3}\sigma\right). \tag{3}$$

- The classical sigmoid function is used for the two last layers:

$$f(\sigma) = \frac{1}{1 + exp^{-\sigma}}. \tag{4}$$

$\sigma^l_m(j)$ represents the scalar product between a set of input neurons and the weight connections between these neurons and the neuron number j in the map m in the layer l. We define $\sigma^l_m(j)$ for the four layers. L_1 and L_2 are convolutional layers, respectively, in the space and time domain. L_2, L_3, and L_4 can be considered as an MLP, where L_2 is the input layer, L_3 is the hidden layer, and L_4 is the output layer.

For L_1 and L_2, we can notice that each neuron of the map shares the same set of weights. The neurons of these layers are connected to a subset of neurons from the previous layer. Instead of learning one set of weights for each neuron, where the weights depend on the neuron position, the weights are learned independently to their corresponding output neuron.

- For L_1:

$$\sigma^1_m(j) = w(1, m, 0) + \sum_{i=0}^{i<N_{elec}} I_{i,j} w(1, m, i), \tag{5}$$

where $w(1, 0, j)$ is a threshold. A set of weights $w(1, m, i)$ with m fixed, $0 \le i < N_{elec}$, corresponds to a spatial filter, i.e., a channel. In this layer, there are $N_{elec} + 1$ weights for each map. This layer aims at finding the best electrodes combination for the classification. The convolution represents spatial filters. The convolution kernel has a size of $[1 \times N_{elec}]$.

- For L_2:

$$\sigma^2_m(j) = w(2, m, 0) + \sum_{i=0}^{i<13} x^1_m(j * 13 + i) w(2, m, i), \tag{6}$$

where $w(2, 0, j)$ is a threshold. This layer transforms the signal of 78 values into six new values in L_2. It reduces the size of the signal to analyze while applying an identical linear transformation for the six neurons of each map. This layer translates subsampling and temporal filters. The convolution kernel has a size of $[13 \times 1]$.

- For L_3:

$$\sigma^3(j) = w(3, 0, j) + \sum_{i=0}^{i<5N_s} \sum_{k=0}^{k<6} x^2_i(k) w(4, i, k), \tag{7}$$

where $w(4, 0, j)$ is a threshold. Each neuron of L_3 is connected to each neuron of L_2. L_2 and L_3 are fully connected. In this layer, each neuron has $N_sN_f + 1$ input weights. L_3 contains $100(5{*}6{*}N_s)$ input connections.

- For L_4:

$$\sigma^4(j) = w(4, 0, j) + \sum_{i=0}^{i<100} x^3(i) w(5, i), \tag{8}$$

where $w(4, 0, j)$ is a threshold. Each neuron of L_4 is connected to each neuron of L_3.

For each neuron, the input weights and the threshold are initialized with a standard distribution around $\pm 1/n(l, m, i)_{N_{input}}$. We define $n(l, m, i)_{N_{input}}$ as the number of inputs of $n(l, m, i)$. For layers L_1 and L_2, the learning rate γ is defined by

$$\gamma = \frac{2\lambda}{n(l, m, 0)_{N_{shared}} \sqrt{n(l, m, i)_{N_{input}}}}, \tag{9}$$

where $n(l, m, 0)_{N_{shared}}$ is the number of neurons that share the same set of weights and λ is a constant. For $L1$ and $L2$, each neuron on each map shares the same number of weights, the learning rate takes into account the number of neurons that share the same set of weights.

For layers L_3 and L_4, the learning rate is

$$\gamma = \frac{\lambda}{\sqrt{n(l, m, i)_{N_{input}}}}. \tag{10}$$

The learning algorithm for tuning the weights of the network uses the classical backpropagation. The weights are corrected due to a gradient descent. Each training epoch is composed of 95 percent of the training database which is effectively used for learning, whereas the remaining 5 percent is dedicated to the validation database. The training stops once the least mean square error is minimized on the validation database. The output layer is composed of two neurons, which represent the two classes. $x^4(0)$ and $x^4(1)$ represent, respectively, the absence and the

presence of a P300 wave. During the test, the detection of a P300 wave is defined by

$$E(X) = \begin{cases} 1 & \text{if } x^4(1) > x^4(0), \\ 0, & \text{otherwise,} \end{cases} \quad (11)$$

where X is the signal to classify and E is the classifier.

4 Classifiers

We present here seven classifiers based on the convolutional neural network that was presented in the previous section. This classifier will be used as the core of the different models. Among the presented classifiers, CNN-1, CNN-2, and CNN-3 are single classifiers whereas MCNN-1, MCNN-2, and MCNN-3 are based on a multiclassifiers strategy, like most of the efficient methods that achieve good results on P300 detection. For an input pattern P, we note E(P) the probability that a classifier determines P as being a P300 response (the values of the output layer are normalized to obtain the probabilities).

The first three classifiers are defined as follows:

- CNN-1: For the first classifier, the training was achieved with the whole training database. CNN-1 is based on the convolutional neural network described in Section 3. This classifier is the reference for further comparisons.
- CNN-2a: This classifier is identical to CNN-1, but it only uses eight electrodes instead of 64. The eight prefixed channels correspond to the location: F_Z, C_Z, P_Z, P_3, P_4, PO_7, PO_8, and O_Z in the international 10-20 system of measurement [28]. These channels were chosen in relation to the guideline provided during the BCI tutorial in Utrecht, Holland, 2008. This classifier aims at providing a realistic view of the accuracy that can be obtained with a relatively small number of electrodes. This classifier translates a more pragmatic approach toward the use of P300-BCIs.
- CNN-2b is identical to CNN-2A. However, the eight prefixed channels are determined in relation to the weight analysis of CNN-1 as described in Section 4.1. CNN-1 must be evaluated first to determine the ideal feature set for CNN-2B, i.e., the ideal set of electrodes.
- CNN-3: This classifier is identical to CNN-1, but it only has one map in the first hidden layer which translates one single spatial filter. In this case, the classifier is based on only one channel. With this classifier, it is easier to interpret the meaning of the learned weights in the first hidden layer. This classifier can also provide information about the relevance of a multichannels classification scheme, where there exist several spatial filters for improving the classification.

The three multiclassifiers systems are based on the same classifiers: CNN-1. Only the training database differs for the three methods. For each multiclassifiers system, the average is used for fusing the output of each classifier [34].

