2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)
Banff Center, Banff, Canada, October 5-8, 2017

Deep Learning-based Classification for Brain-Computer Interfaces

John Thomas*, Tomasz Maszczyk*, Nishant Sinha†‡, Tilmann Kluge§ and Justin Dauwels*
*School of Electrical and Electronics Engineering
Nanyang Technological University, Singapore
Email: jdauwels@ntu.edu.sg, john007@e.ntu.edu.sg
†Institute of Neuroscience, Faculty of Medical Sciences
Newcastle University, Newcastle upon Tyne, UK
‡School of Computing Science
Newcastle University, Newcastle upon Tyne, UK
§Center for Health & Bioresources
AIT Austrian Institute of Technology, Vienna, Austria

***Abstract*—Brain-computer interface (BCI) is an emerging area of research that aims to improve the quality of human-computer applications. It has enormous scope in biomedical applications, neural rehabilitation, biometric authentication, educational programmes, and entertainment applications. A BCI system has four major components: signal acquisition, signal preprocessing, feature extraction, and classification. In this study, we provide a comparison of various traditional classification algorithms to the newer methods of deep learning. We explore two different types of deep learning methods, namely, convolutional neural networks (CNN) and recurrent neural networks (RNN) with long short-term memory (LSTM) architecture. We test the classification accuracies on a recent 5-class steady-state visual evoked potential (SSVEP) dataset. The results prove the superiority of deep learning methods in comparison with the traditional classification algorithms. Amongst the traditional classifiers, support vector machine (SVM) with Gaussian kernel employing sequential forward selection (SFS) of features provided a better classification accuracy of 66.09%, while CNN provided the highest classification accuracy of 69.03%.**

I. Introduction

Brain-computer interface (BCI) is an emerging area of research with enormous scope in medical as well as non-medical applications. BCI is a communication channel to directly control external devices or applications with brain signals. It enables the users to interact with the environment independent of peripheral nerves and muscles, using large-scale neural signaling mechanisms in the brain. BCI finds its applications in various fields, such as medicine, neuroergonomics, smart environment, education and self-regulation, games and entertainment [1], and security and authentication [2]. In recent years, extensive work has been accomplished regarding the application of BCI in biomedical systems: detection and diagnosis of tumors and neurological disorders [3], neurorehabilitation [4], assisting activities of daily living (ADL), and in the design of neuroprosthetics.

A BCI system consists of the following parts: signal acquisition, signal preprocessing, feature extraction, and classification [5]. In signal acquisition, the electrical activity produced by the brain is recorded by performing the specified voluntary task. There are generally two methods of signal acquisition procedures: invasive and non-invasive. The invasive signal acquisition involves surgical interventions and electrodes are placed on the surface of the brain. In non-invasive acquisition, the signal is collected without any surgical interventions. The former provides better signal quality, but the latter is preferred owing to its ease of implementation. Functional magnetic resonance imaging (fMRI), functional near-infrared spectroscopy (fNIRS), magnetoencephalography (MEG), and electroencephalogram (EEG) are some of the widely applied non-invasive signal acquisition methods [5]. EEG is commonly applied for BCI problems due to its non-invasiveness, economic viability, ease of usage, and portability. EEG is a measurement of the electrical signals produced by the brain by placing electrodes on the scalp. The evoked potential (EP) is a widely studied category of BCI electrical signals. EPs are the electrical signals produced by the brain in response to stimuli. EPs can be divided into three categories based on the type of stimuli: visual (VEP), auditory (AEP), and somatosensory (SEP). In this paper, we are analyzing a VEP dataset. When the stimulation frequency is low (less than 4 Hz), the signals are referred to as transient VEPs and when the stimulation frequency is high (greater than 6 Hz), the signals are referred to as steady-state VEPs (SSVEPs) [6]. This is due to the fact that if a series of identical visual stimuli are presented at a high frequency, the brain enters into a steady state and generates neuronal signals at the same (or multiples of) frequency as that of the stimulus, without any transient phase [7]. Here we apply the SSVEP dataset analyzed in the study designed by Oikonomou et al. in [8], available from the Physionet database [9]. Followed by signal acquisition, the data is preprocessed. Numerous features can be extracted from the preprocessed signals including wavelet coefficients, power spectral density, amplitude parameters, etc. The extracted feature vector is then applied to train a classifier. The classifier aims to identify the user's intended action into one of the predefined classes based on the feature values. Various classification algorithms have been implemented in the literature, such as k-nearest neighbors (k-NN) [10], multilayer perceptron (MLP) [11], decision trees [12], linear discriminant analysis (LDA) [13], and SVM [14]. The basic model of a BCI study is illustrated in Fig. 1.

