Convolutional neural networks for event-related potential detection: impact of the architecture

H. Cecotti

***Abstract*— The detection of brain responses at the single-trial level in the electroencephalogram (EEG) such as event-related potentials (ERPs) is a difficult problem that requires different processing steps to extract relevant discriminant features. While most of the signal and classification techniques for the detection of brain responses are based on linear algebra, different pattern recognition techniques such as convolutional neural network (CNN), as a type of deep learning technique, have shown some interests as they are able to process the signal after limited pre-processing. In this study, we propose to investigate the performance of CNNs in relation of their architecture and in relation to how they are evaluated: a single system for each subject, or a system for all the subjects. More particularly, we want to address the change of performance that can be observed between specifying a neural network to a subject, or by considering a neural network for a group of subjects, taking advantage of a larger number of trials from different subjects. The results support the conclusion that a convolutional neural network trained on different subjects can lead to an AUC above 0.9 by using an appropriate architecture using spatial filtering and shift invariant layers.**

I. Introduction

In machine learning, the methods based on deep learning have gained a great success in classification problems, optimization control, and time series analysis. More particularly, deep learning methods for pattern recognition using prior information about the problem for the creation of their architecture, such as the number of main dimensions in the input signal, and the type of variations that can occur across trials, have won a large number of competitions [1]. Convolutional neural networks (i.e., conv nets or CNN) have been initially proposed and evaluated for handwritten character recognition during the 90s, providing state-of-the-art results, and they have been popular in the research community of document analysis and recognition [2]–[4]. Because the architecture of a conv net must be set in relation to a particular problem, with specific connections between units in the network, its implementation must be specific as each hidden layer includes more characteristics than a fully connected hidden layer in a multi-layer perceptron (MLP). Thanks to graphical processing units and recent open-source libraries that allow to focus only on the choice of the architecture, and not the implementation of the conv net itself, more architectures and deeper architectures can be tested rapidly on various databases [5].

While conv nets have been successfully used in computer vision, they have not been fully exploited in electroencephalograhy (EEG) signal processing, for applications such as brain-computer interface. First, the low number of available trials to train a model and the low signal-to-noise ratio do not allow to easily capture the manifold where lies the data. Most of the signal processing and classification techniques rely on linear algrebra, e.g. linear classifiers, for the classification of ERPs. Some studies using conv nets have been proposed for the detection of event-related potentials (ERPs) in the EEG signal, e.g. the P300 [6], with applications to the P300 speller and rapid serial visual presentation tasks for target detection [7]. These architectures typically include a first convolutional layer that acts as a spatial filtering layer, then the next layers decrease the number of features through filtering in the time domain and/or subsampling functions. Finally, the last hidden layers may include a fully connected hidden layer. It is worth noting that as linear discriminant analysis provides relevant results for ERP detection, it is not necessary to use a fully connected hidden layer at the last stage of the architecture, i.e., such as a multi-layer perceptron after the extracted features through a succession of convolution and pooling functions. Special architectures have also been proposed for the detection of steady-state visual evoked potentials (SSVEP), where spatial filtering was achieved through a convolutional layer and then the Fourier transform was inserted between two hidden layers to transfer the power spectrum of the signal to higher levels in the network [8]. In all these cases, these architectures used sigmoid functions as activation units as opposed to the most recent rectified linear unit function (ReLU) [9], which provides a reduced likelihood of vanishing gradient.

In this paper, we propose to compare different architectures of conv nets by using state-of-the-art functions that are readily available online with current libraries and toolboxes (e.g. Matlab). This paper focuses therefore on the choice of the architecture, and it does not aim at providing new convolution or pooling functions. The performance is evaluated on a database of healthy participants who had to search a target image during a rapid serial presentation task [10], [11]. We propose to investigate the performance of CNNs in relation of their architecture, and also in relation to the evaluation method: a single system for each subject, or for all the subjects. More particularly, we want to address the change of performance that can be observed between specifying a neural network to a subject, or by considering a neural network for a group of subjects, taking advantage of a larger number of trials from different subjects. The extent to which it is better to have a subject-specific system or not is important in the field of brain-computer interface (BCI) because the user experience must be enhanced for patients

Intelligent System Research Centre, Ulster University, Magee Campus, Derry ~ Londonderry, N. Ireland, UK. h.cecotti@ulster.ac.uk

who would need to use a BCI daily by removing the need of a calibration session. The remainder of this paper is as follows. The inputs and the convolutional layers are described in Section II. The results are presented in Section IV. Finally, the impact of the results are discussed in Section V.