- MCNN-1: The database is not homogeneous in the distribution of the patterns that represent a P300 and the others. In fact, the database contains five times more patterns that are not a P300. MCNN-1 is composed of five classifiers. Each classifier is trained on a different database. Each training database contains all the P300 patterns and a fifth of the non-P300 patterns from the main training database. The set of patterns that does not represent a P300 is cut into five equal parts. Thus, each classifier is trained with an equal number of P300 and non-P300 patterns.
- MCNN-2: The signals contained in the database can vary in quality and they can also represent different issues that one single classifier cannot model. For the particular problem of P300 detection, it is possible that the subject was not really focused at some times during the experiment. During these moments, the subject may not produce a reliable P300 or it might produce a P300 at an undesired moment. In such a case, we can expect the presence of outliers and many errors in the labeling. MCNN-2 is also composed of five classifiers. The training database is cut into five equal parts which represent five consecutive sequences in the EEG record. Each classifier of MCNN-2 is trained on one sequence. Such classifiers can model different types of P300 over time.
- MCNN-3: This system is composed of three classifiers, like CNN-1. As the weights are initialized randomly, the creation of different classifiers like CNN-1 may lead to different classifiers. This classifier aims at improving the reliability of CNN-1. In case of nonimprovement, this classifier will show that the random initialization leads to equivalent classifiers.

4.1 Feature Selection for CNN-2b

The choice of the electrodes in CNN-2b is based on the weight analysis of the first hidden layer in CNN-1. Indeed, as the first hidden layer corresponds to the creation of the channel combination, it is possible to extract information about the most relevant electrodes once the network is trained. When a weight is close to 0, then it means that its discriminant power is low. At the opposite, weights with a high absolute value denote a high discriminant power, and therefore, a relevant electrode for the classification. We define the power of the electrode i by

$$\xi_i = \sum_{j=0}^{j=N_s} |w(i,j)|, \quad (12)$$

where $0 \leq i < N_{elec}$ and $w(i,j)$ represents the weight of a link between any neuron of the map j to the electrode i at any time. ξ_i is the combination of the different maps that compose the network. It is possible to create a new classifier with a prefixed number of n electrodes by selecting n electrodes, which correspond to the n higher ξ values. CNN-2b is instantiated with $\mathrm{n} = 8$, to be compared with CNN-2a.

4.2 Complexity

For each convolutional layer, the weights are shared for every neuron within one map in the first two hidden layers. It therefore reduces the number of free parameters in the network. For each layer, the number of parameters (the number of weights and thresholds for all the neurons) is defined as follows:

TABLE 2
Results of the P300 Detection for Subject A

Method	TP	TN	FP	FN	Reco.	Recall	Precision	Silence	Noise	Error	F-measure
CNN-1	2021	10645	4355	979	70.37	0.674	0.317	0.326	0.683	1.778	0.431
CNN-2a	1852	10403	4597	1148	68.08	0.617	0.287	0.383	0.713	1.915	0.392
CNN-2b	1835	10554	4446	1165	68.83	0.612	0.292	0.388	0.708	1.870	0.395
CNN-3	1827	10677	4323	1173	69.47	0.609	0.297	0.391	0.703	1.832	0.399
MCNN-1	2071	10348	4652	929	68.99	0.690	0.308	0.310	0.692	1.860	0.426
MCNN-2	1951	10179	4821	1049	67.39	0.650	0.288	0.350	0.712	1.957	0.399
MCNN-3	2023	10645	4355	977	70.38	0.674	0.317	0.326	0.683	1.777	0.431

TABLE 3
Results of the P300 Detection for Subject B

Method	TP	TN	FP	FN	Reco.	Recall	Precision	Silence	Noise	Error	F-measure
CNN-1	2035	12039	2961	965	78.19	0.678	0.407	0.232	0.593	1.309	0.509
CNN-2a	1918	11515	3485	1082	74.63	0.639	0.355	0.361	0.645	1.522	0.456
CNN-2b	1996	11535	3465	1004	75.17	0.665	0.365	0.335	0.634	1.490	0.472
CNN-3	2006	11348	3652	994	74.19	0.669	0.354	0.331	0.645	1.549	0.463
MCNN-1	2202	11453	3547	798	75.86	0.734	0.383	0.266	0.617	1.448	0.503
MCNN-2	2010	11754	3246	990	76.47	0.670	0.382	0.330	0.618	1.412	0.487
MCNN-3	2077	11997	3003	923	78.19	0.692	0.409	0.307	0.591	1.309	0.514

- in L_1, the number of free variables is $N_s(N_{elec}+1)$, e.g., 650, 90, and 65 parameters for CNN-1,2,3;
- in L_2, the number of free variables is $5N_s(13+1)$, e.g., 700, 700, and 70 parameters for CNN-1,2,3;
- in L_3, the number of free variables is $100*(6*5N_s+1)$, e.g., 30,100, 30,100, 3,010 parameters for CNN-1,2,3; and
- in L_4, the number of free variables is $2*(100+1)$, e.g., 202 parameters for CNN-1,2,3;

where $N_{elec}=64$, and $N_s=10$.

Therefore, CNN-1,2,3 contain 31,652, 31,092, and 3,347 free variables, respectively.

The average training time was around 10 min on an Intel Core 2 Duo T7500 CPU for CNN-1. This time depends on the subject and mostly on the initial learning parameter λ, which was set to 0.2. The model was implemented in C++ without any special hardware optimization (multicore or GPU). Convolutional neural networks can be implemented by using GPU. Such implementation can provide a significant speedup for both learning and testing [35], [36].