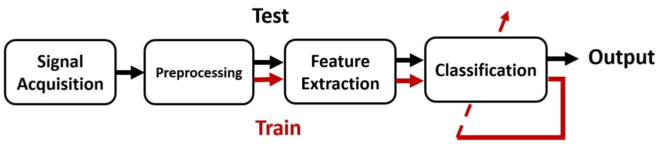

Fig. 1. The basic model of a BCI study. The training part is shown in red.

There are various challenges faced by BCI studies. The design of the experiment, data acquisition, and training requires a high degree of organization and timing. Developing a large database is quite tedious and expensive. The variability of the brain signals produced across subjects is also quite high. A system trained on a particular set of subjects may not work efficiently on a different subject. The quality of the data acquired is another concern. The non-invasive methods are prone to artifacts such as electrocardiograms (ECGs), eye blinks, muscle artifacts, etc. Since the amplitude of brain signal recordings is in the range of microvolts, artifacts pose a major challenge in the classification process. Moreover, the desired features that provide the best classification results are unknown in most of the studies. In general, such highly predictive features are found by a trial and error method, i.e., evaluating all the features and then selecting the best set of features.

In this study, we optimize the classification component of a BCI system by applying deep learning techniques. Deep learning techniques are shown to provide higher classification accuracies in the literature. Moreover, deep networks are able to detect latent structures or features from the raw data. This would help us to reduce the dependency on the feature extraction part of the BCI system. We analyze the performance of traditional classification algorithms with the newer, sophisticated deep learning methods of convolutional neural networks (CNN) [15] and recurrent neural networks (RNN) [16]. In RNN, we implement the long short-term memory (LSTM) architecture [17].

CNN has been applied for SSVEP studies: navigation in virtual environment (VE) [18], LCD flickering frequency identification [19], classification of SSVEPs [20], and other BCI studies like P300 detection [21]. RNN has been applied in BCI studies for real-time ocular artifact suppression [22] and predicting imagined mental tasks [23]. LSTM has also been applied to BCI studies such as automated visual classification [24] and smart sensor manufacturing environment application [25]. The comparison of different classification algorithms is presented in the literature [8], [26]–[30]. Specifically, [8] and [26] provide an extensive study of different classifiers; however, CNN and LSTM were not included in the study. We compare CNN and LSTM with various traditional classification algorithms, namely, k-NN [10], MLP [11], decision trees [12], and SVM [14]. We also test the traditional classifiers in combination with sequential forward selection (SFS) [31] of features. We analyze the accuracy of the various classification algorithms by performing a leave-one-subject-out cross-validation (LOSO-CV) on an SSVEP dataset [8]. In this paper, we demonstrate the superiority of deep learning algorithms of CNN and LSTM over the traditional classification methods. SVM with Gaussian kernel employing SFS of features provided the best accuracy of 66.09% amongst the traditional classifiers, while CNN provided a superior accuracy of 69.03%.

In Section II, we describe the SSVEP dataset, preprocessing steps, and the different classification methods evaluated. In Section III, we elaborate and discuss the results. In Section IV, we summarize our work and highlight on the future scope of the study.