II. Methods

We consider inputs of size $N_t \times N_c$ where N_t and N_c represent the number of time points and the number of channels, respectively. The architecture of a conv net contains several hidden layers, including at least one convolutional layer. In this type of network, the weights of the layer are shared across the different inputs. The weight sharing model reduces the number of parameters to learn in the network, making a conv net faster to train. The convolution on a 2D signal such as an image behaves like the application of a linear filter, follows by an activation function (e.g. sigmoid or ReLU function). The architecture of a conv net depends on the problem and the type of variations across trials. With images, it is easy to get a good feeling about the geometric deformations that can be applied on the signal and keep the same label. For the choice of the architecture with images, it is also possible to get inspired by the visual system in the human brain. However, for the classification of brain responses, it is more difficult to determine what type of features must be extracted. For the detection of ERPs, the experimental protocol has a central role as it will determine what will be the ERP components and their characteristics that can vary between two conditions (i.e., target vs. non-target). For the classification of ERPs for BCI that include the P300, a large ERP component, it is important to determine the type of changes that can occur over time. A first assumption is the stationarity of the spatial location of the brain responses of interest. We assume that there exists a finite subset of spatial distribution where we can find discriminant information for the task. This first stage can be achieved by spatial filtering, where the input signals acquired from different channels is then projected to one or several "virtual" channels. Spatial filtering can be achieved through a convolutional layer of size $[1 \times N_c]$. It is worth noting that the spatial filtering stage can include more layers, e.g. by grouping sensors that are close, taking into account the spatial location of each sensor. In such a case, weight sharing may not be the ideal choice as each neighborhood may rely on a specific function. The next processing step can deal with the reduction of the number of time points. This stage is necessary if the signal has a high sampling rate and the signal has not beeen downsampled. In recent BCI competitions related to ERPs detection, the signal was often bandpassed to frequencies that are limited to the delta, theta, and alpha bands [0.1-12 Hz], suggesting the number of time points could be limited if the main ERP component is the P300. Other signal processing steps that can be included through the architecture of the conv net include the extraction of shift invariant features.

III. Experimental protocol

The EEG database was previously used in [12], [13]. The experimental protocol is described thereafter. Participants were seated 75 cm from a Dell P2210 monitor. They viewed a series of simulated images from a desert metropolitan environment using a rapid serial visual presentation (RSVP) paradigm. Images (960 $\times$ 600 pixels, 96 dpi, subtending 36.3 $\times$ 22.5) were presented using E-prime software on a Dell Precision T7400 PC. Images were presented with a stimulus onset asynchrony of 0.5 s, with no inter-stimulus interval. Images contained either a scene without any people (non-target) or a scene with a person holding a gun (target). 110 target images and 1346 non-target images were presented to each participant. Scenes in which a target appeared were also presented without the person in the non-target condition. All stimuli appeared within 6.5 degrees of center of the monitor. The purpose of the task was to discriminate target images from non-target images. In the considered data, 16 participants responded to targets by silently counting the number of targets. Electrophysiological recordings were digitally sampled at 1024 Hz from 64 scalp electrodes arranged in a 10-10 montage using a BioSemi Active Two system (Amsterdam, Netherlands). Impedances were kept below 25 $k\Omega$. External leads were placed on the outer canthus of both eyes, and above and below the right orbital fossa to record the electrooculogram signal. The signal was then bandpassed between 0.1 and 21.33 Hz using a 4^{th} order Butterworth filter. Finally, the signal was downsampled to 64 Hz. For each stimulus, we selected the signal corresponding to 800 ms after the stimulus onset, i.e., 51 time points. Hence, each example in the database is a matrix of size 51×64, which is about the same size of the images in computer vision problems.

A. Conv nets

We propose 6 neural networks architectures in order to highlight the effect of some hidden layers. In all the models, the activation unit is a ReLU function ($f(x) = max(0, x)$), the number of neurons in the output layer is set to 2, i.e., the number of classes, and the outputs are normalized with a softmax function. The first architecture CNN_1 has no hidden layers: the inputs are directly connected to the output layer. In CNN_2, there is a single hidden layer with 10 neurons. With CNN_3, we consider a first convolutional layer with 8 maps, the convolution window is set to $[1 \times 64]$, it corresponds to a spatial filtering function. In the second convolutional layer, the layer has only a single dimension (in time). We use 16 maps with a convolution window set to $[48 \times 1]$, hence this layer is almost a classification layer as it considers most of the inputs. Thanks to this convolution, we obtain a set of outputs that corresponds to different shifts of a part of the input features. Finally, this layer is fully connected to the output layer. For CNN_4, CNN_5, and CNN_6, we use only a single convolutional layer with spatial filtering purpose ($[1 \times 64]$) with 4, 8, and 16 maps, respectively. The selection of the best model is based on the maximization of the area under the ROC curve (AUC) by using a validation dataset.