5 Results

5.1 P300 Detection

The analysis of the basic P300 detection is not the main focus in works dedicated to BCI. In P300-BCI, the main task is the speller and the raw classification of the P300 waves is usually not specified. For the first time, we try to find some measurements related to the P300 detection, which could be considered as indexes correlated to the further character recognition. The classification results obtained for the six classifiers are given in Tables 2 and 3. For each method, the following information is provided: the number of true positive (TP), true negative (TN), false positive (FP), false negative (FN) in the test database. If we consider the P300 detection as a binary classification problem, the recognition rate (Reco.), presented in percent, is defined as $(TP+TN)/NP$, where $NP=TP+TN+FP+FN$, i.e., the total number of patterns. Other classical widely used measures for evaluating the quality of results are presented in Tables 2 and 3:

$$\text{Recall}=\frac{TP}{TP+TN},\quad \text{Precision}=\frac{TP}{TP+FP}, \tag{13}$$

$$\text{Silence}=1-\text{Recall},\quad \text{Noise}=1-\text{Precision}, \tag{14}$$

$$\text{Error}=\frac{FP+FN}{TP+FN},\quad \text{F-measure}=2\frac{\text{Recall.Precision}}{\text{Precision}+\text{Recall}}. \tag{15}$$

The first observation is the large difference between the two subjects. Subject B allows getting better results for the classification. Besides, the methods have about the same ranking in relation to the recognition rate. For subject A, the best recognition and precision are obtained with CNN-1 and MCNN-3, whereas the best recall is achieved with MCNN-1 with a score of 0.69. The results of subject B respect the same dichotomy between the methods. For the single classifiers, the reduction of one channel or the use of only eight electrodes reduces the recognition rate of 0.9 and 2.29 percent, respectively, for subject A. For subject B, the difference is more significant, with a difference of about 4 percent in the recognition rate. The difference between CNN-1 and MCNN-1 is not significant as MCNN-1 is built with several CNN-1. Nevertheless, MCNN-1 offers a slight advantage in the recognition rate.

The main outcome of these first results is the difference between the precision and recall in relation to the methods. One interest is to find a link between one of these measurements and the results obtained in the second step for detecting the characters. One problem to solve is to define if it is the recall or the precision that is the most relevant feature for estimating the character recognition quality in the next classification step.

5.2 Network Analysis

The topology of the classifiers allows extracting information about the location of the best electrodes for each subject. For the layers that are not fully connected, it is possible to extract information from the connection weights. Figs. 5 and 6 represent the weights that define

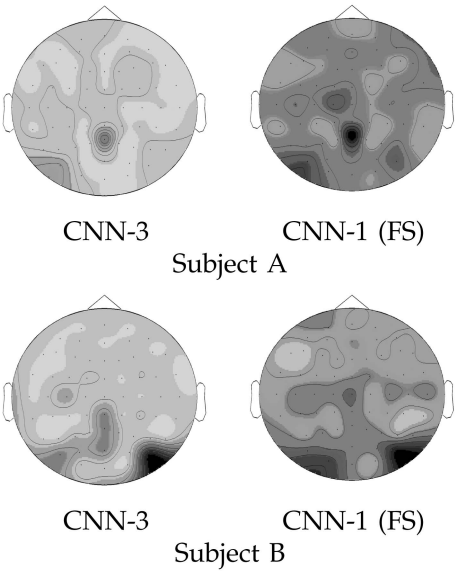


Fig. 4. Discriminant electrodes for subject A and subject B based on the weights of the first hidden later of CNN-3 and CNN-1 (FS).

each map of the first hidden layer of CNN-1. The parts in dark represent weights with a high absolute value. The parts in light correspond to weights that are around 0, i.e., the electrodes that correspond to locations that have a very low discriminant power. Although the analysis of the maps of CNN-1 can be difficult as there exist 10 channels, we observe some similarities between some maps of CNN-1 for the subject A and the map obtained with CNN-3. This is particularly evident in maps 3, 4, and 7, where it is possible to distinguish a precise location in the middle of the head that corresponds to P_Z. For subject B, the information is more widely spread between C_Z and PO_Z. It is interesting to note the difference of location for the same brain activity between two people. The accuracy of the P300 detection between both subjects could be explained by the location of the P300 response. The information is very dense in a particular location for subject A, whereas the dispersion of the information in subject B provides probably reliable results.

The absolute values of the weights in the first hidden layer of CNN-3 and the ξ values of CNN-1 are displayed in Fig. 4. As CNN-3 has only one map that describes the channel combination, the weight set for this map is equivalent to the optimal spatial filter according to the gradient-based learning. Thus, it is possible to extract

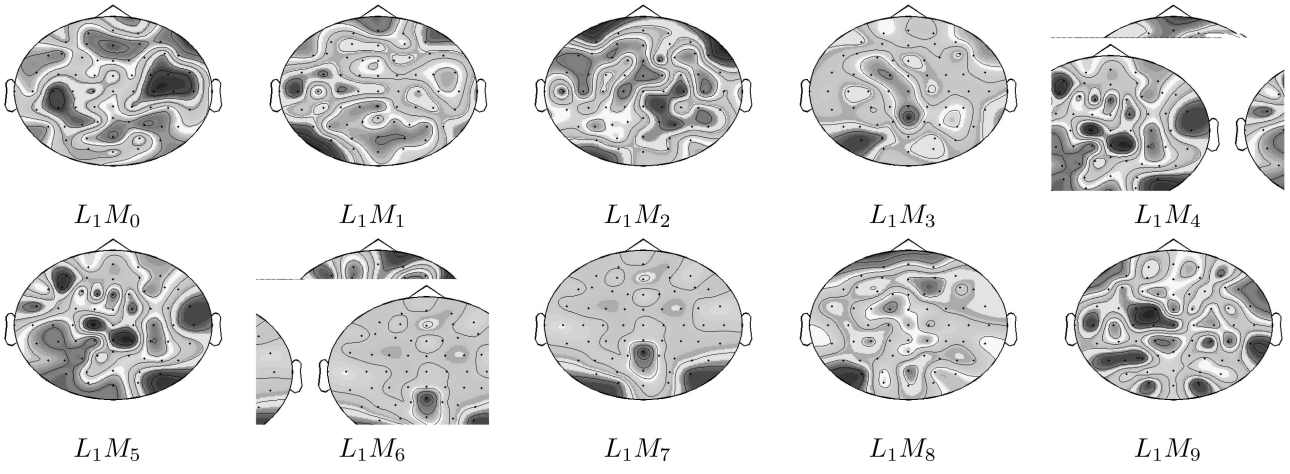


Fig. 5. Spatial filters obtained with CNN-1 for subject A.