II. Methods

A. BCI Dataset and Preprocessing

In this study, we analyze the SSVEP dataset introduced in [8], available from the Physionet database [9]. The experiment consists of 256 channel SSVEP recordings of 11 subjects (8 males and 3 females) with a sampling frequency of 250 Hz. The stimuli are non-overlapping flickering lights from five magenta boxes with frequencies 6.66 Hz, 7.5 Hz, 8.5 Hz, 10 Hz, and 12 Hz. The objective is to distinguish between the different stimuli frequencies by analyzing the SSVEP signals. The EEG signals are normalized, bandpass filtered between 5-48 Hz, and notch filtered at 50 Hz as suggested in [8].

B. Traditional Classifiers

We investigate various traditional classification algorithms: k-NN [10], decision trees [12], MLP [11], and SVM [14].

The k-NN is a popular, simple non-parametric classification algorithm in which the grouping is performed based on proximity [10]. Euclidean distance (ED) is the commonly applied distance measure. The prediction is made based on the k training examples that are closest to the test input and the most common class is assigned as the test label. We evaluate k-NN for different values of k.

A decision tree [12] is a popular and powerful tool for classification and prediction. It generates easily understandable rules and can be applied in knowledge systems. While the decision trees are fast in generating decisions, error propagation is the major challenge associated with decision trees. If an error happens at a decision node, the effect is amplified along the path originating from the node. Overfitting and underfitting are also prominent problems in decision trees. Overfitting results in a decision tree that is more complex than necessary. Pruning techniques are adopted to overcome the challenge of overfitting. Here we implement the C4.5 [32] algorithm.

MLP is a simple neural network consisting of interconnected nodes, called neurons, which maps the input to the output class [11]. The artificial neuron receives one or more inputs (resembling dendrites), adds up the inputs according to the connection weights, and produce an output (resembling a neuron's axon). The sum is then passed through a non-linear activation function. The output y_i of a neuron i is given by:

$$y_i = \phi\bigg(\sum_{k=0}^{n} x_k w_{ik}\bigg), \tag{1}$$

where $x_1, x_2, \cdots, x_n$ are the inputs, $w_{i1}, w_{i2}, \cdots, w_{in}$ are the connection weights, and $\phi(.)$ is the activation function. The weights are randomly initialized and are updated by the backpropagation algorithm. There are mainly three layers for an MLP: the input layer, hidden layer, and the output layer. The number of neurons in the input layer is defined by the dimension of the input and the number of neurons in the output layer is determined by the number of output classes. We optimize the number of hidden layer neurons for best performance.

SVM is one of the most widely applied classifiers for BCI studies [14]. SVM aims to find the optimum hyperplane that separates the different classes to the maximum extent. It is able to perform multi-class classification. Kernel-based SVMs can be considered for non-linear classes. The major advantage of SVM is that it is independent of the dimensionality of the data. We implement two different types of SVMs: SVM with linear kernel (SVML) and SVM with Gaussian kernel (SVMG). For SVML, we optimize C (soft margin cost function parameter) and for SVMG, we optimize C and γ (free parameter of the Gaussian kernel).

We also evaluate AdaBoost [33], a popular boosting algorithm for classification. In many cases, a single training run of a classifier might not provide the best result as the training process can get trapped in a local error minimum. In AdaBoost, a set of identical classifiers is trained sequentially. We evaluate AdaBoost classifiers based on C4.5 and LDA. The different parameters of the traditional classifiers explored are detailed in Table I.

TABLE I
THE DIFFERENT PARAMETER VALUES EVALUATED FOR TRADITIONAL CLASSIFIERS

Classifier	Parameter	Values
k-NN	Distance measure k	ED 1 to 10
C4.5	Number of leaves	Optimized by CV on training data
SVML	Kernel C	Linear 2^{-5} to 2^5
SVMG	Kernel C γ	Gaussian 2^{-5} to 2^5 2^{-5} to 2^5
AdaBoost (C4.5)	Weak learner Number of learners	C4.5 100
AdaBoost (LDA)	Weak learner Number of learners	LDA 100
MLP	Number of input layer neurons Number of hidden layer neurons Number of output layer neurons	257 10, 20, 30, 50, 100 5

C. Deep Learning methods

Deep neural networks consist of several stacked layers of neurons. They make use of additional hidden layers to improve the performance of the network [15]. They are able to perform automatic feature extraction without human intervention, unlike most traditional machine learning algorithms. In deep neural networks, each layer trains on a distinct set of features depending on the output of the previous layer. As we advance further with layers, more complex features are trained upon from the aggregated output from the previous layers. We evaluate two types of deep learning methods: CNN and RNN with LSTM architecture.