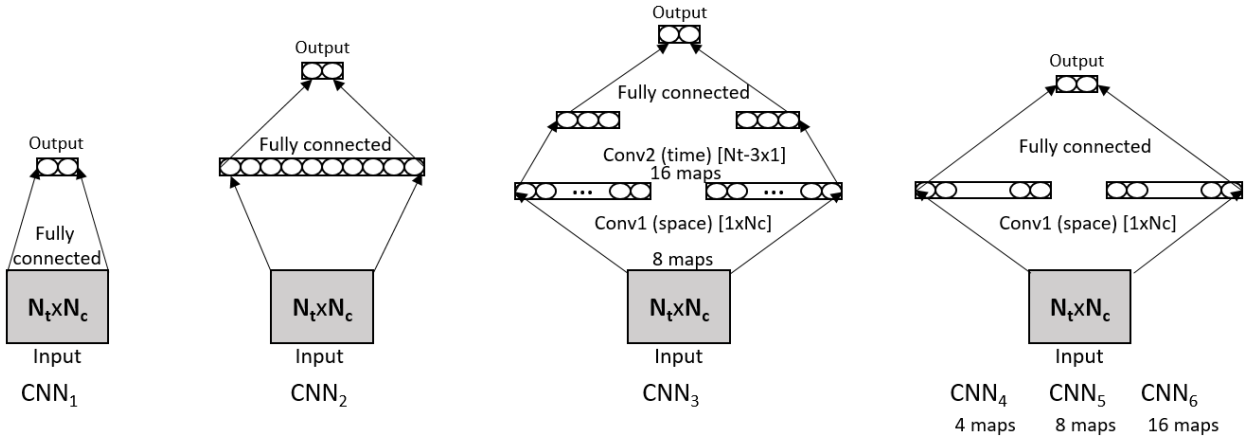

Fig. 1. Description of the different architectures.

Because the number of target trials is significantly inferior to the number of non-target trials, the target trials have been replicated in the training dataset for each epoch.

B. Performance evaluation

For the performance evaluation, we consider two conditions. In the first condition, a model is trained for each participant. We consider a 5-fold cross validation procedure, where 1 fold is used for the test, 1 fold is used for the validation, and the 3 remaining folds are use for training the model. In the second condition, a model is trained for all the participants. We consider a 16-fold cross validation procedure where 1 fold, i.e. 1 subject, is dedicated to the test, 1 fold is used for the validation, and the other blocks are used to train the model. With the last condition, the trials from different subjects are not mixed, hence the classifier is never trained on data from a subject that is used for the test or the validation. The system was implemented with Matlab2016b and its deep learning toolbox using an Intel i7-6700K, 32Gb, and an NVidia GTX1080 graphic card.

IV. Results

The results for the models that were trained for each subject are presented in Table I. The mean and standard deviation across the 5 folds are given for each subject and architecture. The worst results are obtained with CNN_1 that is the simplest architecture, as there is no hidden layer. While the results are inferior to other architectures, the AUC is 0.831, which is significantly above chance level, confirming that a simple method, with no prior information about the problem can lead to an efficient solution. Between CNN_4, CNN_5, CNN_6, the variation of the number of spatial filters does not have a fundamental impact on the overall performance, with an AUC of 0.897 ± 0.041, 0.894 ± 0.039, and 0.891 ± 0.043. With pairwise comparisons using a Wilcoxon signed rank test, the performance obtained with CNN_1 is significantly lower than with the other methods. However, there is no statistically significant difference between the other methods, showing that a deep architecture does not have a significant effect.

The results for the models that are trained with several subjects are presented in Table II. The best performance is obtained with CNN_3 with an average AUC of 0.905. With a Wilcoxon signed rank test, pairwise comparisons reveal that CNN_3 provides better performance than CNN_1, CNN_2, and CNN_4. As there is no difference between CNN_3 and CNN_5, it indicates that the second convolutional layers has no key impact on the performance. Overall, based on the results from both conditions, the results indicate the need of a special architecture for the features extraction in the ERPs.