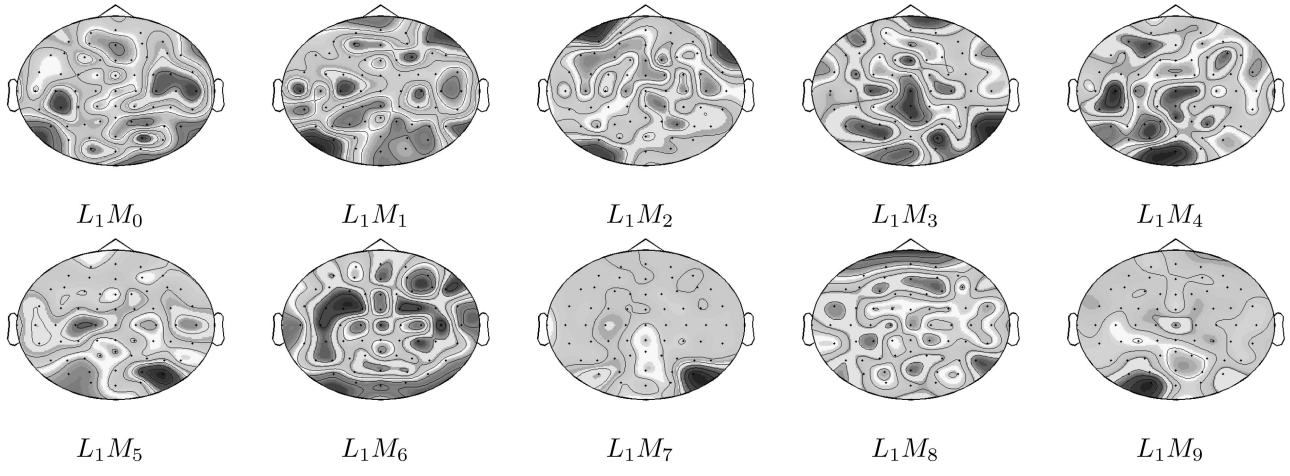


Fig. 6. Spatial filters obtained with CNN-1 for subject B.

TABLE 4
Electrode Ranking

	Best electrodes							
	1	2	3	4	5	6	7	8
A	P_Z	PO_7	C_1	PO_Z	C_5	CP_Z	PO_8	C_Z
B	PO_8	O_1	PO_7	C_Z	PO_3	P_Z	CP_Z	PO_4

directly information from the weights to determine the most discriminant electrodes for the classification. It is possible to observe some common points between ξ and the weights from CNN-3. However, ξ is more heterogeneous. The light gray values of CNN-3 show that it not possible to extract precisely the location where the P300 wave occurs.

Table 4 presents the ranking of the chosen electrodes for creating CNN-2b. The electrodes are sorted from the most to the least discriminant electrode. The set of electrodes is closed to the set that was chosen for CNN-2a. Both subjects share P_Z, C_Z, CP_Z, PO_7, and PO_8. The common electrodes with CNN-2a are C_Z, P_Z, PO_7, and PO_8.

Table 5 represents the recognition rate in percent for each method and each subject for the character recognition problem. The number of epochs corresponds to the number of times a row/column has to flash on one character. The maximum number is 15, as described in the protocol experiment. When the number of epochs is n, it means that only the n first epochs are considered. With one epoch, there are only two P300 possible responses for determining a character: one for the x-coordinate and another for the y-coordinate. The evolution of the accuracy in relation to the number of epochs is not linear. Most of the characters are recognized within 10 epochs. It is noteworthy that adding more epochs does not necessarily improve accuracy. For example, the MCNN-1 method performed better after 12 epochs than 13 and the MCNN-3 approach performed better after 12 epochs than 11. Furthermore, the marginal benefit of additional epochs after the 10th epoch is minimal for subject B, but not subject A. These observations may be noteworthy for P300-BCIs that use the variable averaging approach [37].

We note v, the vector containing the cumulated probabilities of the P300 detection for each of the 12 flashes. The first six values represent the six columns. The last six values represent the rows.

$$v(j) = \sum_{i=1}^{i=n} E(P(i,j)) \quad 1 \leq j \leq 12, \tag{16}$$

where $P(i,j)$ is the pattern at the epoch i corresponding to the subject response for the flash j.

The coordinate of the character are defined by

$$x = \underset{1 \leq i \leq 6}{\operatorname{argmax}}\ v(i), \tag{17}$$

$$y = \underset{7 \leq i \leq 12}{\operatorname{argmax}}\ v(i). \tag{18}$$

The best accuracy is achieved by MCNN-1 with 95.5 percent. In the second position, CNN-1 and MCNN-3 both give the same accuracy. Compared to the P300 detection, the rank of the methods for the character recognition rate respects the order given by the recall. A high recall in the P300 detection involves a high accuracy in the character recognition. This observation should be used for further comparisons where only the recall could describe the quality of the classification and its impact for P300-BCIs.

Fig. 7 displays the information transfer rate (ITR), in bits per minute (bpm), in relation to the number of considered epochs, i.e., over the time needed for the recognition of a character. The ITR is common for measuring communication and control systems; it is used in BCI [38] and was first introduced by Shannon and Weaver [39]. The ITR is defined by

TABLE 5
Character Recognition Rate (in Percent) for the Different Classifiers

		Epochs														
Method	Subject	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CNN-1	A	16	33	47	52	61	65	77	78	85	86	90	91	91	93	97
	B	35	52	59	68	79	81	82	89	92	91	91	90	91	92	92
	Mean	25.5	42.5	53	60	70	73	79.5	83.5	88.5	88.5	90.5	90.5	91	92.5	94.5
CNN-2a	A	14	22	28	44	46	52	62	62	64	64	70	74	77	83	84
	B	28	38	55	68	71	75	79	77	82	85	85	85	88	86	91
	Mean	21	30	41.5	56	58.5	63.5	70.5	69.5	73	74.5	77.5	79.5	82.5	84.5	87.5
CNN-2b	A	12	24	28	38	48	50	59	59	67	67	69	74	75	83	87
	B	32	48	53	66	72	73	75	78	82	82	82	86	87	87	87
	Mean	22	36	40.5	52	60	61.5	67	68.5	74.5	74.5	75.5	80	81	85	87
CNN-3	A	17	20	41	41	44	54	58	66	69	75	79	80	83	84	87
	B	29	38	44	60	70	75	78	76	83	82	86	86	90	90	90
	Mean	23	29	42.5	50.5	57	64.5	68	71	76	78.5	82.5	83	86.5	87	88.5
MCNN-1	A	18	31	50	54	61	68	76	76	79	82	89	92	91	93	97
	B	39	55	62	64	77	79	86	92	91	92	95	95	95	94	94
	Mean	28.5	43	56	59	69	73.5	81	84	85	87	92	93.5	93	93.5	95.5
MCNN-2	A	13	25	33	33	44	53	62	65	67	73	77	82	82	85	86
	B	31	53	59	63	66	75	80	84	88	91	91	94	94	95	95
	Mean	22	39	46	48	55	64	71	74.5	77.5	82	84	88	88	90	90.5
MCNN-3	A	17	35	50	55	63	67	78	79	84	85	91	90	92	94	97
	B	34	56	60	68	74	80	82	89	90	90	91	88	90	91	92
	Mean	25.5	45.5	55	61.5	68.5	73.5	80	84	87	87.5	91	89	91	92.5	94.5