CNN is a multi-layered feed-forward neural network [15]. The weights of the system are updated through the process of error backpropagation. It consists of a combination of the following types of layers: input layer, convolutional layer, rectified linear units (ReLU) layer, pooling layer, and fully connected layer. Various parameters can be optimized for the CNN network which leads to many possible configurations. The major parameters for a CNN network design are number of convolutional layers, number of convolution filters, dimension of convolution filters, number of pooling layers, number of fully connected layers, activation function, number of hidden layer neurons, drop probability, and number of iterations. We optimize different parameter values of the CNN network by brute force with one feature at a time. The training is terminated when the validation error saturates. To counter overfitting, we apply a dropout probability of 0.5. Table II details the different optimized parameters applied for CNN. The block diagram of the CNN structure is illustrated in Fig. 2.

TABLE II
OPTIMIZED PARAMETERS OF THE CNN NETWORK

Parameters	Value
Number of convolution layers	1
Number of pooling layers	1
Number of fully connected layers	1
Number of convolution filters	16
Dimension of convolution filters	1×4 each
Number of hidden layer neurons	2000
Drop probability	0.5
Activation function	ReLu

A recurrent neural network (RNN) is a type of neural network where connections between units form a directed loop. This creates an internal state of the network that allows it to exhibit dynamic temporal behavior The advantage of RNNs over feedforward neural networks is that they can utilize their internal memory to process arbitrary sequences of inputs. LSTM is an RNN architecture with the ability to

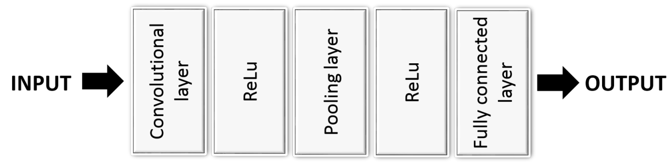

Fig. 2. The block structure of the implemented CNN network.

learn from observations when there are arbitrary long time lags between relevant events. This gives an added advantage for LSTM over alternative RNNs, hidden Markov models, and other sequence learning methods for a variety of applications. The basic structure of an LSTM cell is illustrated in Fig. 3. The optimization of LSTM parameters is performed in a sequential manner. First we optimized the number of units (in the range from 10 to 1500). Later we choose the number of epochs and best weight's initialization by applying LOSO-CV on the training set, for each step of main LOSO-CV. The optimized parameter values for LSTM are detailed in Table III.

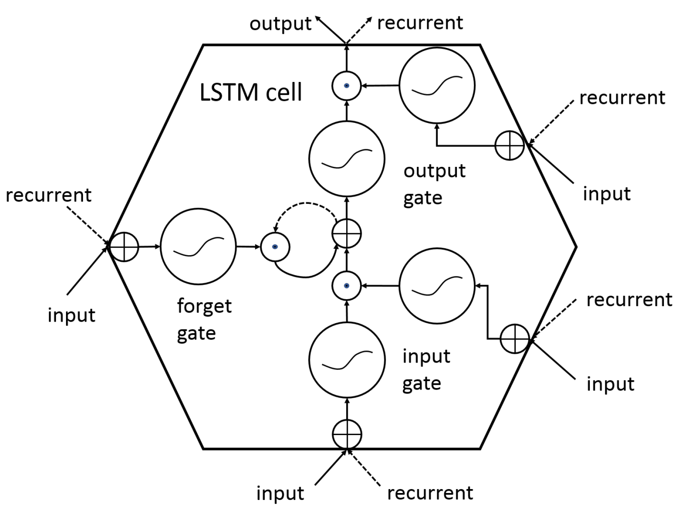

Fig. 3. The basic structure of an LSTM cell.