V. Discussion and conclusion

Methods based on linear algebra such as xDAWN [14] for spatial filtering and LDA and its variant for classification have offered in different applications state-of-the-art performance. With the recent availability of tools that allow biomedical engineers to focus only on the architecture, conv nets provide now a reliable alternative to linear algebra based techniques. In the present paper, we have focus on the evaluation of artificial neural networks, more particularly convolutional neural networks. We have shown that it is possible to achieve a better performance by training a model with a large number of subjects, instead of learning a model for each individual, by taking advantage of a high number of training samples. A key problem in BCI and in many human-computer interaction applications is the need of a calibration session to model the characteristics of the user. A calibration session before each use of the system can significantly decrease the users comfort. Thanks to a model that is able to capture a large variability across trials, it is then possible to model a multi-subject classifier. We have shown that while the type of system requires a long time for the estimation of the model, i.e., to train the classifier, the performance can be robust enough for the creation of a generic classifier. Different preprocessing steps can be included within the architecture of a conv net, such a spatial filtering, and the extraction of shift invariant features. Surprisingly, the results on the chosen ERP database suggest that a deep architecture is not necessary and that the convolutional layers, i.e. the decomposition of the different dimensions of the input signal,

TABLE I

SINGLE-TRIAL PERFORMANCE (AUC) FOR DIFFERENT CNN ARCHITECTURES (EACH CNN MODEL IS TRAINED FOR EACH SUBJECT).

Subject	CNN_1	CNN_2	CNN_3	CNN_4	CNN_5	CNN_6
1	0.888 ± 0.027	0.854 ± 0.026	0.846 ± 0.035	0.859 ± 0.025	0.849 ± 0.026	0.844 ± 0.030
2	0.771 ± 0.089	0.817 ± 0.046	0.831 ± 0.090	0.807 ± 0.056	0.830 ± 0.050	0.777 ± 0.063
3	0.949 ± 0.026	0.967 ± 0.017	0.969 ± 0.014	0.966 ± 0.010	0.965 ± 0.015	0.963 ± 0.009
4	0.862 ± 0.076	0.878 ± 0.057	0.885 ± 0.029	0.896 ± 0.046	0.872 ± 0.023	0.879 ± 0.034
5	0.854 ± 0.063	0.873 ± 0.070	0.871 ± 0.045	0.879 ± 0.051	0.873 ± 0.052	0.877 ± 0.063
6	0.831 ± 0.072	0.835 ± 0.057	0.881 ± 0.044	0.874 ± 0.052	0.866 ± 0.040	0.862 ± 0.052
7	0.903 ± 0.023	0.929 ± 0.021	0.929 ± 0.031	0.942 ± 0.023	0.928 ± 0.032	0.932 ± 0.024
8	0.454 ± 0.149	0.706 ± 0.083	0.832 ± 0.040	0.846 ± 0.031	0.826 ± 0.036	0.819 ± 0.045
9	0.911 ± 0.038	0.965 ± 0.023	0.944 ± 0.046	0.946 ± 0.037	0.945 ± 0.041	0.950 ± 0.037
10	0.600 ± 0.141	0.719 ± 0.064	0.801 ± 0.078	0.816 ± 0.050	0.832 ± 0.056	0.805 ± 0.062
11	0.812 ± 0.083	0.896 ± 0.051	0.892 ± 0.073	0.892 ± 0.074	0.884 ± 0.062	0.884 ± 0.078
12	0.951 ± 0.022	0.953 ± 0.025	0.952 ± 0.037	0.955 ± 0.028	0.956 ± 0.020	0.956 ± 0.027
13	0.930 ± 0.073	0.974 ± 0.021	0.969 ± 0.024	0.977 ± 0.010	0.970 ± 0.025	0.970 ± 0.027
14	0.941 ± 0.028	0.938 ± 0.058	0.953 ± 0.039	0.942 ± 0.031	0.963 ± 0.029	0.964 ± 0.030
15	0.775 ± 0.067	0.841 ± 0.062	0.894 ± 0.052	0.876 ± 0.071	0.857 ± 0.095	0.882 ± 0.066
16	0.860 ± 0.078	0.880 ± 0.050	0.873 ± 0.027	0.873 ± 0.055	0.893 ± 0.024	0.888 ± 0.040
Mean	0.831 ± 0.066	0.876 ± 0.046	0.895 ± 0.044	0.897 ± 0.041	0.894 ± 0.039	0.891 ± 0.043
SD	0.134 ± 0.039	0.081 ± 0.021	0.053 ± 0.021	0.053 ± 0.019	0.052 ± 0.020	0.061 ± 0.019

TABLE II

SINGLE-TRIAL PERFORMANCE (AUC) FOR DIFFERENT CNN ARCHITECTURES (EACH CNN MODEL IS TRAINED WITH 14 SUBJECTS).