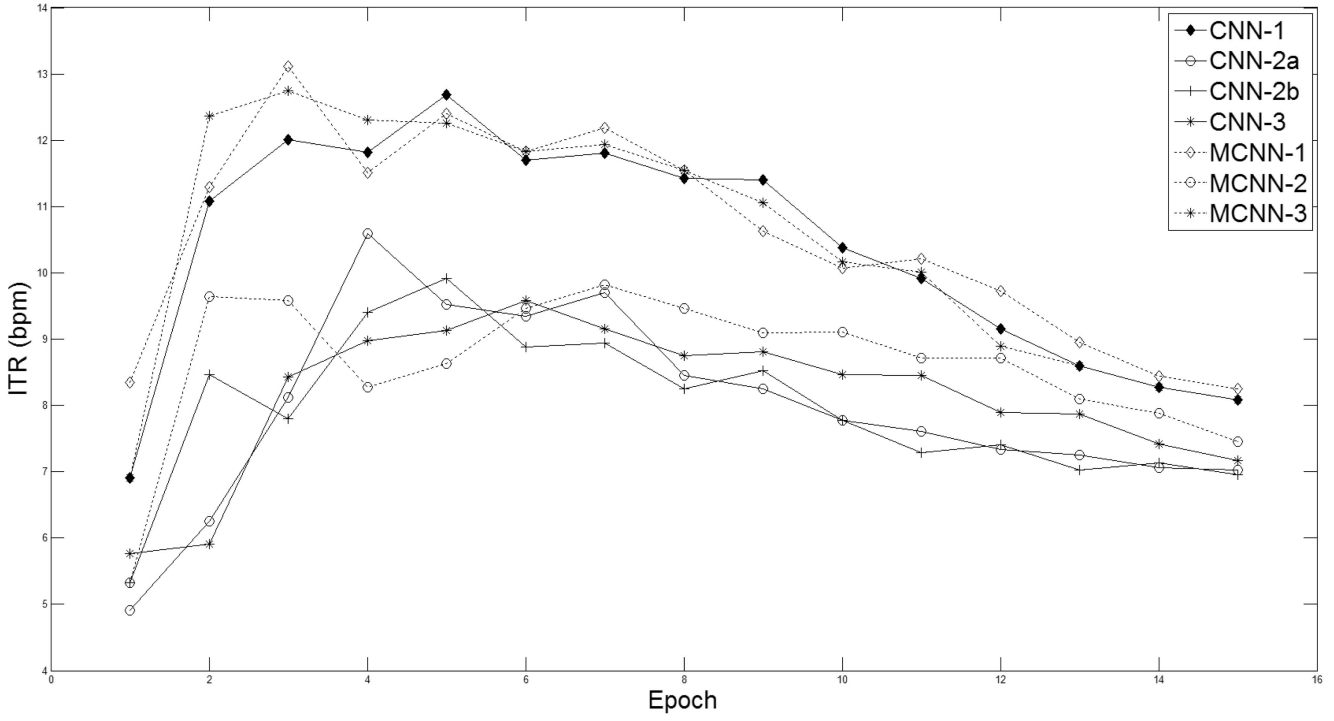


Fig. 7. ITR (in bits per minute) in relation to the number of epochs.

$$ITR = \frac{60\ \left(P\log_2(P) + (1-P)\log_2(\frac{1-P}{N-1}) + \log_2(N)\right)}{T}, \quad (19)$$

where p is the probability to recognize a character, N is the number of classes ($N = 36$), and T is the time needed to recognize one character. T is defined by

$$T = 2.5 + 2.1n \quad 1 \leq n \leq 15, \quad (20)$$

where n is the number of considered epochs. This time is established according to the protocol experiment, where each character epoch starts with a pause of 2.5 s and then each row/column is intensified for 100 ms with a pause of 75 ms ($12 * (100 + 75) = 2{,}100$). Contrary to the recognition rate that increases in relation to the number of epochs, the ITR takes into account the time needed for the recognition. The ITR is maximized with six epochs. However, we can observe that a fast ITR usually implies an average recognition rate. The question of an optimal recognition versus a fast ITR is opened and depends of the application. For instance, reliability is less important in a speller than in some robotic applications or emergencies.

Table 7 presents a comparison of the presented method with other systems. Among the CNN classifiers, only CNN-1 and MCNN-1 are presented. Each cell of the table contains the couple Reco./ITR that represents the character recognition rate in percent and the average ITR in bits per minute. The best recognition rate is achieved with the solution of Rakotomamonjy with the use of 15 epochs [16]. However, the proposed method offers the best recognition rate when only 10 epochs are used. One first observation is the difference between the methods over the number of epochs. One explanation can come from the number of characters to recognize, only 100, which limits the impact of the results. With a difference of 1 percent between two methods, it is impossible to qualify the impact of the classification quality. Nevertheless, we can argue that the CNN method does not consider any electrode selection before the classification contrary to the other methods. All of the electrodes are used without any neuroscience knowledge about the best electrodes or some prior features selection. The classification is done directly on the EEG with few preprocessing. This advantage is relevant for its implementation in a real BCI system, where its all embedded approach can highlight the subject particularities without any tuning.

For a pragmatic BCI, the number of electrodes must be reduced. Table 6 presents a comparison between the best SVM solution and the CNN when both methods consider only eight electrodes as input. The selection of the electrodes given by CNN-2b gives about the same results as the predefined set of CNN-2a.

As the database is available for free online [43], we present each error for both subjects for further comparisons in Table 8. We can note that the errors can be explained, as each error is near the expected character in the speller layout. Most of the time, it is either on the same row or the same column. The creation of a new paradigm that would include flashing diagonals, for instance, could improve the character recognition by cross-validating the P300 responses.