TABLE III
OPTIMIZED PARAMETERS OF THE LSTM NETWORK

Parameters	Value
Type	Bidirectional
Number of layers	1
Learning rate	0.001
Number of units	1100
Maximum number of epochs	200
Number of initializations	100

D. Approach

The performance of classifiers primarily depends on feature extraction and feature selection. We performed channel selection and feature extraction as suggested by Oikonomou et al. in [8]. SSVEP potential manifests most prominently on the EEG channels placed over the occipital lobe. Therefore, we extract the EEG signal from the channel-126 that was located directly above the occipital lobe. We implemented a non-parametric approach of estimating the power spectral density of the SSVEP EEG signal by applying the Welch method. The Welch method windows the signal into overlapping segments, computes periodogram of each window, and estimates the power spectral density by averaging the periodograms [34], [35]. It has been reported in [29] that Welch method is an effective feature extraction method for SSVEP problems. We applied the MATLAB implementation of the Welch method by applying the 'pwelch' function with a hamming window with 50% overlap between adjacent segments and 512 points discrete Fourier transform (DFT). We used the aforementioned power spectral density values computed in the frequency range from 0 to 125Hz, consisting of 257 points as the feature vector in our classification framework.

We apply this 257 point feature vector as the input of classification algorithms. Feature selection is a widely applied technique to boost the performance of classifiers. Here we also compare the results for traditional classifiers with feature selection based on correlation. Very often we are working with data consisting a lot of features, where only part of them are important to solve our problem. There can be redundant attributes that provide the same information as other features. Our aim is to select and apply only "interesting" attributes or in other words, select such subset of features which gives the highest possible classification accuracy. There exist many types of feature selection algorithms like filters, wrapper etc. In this paper, we develop feature ranking based on correlation coefficient values. Given a feature set $X = \{x_i | i = 1, 2, \cdots, N\}$, we find a subset $Y_M = \{x_{i1}, x_{i2}, \cdots, x_{iM}\}$, with M, N that optimizes the objective function, $J(Y)$, given by:

$$J(Y_M) = \frac{\sum_{i=1}^{M} \rho_{ic}}{\sum_{i=1}^{M} \sum_{j=i+1}^{M} \rho_{ij}}, \tag{2}$$

where ρ_{ic} is the correlation between feature i and the class label c, and ρ_{ij} is the correlation coefficient between features i and j. Subsequently, we apply SFS [31] for the different features. Here we start from an empty set and sequentially add up the next most important attribute. This method will provide the best subset of features for a selected classifier.

The traditional classifiers are implemented with MATLAB 2016a Statistics and Machine Learning Toolbox by applying the default parameters. CNN and LSTM are implemented using Tensorflow r0.12 [36] with Tesla K40 graphical processing unit (GPU) on Ubuntu 16.04. We perform a LOSO-CV on the 11 patient SSVEP dataset and the accuracies are averaged. We also perform a nested 3-fold cross-validation on the training set for optimizing the parameters of traditional classifiers as well as selecting the number of iterations for deep learning techniques.

III. Results

The different classification algorithms were tested with the SSVEP dataset and the LOSO-CV accuracies are tabulated in Table IV. The deep learning methods of CNN and LSTM shows superior accuracies in comparison with the other traditional classification algorithms. SVM has also shown better accuracies for certain subjects when applied along with feature selection. This is mainly attributed to the fact that the performance of shallow classifiers predominantly dependents on the quality of the feature selection procedure. The feature selection process improved the performance of traditional classifiers significantly, except for MLP. The deep learning methods outperformed the traditional classifiers even with the inclusion of the feature selection process. Deep learning methods are able to perform with minimal or no feature extraction and the desired features are automatically learned by the network. This makes deep learning methods suitable for studies in which the desired features are unknown.