Subject	CNN_1	CNN_2	CNN_3	CNN_4	CNN_5	CNN_6
1	0.897	0.886	0.915	0.909	0.917	0.914
2	0.801	0.797	0.855	0.796	0.797	0.815
3	0.959	0.961	0.972	0.972	0.969	0.960
4	0.907	0.910	0.901	0.901	0.921	0.921
5	0.907	0.879	0.935	0.894	0.916	0.904
6	0.866	0.860	0.857	0.844	0.838	0.843
7	0.903	0.915	0.948	0.921	0.945	0.949
8	0.780	0.705	0.787	0.791	0.742	0.777
9	0.925	0.955	0.956	0.961	0.948	0.962
10	0.694	0.721	0.803	0.755	0.780	0.720
11	0.909	0.900	0.920	0.894	0.904	0.907
12	0.958	0.957	0.975	0.972	0.974	0.983
13	0.944	0.923	0.950	0.927	0.932	0.946
14	0.928	0.915	0.930	0.933	0.937	0.922
15	0.848	0.860	0.845	0.857	0.870	0.859
16	0.920	0.936	0.924	0.918	0.918	0.937
Mean	0.884	0.880	0.905	0.890	0.894	0.895
SD	0.072	0.078	0.058	0.065	0.070	0.073

are not necessary either.

Further work will be dedicated to the optimization of the architecture and the addition of artificial trials to extend the training database. Finally, while very deep architectures could provide better results than the results presented in this paper, the choice of the architecture should be ideally justified and its performance could help to better understand the variations that occur across trials during an experimental task.

Acknowledgment

H.C. is supported by the Northern Ireland Functional Brain Mapping Facility project (1303/101154803).

REFERENCES

[1] D. Cireşan, U. Meier, L. M. Gambardella, and J. Schmidhubern, "Convolutional neural network committees for handwritten character classification," in *Proc. of the 11th Int. Conf. on Document Analysis and Recognition (ICDAR)*, 2011, pp. 1135–1139.

[2] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel, "Handwritten digit recognition with a back-propagation network," in *Advances in Neural Information Processing Systems (NIPS) 2*, 1990, pp. 396–404.

[3] 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.

[4] P. Simard, D. Steinkraus, and J. Platt, "Best practices for convolutional neural networks applied to visual document analysis," in *Proc. of the 7th Int. Conf. Document Analysis and Recognition (ICDAR)*, Aug. 2003, pp. 958–962.

[5] R. Manor and A. B. Geva, "Convolutional neural network for multi-category rapid serial visual presentation BCI," *Front Comput Neurosci.*, vol. 9, no. 146, 2015.

[6] H. Cecotti and A. Gräser, "Convolutional neural networks for P300 detection with application to brain-computer interfaces," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 33, no. 3, pp. 433–445, Mar. 2011.

[7] H. Cecotti, M. P. Eckstein, and B. Giesbrecht, "Single-trial classification of event-related potentials in rapid serial visual presentation tasks using supervised spatial filtering," *IEEE Trans. Neural Networks and Learning Systems*, vol. 15, pp. 2030–42, Nov. 2014.

[8] H. Cecotti, "A time-frequency convolutional neural network for the offline classification of steady-state visual evoked potential responses," *Pattern Recognition Letters*, vol. 32, no. 8, pp. 1145–1153, 2011.

[9] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun, "What is the best multi-stage architecture for object recognition?" in *Proc. of the 12th Int. Conf. on Computer Vision (ICCV'09)*, 2009, pp. 2146–2153.

[10] A. Gerson, L. Parra, and P. Sajda, "Cortically-coupled computer vision for rapid image search," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 14, no. 2, pp. 174–179, 2006.

[11] E. A. Pohlmeyer, J. Wang, D. C. Jangraw, B. Lou, S. Chang, and P. Sajda, "Closing the loop in cortically-coupled computer vision: a brain-computer interface for searching image databases," *J. Neural Eng.*, vol. 8, p. 036025, 2011.

[12] A. R. Marathe, A. J. Ries, V. J. Lawhern, B. J. Lance, J. Touryan, K. McDowell, and H. Cecotti, "The effect of target and non-target similarity on neural classification performance: a boost from confidence," *Frontiers in Neuroscience*, vol. 9, pp. 1–11, 2015.

[13] H. Cecotti, A. Marathe, and A. J. Ries, "Optimization of single-trial detection of event-related potentials through artificial trials," *IEEE trans. Biomed. Eng.*, pp. 1–7, 2015.

[14] B. Rivet and A. Souloumiac, "Optimal linear spatial filters for event-related potentials based on a spatio-temporal model: Asymptotical performance analysis," *Signal Processing*, vol. 93, no. 2, pp. 387–398, 2013.