TABLE 6
Comparison of the Recognition Rate (in Percent) with Only Eight Electrodes as Input Feature

Epoch	Method			
	E-SVM [16]	S-SVM [16]	CNN-2a	CNN-2b
5	40.0	31.0	58.5	60.0
15	80.0	70.0	87.5	87.0

TABLE 7
Comparison of the Recognition Rate and the ITR with Other Results in the Literature

Subject	Epoch	Method							
		Hoffmann [40], [41]	Zongtan [40]	Yandong [40]	mLVQ [42]	LDA [42]	ESVM [16], [40]	CNN-1	MCNN-1
A	5	-	-	-	47/6.71	45/6.26	72/13.28	61/10.18	61/10.18
	10	-	-	-	77/8.2	78/8.38	83/9.29	86/9.87	82/9.11
	15	-	-	-	87/6.92	88/7.1	97/8.51	97/8.51	97/8.51
B	5	-	-	-	72/13.28	76/14.51	75/14.2	79/15.47	77/14.83
	10	-	-	-	91/10.91	92/11.13	91/10.91	91/10.91	92/11.13
	15	-	-	-	96/8.33	96/8.33	96/8.33	92/7.69	94/8
Mean	5	53/8.13	59.5/9.78	55/8.63	59.5/9.78	60.5/10.04	73.5/13.74	70/12.69	69/12.4
	10	-	-	-	84/9.48	85/9.68	87/10.07	88.5/10.38	87/10.07
	15	89.5/7.32	90.5/7.46	90.5/7.46	91.5/7.61	92/7.69	96.5/8.42	94.5/8.08	95.5/8.25

6 Discussion

The question arises whether the quality of the classification impairs with a real use in a BCI application. It is important to limit the prospects of this paper for BCI. These results were obtained with only two subjects. In addition, these two subjects of the database are not representative for P300-BCI. These two subjects have an average P300 response compared to other studies like the BCI competition 2003 [44]. In this competition, for the same problem, many participants got a perfect accuracy for the character recognition problem for a low number of epochs: 6 or 5.

Therefore, the data from the third BCI competition are noteworthy primarily because they present an excellent challenge. Researchers can explore different approaches and push the limits of classification approaches. The database has two main interests. First, it forces the system to reach the limit of the P300 detection. It can extend the potential number of persons who can use a P300-BCI. Second, it is an excellent challenge for the machine learning community. Unlike a well-known problem like character recognition, the gap between research and real applications is still important for BCI and many improvements shall be done. The current limits come both from the noninvasive input signal and the algorithms used for the detection of particular brain responses.

While many pattern recognition methods are used in the BCI field, the question of the ground truth creation arises. Indeed, the main interest of BCI is to detect brain activity, which can be related to stimuli or not. In the case of mental imagery, the user has to imagine moving the left/right hand, for example. For the detection of visual evoked potential, like the P300 waves or steady-state visual evoked potentials (SSVEPs), the user has to focus on some visual stimuli (flashing light for P300, flickering light for SSVEP).

TABLE 8
Confusion of Character Recognition

Subject	Character	Output	Expected
A	16	Y	Z
A	30	P	Q
A	52	8	6
B	10	Z	H
B	24	Q	P
B	31	C	A
B	39	V	W
B	55	1	T
B	74	3	4

The ground truth is usually determined on what the subject has to perform. Its creation can therefore be tricky as it is impossible to know what the subject is thinking or where the subject is exactly focusing without the use of an eye tracker. For instance, in the P300 speller, it is possible that the subject may not have always focused on the expected target. In addition, the user can be sensitive to the peripherical lights around the target, where a P300 wave may also occur. This effect is suggested in the errors in the character recognition that are described in the previous section. The possibility of outliers and mislabeled patterns is high. Further works should consider these effects during learning. For instance, the surrounding flashes around a target could not be taken into account during the classifier learning as the probability of mislabeling the corresponding brain response can be high, i.e., a P300 wave can occur when it is not desired.

One of the most important parameters in BCI is the ITR, which depends on the time. The ITR presented in the previous section suggests that the optimal number of epochs should be around six. Some investigations should be carried out in the links between the number of classes in the P300 speller and the number of epochs to get the best ITR.

The interest of convolutional neural networks is double. First, it allows a high performance in the classification. The CNN approach can be qualified as almost naive as the preprocessing steps are limited. It just classifies a signal without directly considering the usual shape of the expected signal to detect, i.e., the deflection after 300 ms of the P300 wave is not used. Second, they can allow deeper analysis of brain activity. During the learning step, particular features can be discovered. Whereas most of the other techniques separate the different parts of the classification (features selection, spatial filters, ...), a CNN can extract all of the needed information during its learning. The weight semantic in the network can carry out other relevant information that may still be unknown to neuroscience.

Whereas some of the techniques use specific preprocessing tools for removing artifacts such as eye blinking and other muscle movements, the CNN solution got excellent results while being invariant to such noise. The CNN can still be improved. The differences between CNN-1 and CNN-3 advocate the critical choice of the topology. The spatial filters in the first hidden layer were created in one step and they don't include any contextual information about the electrode placement. The first hidden layer could be decomposed into several other layers that describe a hierarchical view of the electrodes from Fig. 2. Instead of

processing all of the electrodes together in one layer, several layers could successively reduce the number of channels from 64 to 1. Indeed, models such as LeNet-5 [32] and LeNet-6 [20] have a deep architecture for learning progressively higher level features. One challenge is to determine the key layers and the best topology.

7 Conclusion

A new approach for P300-BCI classification has been presented. This model is based on a convolutional neural network. Its accuracy is equivalent to the best current method on the Data set II of the third BCI competition [16]. It outperforms the best method in two situations: first, when the number of electrodes is restricted to 8; second, when the number of considered epochs is 10. In addition, the classifier does not consider a prior set of selected features or high-level features as input, contrary to the other solution, and it provides some tools throughout the learned weights for interpreting the brain signals. As expected, the combination of different classifiers is the best strategy for obtaining the best results. The recall of the P300 detection is the main feature that dictates the overall performance of the P300 speller. The detection of P300 waves remains a very challenging problem for both the machine learning and neuroscience communities. It possesses a large variability over subjects. As its presence is unsure, it presents high potential of outliers for the classification. Further works will deal with the links between the P300 detection and its impact for the character recognition problem in relation to the number of epochs.