IV. Discussion and Conclusion

The comparison studies between various classifiers have been presented in [8], [26]–[30]. In [29], Carvalho et al. made a comparison between SVM, LDA, and extreme learning machine (ELM), which is a type of MLP. ELM yields the best results in this study. Oikonomou et al. proved SVM as the best classifier for BCI problems in comparison with decision trees, LDA, k-NN, and Naive Bayes classifier [8]. In [30], Singla et al. made a comparison between SVM and artificial neural networks (ANN). ANN was tested in two different configurations: feed-forward backpropagation and cascade-forward backpropagation. In this study, SVM provides better classification results compared to ANN. Similarly, an extensive study of classification algorithms was performed by Lotte et al. in [26] and presented SVM as the superior classifier. In [28] Krusienski et al. showed that LDA performs better than SVM for P300 speller classification. In this paper, we have made the comparison between deep learning methods and various traditional classifiers for BCI problems. CNN is shown to outperform all the traditional methods available in the literature. It is a proven fact that as the size of the dataset increases, deep learning techniques tend to perform better than traditional classifiers [20]. Currently, we have only proved the superiority of deep learning on a relatively small dataset. In comparison with the traditional algorithms, deep learning techniques are complex, computationally expensive, and require a large amount of data. The new development of powerful GPUs and cloud-based shared services have made the deep learning systems cost-effective. Determining the different parameters of deep learning systems is still a major challenge.

TABLE IV
LOSO-CV classification accuracies of different classifiers on the SSVEP dataset

Classifiers	S01	S02	S03	S04	S05	S06	S07	S08	S09	S10	S11	Mean prediction accuracy
k-NN	52.17%	62.61%	39.13%	28.26%	23.48%	52.17%	31.30%	13.04%	83.48%	57.89%	64.35%	46.17%
k-NN SFS	78.26%	63.48%	36.23%	35.87%	25.22%	34.78%	61.74%	17.39%	78.26%	57.02%	73.04%	51.03%
C4.5	69.57%	53.91%	28.99%	42.39%	20.87%	30.43%	54.78%	20.29%	87.83%	58.77%	75.65%	49.41%
C4.5 SFS	66.67%	73.04%	26.09%	34.78%	18.26%	39.13%	66.09%	21.74%	85.22%	65.79%	79.13%	52.36%
AdaBoost C4.5	76.81%	44.35%	21.74%	26.09%	23.48%	22.83%	61.74%	21.74%	84.35%	45.61%	63.48%	44.75%
AdaBoost C4.5 SFS	66.67%	44.35%	21.74%	26.09%	23.48%	22.83%	61.74%	18.84%	77.39%	47.37%	63.48%	44.75%
AdaBoost LDA	88.41%	76.52%	39.13%	42.39%	17.39%	79.35%	31.30%	27.54%	93.91%	86.84%	61.74%	58.59%
AdaBoost LDA SFS	94.2%	82.61%	39.13%	55.43%	25.22%	80.43%	57.39%	24.64%	91.30%	87.72%	78.26%	65.12%
MLP	89.86%	88.70%	30.43%	47.83%	27.83%	83.70%	58.26%	27.54%	98.26%	96.49%	41.74%	62.78%
MLP SFS	88.41%	77.39%	43.48%	55.43%	23.48%	72.83%	68.70%	26.09%	88.70%	87.72%	84.35%	65.14%
SVML	75.36%	67.83%	21.74%	45.65%	23.48%	22.83%	53.91%	24.64%	96.52%	69.30%	93.91%	54.11%
SVML SFS	72.46%	79.13%	28.99%	45.65%	27.83%	41.30%	70.43%	20.29%	88.70%	88.60%	88.70%	59.28%
SVMG	89.86%	93.91%	27.54%	56.52%	23.48%	79.35%	49.57%	18.84%	97.39%	93.86%	86.09%	65.13%
SVMG SFS	89.86%	93.91%	33.33%	56.52%	22.61%	70.65%	70.43%	27.54%	97.39%	93.86%	70.90%	66.09%
CNN	90.34%	91.22%	42.24%	57.14%	27.01%	85.71%	57.32%	26.14%	98.27%	94.32%	89.65%	69.03%
LSTM	92.75%	91.30%	39.13%	53.26%	21.73%	88.04%	54.78%	17.39%	96.52%	93.91%	86.95%	66.89%