Acknowledgments

The authors would like to thank Dr. Brendan Allison and the reviewers for their comments. This research was supported by a Marie Curie European Transfer of Knowledge grant BrainRobot, MTKD-CT-2004-014211, within the Sixth European Community Framework Program.

References

[1] B.Z. Allison, E.W. Wolpaw, and J.R. Wolpaw, "Brain-Computer Interface Systems: Progress and Prospects," *Expert Rev. Medical Devices,* vol. 4, no. 4, pp. 463-474, 2007.

[2] N. Birbaumer and L.G. Cohen, "Brain-Computer Interfaces: Communication and Restoration of Movement in Paralysis," *J. Physiology—London,* vol. 579, no. 3, pp. 621-636, 2007.

[3] A. Kostov and M. Polak, "Parallel Man-Machine Training in Development of EEG-Based Cursor Control," *IEEE Trans. Rehabilitation Eng.,* vol. 8, no. 2, pp. 203-205, June 2000.

[4] N. Birbaumer, N. Ghanayim, T. Hinterberger, I. Iversen, B. Kotchoubey, A. Kübler, J. Perelmouter, E. Taub, and H. Flor, "A Spelling Device for the Paralysed," *Nature,* vol. 398, pp. 297-298, 1999.

[5] B. Blankertz, G. Dornhege, S. Lemm, M. Krauledat, G. Curio, and K.-R. Müller, "The Berlin Brain-Computer Interface: EEG-Based Communication without Subject Training," *IEEE Trans. Neural Systems and Rehabilitation Eng.,* vol. 14, no. 2, pp. 147-152, June 2006.

[6] F. Lotte, M. Congedo, A. Lecuyer, F. Lamarche, and B. Arnaldi, "A Review of Classification Algorithms for EEG-Based Brain-Computer Interfaces," *J. Neural Eng.,* vol. 4, pp. R1-R13, 2007.

[7] K.-R. Müller, M. Krauledat, G. Dornhege, G. Curio, and B. Blankertz, "Machine Learning Techniques for Brain-Computer Interfaces," *Biomedical Technology,* vol. 49, no. 1, pp. 11-22, 2004.

[8] K.-R. Müller, M. Tangermann, G. Dornhege, M. Krauledat, G. Curio, and B. Blankertz, "Machine Learning for Real-Time Single-Trial EEG-Analysis: From Brain-Computer Interfacing to Mental State Monitoring," *J. Neuroscience Methods,* vol. 167, no. 1 pp. 82-90, 2008.

[9] C.W. Anderson, S.V. Devulapalli, and E.A. Stolz, "Determining Mental State from EEG Signals Using Parallel Implementations of Neural Networks," *Proc. IEEE Workshop Neural Networks for Signal in Processing,* pp. 475-483, 1995.

[10] H. Cecotti and A. Gräser, "Time Delay Neural Network with Fourier Transform for Multiple Channel Detection of Steady-State Visual Evoked Potential for Brain-Computer Interfaces," *Proc. European Signal Processing Conf.,* 2008.

[11] T. Felzer and B. Freisieben, "Analyzing EEG Signals Using the Probability Estimating Guarded Neural Classifier," *IEEE Trans. Neural Systems and Rehabilitation Eng.,* vol. 11, no. 4, pp. 361-371, Dec. 2003.

[12] E. Haselsteiner and G. Pfurtscheller, "Using Time Dependent Neural Networks for EEG Classification," *IEEE Trans. Rehabilitation Eng.,* vol. 8, no. 4, pp. 457-463, Dec. 2000.

[13] N. Masic and G. Pfurtscheller, "Neural Network Based Classification of Single-Trial EEG Data," *Artificial Intelligence in Medicine,* vol. 5, no. 6, pp. 503-513, 1993.

[14] N. Masic, G. Pfurtscheller, and D. Flotzinger, "Neural Network-Based Predictions of Hand Movements Using Simulated and Real EEG Data," *Neurocomputing,* vol. 7, no. 3, pp. 259-274, 1995.

[15] B. Blankertz, G. Curio, and K.-R. Müller, "Classifying Single Trial EEG: Towards Brain Computer Interfacing," *Advances in Neural Information Processing Systems,* T.G. Diettrich, S. Becker, and Z. Ghahramani, eds., vol. 14, pp. 157-164, MIT Press, 2002.

[16] A. Rakotomamonjy and V. Guigue, "BCI Competition III: Data Set II—Ensemble of SVMs for BCI p300 Speller," *IEEE Trans. Biomedical Eng.,* vol. 55, no. 3, pp. 1147-1154, Mar. 2008.

[17] B. Obermaier, C. Guger, C. Neuper, and G. Pfurtscheller, "Hidden Markov Models for Online Classification of Single Trial EEG data," *Pattern Recognition Letters,* vol. 22, no. 12, pp. 1299-1309, 2001.

[18] S. Zhong and J. Gosh, "HMMs and Coupled HMMs for Multi-Channel EEG Classification," *Proc. IEEE Int'l Joint Conf. Neural Networks,* vol. 2, pp. 1154-1159, 2002.

[19] A. Hiraiwa, K. Shimohara, and Y. Tokunaga, "EEG Topography Recognition by Neural Networks," *IEEE Eng. in Medicine and Biology Magazine,* vol. 9, no. 3, pp. 39-42, Sept. 1990.

[20] Y. Bengio and Y. LeCun, "Scaling Learning Algorithms towards AI," *Large-Scale Kernel Machines,* L. Bottou, O. Chapelle, D. DeCoste, and J. Weston, eds., MIT Press, 2007.

[21] P.Y. Simard, D. Steinkraus, and J.C. Platt, "Best Practices for Convolutional Neural Networks Applied to Visual Document Analysis," *Proc. Seventh Int'l Conf. Document Analysis and Recognition,* pp. 958-962, 2003.

[22] S. Sukittanon, A.C. Surendran, J.C. Platt, and C.J.C. Burges, "Convolutional Networks for Speech Detection," *Proc. Eighth Int'l Conf. Spoken Language Processing,* pp. 1077-1080, 2004.

[23] D.J. Krusienski, E.W. Sellers, D. McFarland, T.M. Vaughan, and J.R. Wolpaw, "Toward Enhanced P300 Speller Performance," *J. Neuroscience Methods,* vol. 167, pp. 15-21, 2008.

[24] E. Donchin, K.M. Spencer, and R. Wijesinghe, "Assessing the Speed of a P300-Based Brain-Computer Interface," *IEEE Trans. Neural Systems and Rehabilitation Eng.,* vol. 8, no. 2, pp. 174-179, June 2000.

[25] L. Farwell and E. Donchin, "Talking Off the Top of Your Head: Toward a Mental Prosthesis Utilizing Event-Related Brain Potentials," *Electroencephalography and Clinical Neurophysiology,* vol. 70, pp. 510-523, 1988.

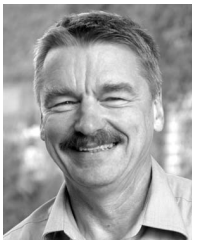
[26] B. Blankertz, K.-R. Müller, D.J. Krusienski, G. Schalk, J.R. Wolpaw, A. Schlögl, G. Pfurtscheller, J.R. Millán, M. Schröder, and N. Birbaumer, "The BCI Competition. III: Validating Alternative Approaches to Actual BCI Problems," *IEEE Trans. Neural Systems and Rehabilitation Eng.,* vol. 14, no. 2, pp. 153-159, June 2006.

[27] G. Schalk, D.J. McFarland, T. Hinterberger, N. Birbaumer, and J. Wolpaw, "BCI2000: A General-Purpose Brain-Computer Interface (BCI) System," *IEEE Trans. Biomedical Eng.,* vol. 51, no. 6, pp. 1034-1043, June 2004.

[28] G.-E. Sharbrough, R.P. Chatrian, H. Lesser, M. Luders, T.W. Nuwer, and T.W. Picton, "American Electroencephalographic Society Guidelines for Standard Electrode Position Nomenclature," *J. Clinical Neurophysiology,* vol. 8, pp. 200-202, 1991.

[29] L. Breiman, "Bagging Predictors," *Machine Learning,* vol. 26, no. 2 pp. 123-140, 1996.
[30] U. Hoffmann, G. Garcia, J.-M. Vesin, K. Diserens, and T. Ebrahimi, "Boosting Approach to p300 Detection with Application to Brain-Computer Interfaces," *Proc. IEEE Conf. Neural Eng.,* pp. 97-100, 2005.
[31] Y. LeCun, F.-J. Huang, and L. Bottou, "Learning Methods for Generic Object Recognition with Invariance to Pose and Lighting," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* 2004.
[32] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," *Proc. IEEE,* vol. 86, no. 11, pp. 2278-2324, Nov. 1998.
[33] Y. LeCun, L. Bottou, G. Orr, and K.-R. Müller, "Efficient Backprop," *Neural Networks: Tricks of the Trade,* G. Orr and K. Muller, eds., Springer, 1998.
[34] J. Kittler, M. Hatef, R.P.W. Duin, and J. Matas, "On Combining Classifiers," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 20, no. 3, pp. 226-239, Mar. 1998.
[35] K. Chellapilla, S. Puri, and P.Y. Simard, "High Performance Convolutional Neural Networks for Document Processing," *Proc. 10th Int'l Workshop Frontiers in Handwriting Recognition,* 2006.
[36] R.J. Meuth and D.C. Wunsch, "Approximate Dynamic Programming and Neural Networks on Game Hardware," *Proc. Int'l Joint Conf. Neural Networks,* 2007.
[37] H. Serby, E. Yom-Toy, and G.F. Inbar, "An Improved P300-Based Brain-Computer Interface," *IEEE Trans. Neural Systems and Rehabilitation Eng.,* vol. 13, no. 1, pp. 89-98, Mar. 2005.
[38] J.R. Wolpaw, N. Birbaumer, D.J. McFarland, G. Pfurtscheller, and T.M. Vaughan, "Brain-Computer Interfaces for Communication and Control," *Clinical Neurophysiology,* vol. 113, pp. 767-791, 2002.
[39] C.E. Shannon and W. Weaver, *The Mathematical Theory of Communication.* Univ. of Illinois Press, 1964.
[40] B. Blankertz, "BCI Competition III—Final Results," http://ida.first.fraunhofer.de/projects/bci/competition_iii/results/, 2008.
[41] U. Hoffmann, G. Garcia, J.M. Vesin, and T. Ebrahimi, "Application of the Evidence Framework to Brain-Computer Interfaces," *Proc. Conf. IEEE Eng. Medicine and Biology Soc.,* vol. 1, pp. 446-449, 2004.
[42] N. Liang and L. Bougrain, "Averaging Techniques for Single-Trial Analysis of Oddball Event-Related Potentials," *Proc. Fourth Int'l BCI Workshop and Training Course,* pp. 44-49, 2008.
[43] B. Blankertz, "BCI Competition III Webpage," http://ida.first.fraunhofer.de/projects/bci/competition_iii, 2008.
[44] B. Blankertz, K.-R. Müller, G. Curio, T.M. Vaughan, G. Schalk, J.R. Wolpaw, A. Schlögl, C. Neuper, G. Pfurtscheller, T. Hinterberger, M. Schröder, and N. Birbaumer, "The BCI Competition 2003: Progress and Perspectives in Detection and Discrimination of EEG Single Trials," *IEEE Trans. Biomedical Eng.,* vol. 51, no. 6, pp. 1044-1051, June 2004.

Hubert Cecotti received the MSc and PhD degrees in computer science from the Universities of Nancy, France, in 2002 and 2005, respectively. Since 2008, he has been a research scientist in the Institute of Automation, Bremen University, Germany, where he has worked on EEG signal processing and brain-computer interfacing on the European project Brainrobot. In 2006 and 2007, he was a lecturer in computer science at the University Henri Poincaré and ESIAL, Nancy, France. His research interests include neural networks, multiclassifiers systems, character recognition, and brain-computer interfaces.



Axel Gräser received the diploma degree in electrical engineering from the University of Karlsruhe, Germany, in 1976, and the PhD degree in control theory from the Technical University of Darmstadt, Germany, in 1982. Since 1994, he has been the head of the Institute of Automation, University of Bremen, Germany. From 1982 to 1990, he was the head of the Control and Software Department of Lippke GmbH, Germany. From 1990 to 1994, he was a professor of control systems, process automation, and real-time systems at the University of Applied Sciences, Koblenz, Germany. He is the manager and coordinator of the European Union Project BRAIN. His research interests include service robotics, brain-computer interfaces, visual servoing, digital image processing, and augmented reality.