In this study, we have demonstrated the superiority of deep learning methods for the classification part of BCI systems in comparison with the traditional classification algorithms explored in the literature. We have also shown that the application of deep learning methods has reduced the dependency on the feature extraction part of BCI systems. In our future work, we intend to evaluate different datasets to substantiate the performance of deep learning. Also, in order to study the various common features of a particular BCI dataset, we plan to extract the features by the deep learning methods. This would provide a better understanding of the relation between EEG and the specified BCI tasks.

Acknowledgment

This work is supported by the FY2015 NAM Project Funding (NAM/15001), NTU-AIT-MUV Programme of Advanced Biomedical Imaging (NAM). The authors would like to thank Dr. Kavitha P. Thomas, Aamani Budhota, and Nikhita Shaju for their assistance.

References

[1] K. P. Thomas and A. Vinod, "A study on the impact of neurofeedback in eeg based attention-driven game," in *Systems, Man, and Cybernetics (SMC), 2016 IEEE International Conference on*, pp. 000320–000325, IEEE, 2016.

[2] K. P. Thomas, A. Vinod, and N. Robinson, "Online biometric authentication using subject-specific band power features of eeg," in *Proceedings of the 2017 International Conference on Cryptography, Security and Privacy*, pp. 136–141, ACM, 2017.

[3] J. Thomas, J. Jin, J. Dauwels, S. S. Cash, and M. B. Westover, "Automated epileptiform spike detection via affinity propagation-based template matching," in *Engineering in Medicine and Biology Society (EMBC), 2017 IEEE 39th Annual International Conference of the*, IEEE, in press.

[4] A. Budhota, A. Hussain, C. Hughes, C. Hansen, S. Kager, D. A. Vishwanath, C. W. K. Kuah, K. Chua, and D. Campolo, "Role of emg as a complementary tool for assessment of motor impairment," in *Biomedical Robotics and Biomechatronics (BioRob), 2016 6th IEEE International Conference on*, pp. 692–697, IEEE, 2016.

[5] S. N. Abdulkader, A. Atia, and M.-S. M. Mostafa, "Brain computer interfacing: Applications and challenges," *Egyptian Informatics Journal*, vol. 16, no. 2, pp. 213–230, 2015.

[6] American Clinical Neurophysiology Society, "Guideline 9b: Recommended standards for visual evoked potentials." http://www.acns.org/pdf/guidelines/Guideline-9B.pdf.

[7] S. J. Luck, *An introduction to the event-related potential technique*. MIT press, 2014.

[8] V. P. Oikonomou, G. Liaros, K. Georgiadis, E. Chatzilari, K. Adam, S. Nikolopoulos, and I. Kompatsiaris, "Comparative evaluation of state-of-the-art algorithms for ssvep-based bcis," *arXiv preprint arXiv:1602.00904*, 2016.

[9] A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "Physiobank, physiotoolkit, and physionet," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[10] L. E. Peterson, "K-nearest neighbor," *Scholarpedia*, vol. 4, no. 2, p. 1883, 2009.

[11] H. Ramchoun, M. Amine, J. Idrissi, Y. Ghanou, and M. Ettaouil, "Multilayer perceptron: Architecture optimization and training.," *IJIMAI*, vol. 4, no. 1, pp. 26–30, 2016.

[12] J. R. Quinlan, "Induction of decision trees," *Machine learning*, vol. 1, no. 1, pp. 81–106, 1986.

[13] D. F. Morrison, *Multivariate analysis, overview*. Wiley Online Library, 1998.

[14] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their Applications*, vol. 13, no. 4, pp. 18–28, 1998.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[16] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural computation*, vol. 1, no. 2, pp. 270–280, 1989.

[17] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[18] V. Bevilacqua, G. Tattoli, D. Buongiorno, C. Loconsole, D. Leonardis, M. Barsotti, A. Frisoli, and M. Bergamasco, "A novel bci-ssvep based approach for control of walking in virtual environment using a convolutional neural network," in *Neural Networks (IJCNN), 2014 International Joint Conference on*, pp. 4121–4128, IEEE, 2014.

[19] H. Cecotti and A. Graeser, "Convolutional neural network with embedded fourier transform for eeg classification," in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pp. 1–4, IEEE, 2008.

[20] N.-S. Kwak, K.-R. Müller, and S.-W. Lee, "A convolutional neural network for steady state visual evoked potential classification under ambulatory environment," *PloS one*, vol. 12, no. 2, p. e0172578, 2017.

[21] H. Cecotti and A. Graser, "Convolutional neural networks for p300 detection with application to brain-computer interfaces," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 3, pp. 433–445, 2011.

[22] A. Erfanian and B. Mahmoudi, "Real-time ocular artifact suppression using recurrent neural network for electro-encephalogram based brain-computer interface," *Medical and Biological Engineering and Computing*, vol. 43, no. 2, pp. 296–305, 2005.

[23] E. M. Forney and C. W. Anderson, "Classification of eeg during imagined mental tasks by forecasting with elman recurrent neural networks," in *Neural Networks (IJCNN), The 2011 International Joint Conference on*, pp. 2749–2755, IEEE, 2011.

[24] C. Spampinato, S. Palazzo, I. Kavasidis, D. Giordano, M. Shah, and N. Souly, "Deep learning human mind for automated visual classification," *arXiv preprint arXiv:1609.00344*, 2016.

[25] D. Balderas, A. Molina, and P. Ponce, "Alternative classification techniques for brain-computer interfaces for smart sensor manufacturing environments," *IFAC-PapersOnLine*, vol. 48, no. 3, pp. 680–685, 2015.

[26] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi, "A review of classification algorithms for eeg-based brain–computer interfaces," *Journal of neural engineering*, vol. 4, no. 2, p. R1, 2007.

[27] T. Hinterberger, A. Kübler, J. Kaiser, N. Neumann, and N. Birbaumer, "A brain–computer interface (bci) for the locked-in: comparison of different eeg classifications for the thought translation device," *Clinical Neurophysiology*, vol. 114, no. 3, pp. 416–425, 2003.

[28] D. J. Krusienski, E. W. Sellers, F. Cabestaing, S. Bayoudh, D. J. McFarland, T. M. Vaughan, and J. R. Wolpaw, "A comparison of classification techniques for the p300 speller," *Journal of neural engineering*, vol. 3, no. 4, p. 299, 2006.

[29] S. N. Carvalho, T. B. Costa, L. F. Uribe, D. C. Soriano, G. F. Yared, L. C. Coradine, and R. Attux, "Comparative analysis of strategies for feature extraction and classification in ssvep bcis," *Biomedical Signal Processing and Control*, vol. 21, pp. 34–42, 2015.

[30] R. Singla and B. Haseena, "Comparison of ssvep signal classification techniques using svm and ann models for bci applications," *International Journal of Information and Electronics Engineering*, vol. 4, no. 1, p. 6, 2014.

[31] A. W. Whitney, "A direct method of nonparametric measurement selection," *IEEE Transactions on Computers*, vol. 100, no. 9, pp. 1100–1103, 1971.

[32] J. R. Quinlan, *C4. 5: programs for machine learning*. Elsevier, 2014.

[33] J. Zhu, H. Zou, S. Rosset, and T. Hastie, "Multi-class adaboost," *Statistics and its Interface*, vol. 2, no. 3, pp. 349–360, 2009.

[34] A. V. Oppenheim, *Discrete-time signal processing*. Pearson Education India, 1999.

[35] J. G. Proakis, "Digital signal processing: principles, algorithms, and application-3/e.," 1996.

[36] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.