

Improving the Performance of Brain-Computer Interface using Deep Learning Algorithms and Event-Related Spectral Perturbation



Abdulmajeed Alsufyani

Abstract: Brain Computer Interface allows disabled people to communicate with the external world by using their brain signals. The main goal of a BCI is to provide patients who suffer from any neuromuscular disorders with a communication channel based on their brain signals. In this paper, the aim is to explore the effects of applying deep learning algorithms and Event Related Spectral Perturbation analyses on the performance of different EEG-based BCI paradigms. Two paradigms were investigated: one is based on the Matrix paradigm (known as oddball); and the other one utilizes the Rapid serial visual Presentation (RSVP) for presenting the stimuli. Deep learning algorithms of convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) were utilized to evaluate the two paradigms. Our findings showed that Matrix paradigm is more effective in detecting P300 signal. In terms of classification methods, deep learning of CNN algorithm has shown superiority performance in comparison with the other machine learning algorithms.

Keywords: Brain Computer interface, EEG, Deep learning, CNN, ERSP.

INTRODUCTION

BCI is a direct pathway that provides communication between the brain and a computer device. In other words it 'allows a person to communicate with or control the external world without using the brain's normal output pathways of peripheral nerves and muscles' [1], [2] [3]. In neuroscience experiments, the alternative output pathway for the brain can be one of the brain imaging methods, e.g., EEG, fMRI, and MEG. Two types of BCIs have been deployed in several studies. The first type is called Invasive in which the brain's neural activity is recorded by placing microelectrodes under a person's scalp (intracortical electrodes). A study conducted based on invasive brain imaging technique has shown the feasibility of planting some microelectrodes under a patient's scalp to generate a signal that could be used to move a cursor on the screen [4](Kennedy et al., 2000). In spite of the fact that invasive methods record electrical activity with high-quality resolution, their use is rare due to the risk coming from the need to undertake neurosurgery to plant the electrodes.

The second type is the non-invasive BCIs in which the brain's activity is recorded directly by using an appropriate brain imaging technique. It is obvious that non-invasive methods have some favourable aspects such as their simple use, durability, and the ability to use them without any risk for participants [5], [6].

BCIs that require the brain's output pathways (e.g., eye muscle) to stimulate the brain activity are called *dependent* BCIs. An example of this type of BCI is the one that presents a user with a matrix of characters and asks the user to spell a word by using eye gaze [7]. On the other hand, the *independent* BCI relies directly on the neural activity that is generated directly and immediately in response to a stimulus, such as a BCI that works by using only its user's attention to detect a target stimulus [8].

All BCIs that have been developed so far rely either on the visual or auditory input pathways (except one, which requires touch [9]). Displaying information to the user is generally considered to be quicker than playing audio, but sometimes audio is preferred (e.g. in case of a particular disability or in conjunction with, but not concurrent to, visual stimuli in order to reduce eye strain). In such cases, auditory stimuli can be used without any major concern, apart from in respect of speed [10].

A. Stimuli presentation

Different presentation techniques have been used in BCI. However, most of the paradigms were built using the Matrix design. The matrix paradigm (oddball) was first presented by [11] and is considered to be the most commonly used in brain computer interface (BCI) experiments [12]. The paradigm consists of presenting a participant with a matrix $n \times n$ of specific items (usually alphabet, digits, or alphanumeric characters). The rows and columns of the matrix are intensified in a pseudo-random sequence. Each flash lasts for a specific period (in milliseconds). A participant is usually presented with a word and the task is to spell that word by paying attention to the intensified character in the desired word and to mentally count how many times the row/column containing the target character is flashed (a standard design of matrix speller is shown in Fig. 1).

Manuscript received on March 12, 2020.

Revised Manuscript received on March 25, 2020.

Manuscript published on March 30, 2020.

* Correspondence Author

Abdulmajeed Alsufyani*, Computer Science Department, College of Computers and Information Technology, Taif University, Taif, Saudi Arabia. Email: a.s.alsufyani@tu.edu.sa

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](https://creativecommons.org/licenses/by-nc-nd/4.0/) article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

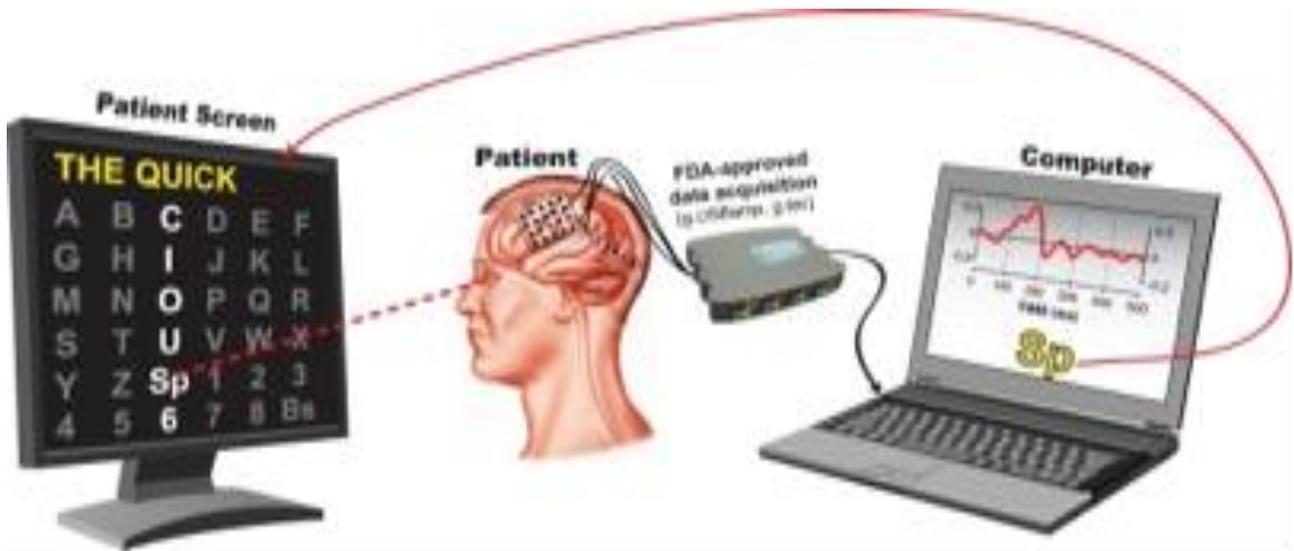


Fig. 1 The participant usually sits in front of a screen at a specific distance (usually 60 cm). The 6×6 matrix is centred on the screen, The participant's task is to focus on the intended character in the desired word 'THE QUICK' (reintroduced from [7], [13]).

The other paradigm utilises Rapid Serial Visual Presentation technique. RSVP is a presentation technique that allows a series of items to be presented in a specific position on a screen with a fixed rate. The presentation rate varies depending on the nature of the experiment, but commonly ranges between 6 to 20 items per second [14]. RSVP allows the presentation of a variety of items such as photos, words,

letters, etc. Despite the feasibility of using RSVP as a presentation method, some perceptual errors may occur in RSVP-based experiments. *Attentional blink (AB)* is one of these errors that may occur during RSVP experiments. It can be observed when the SOA between two salient stimuli is less than 500 ms [14].

B. BCI Structure

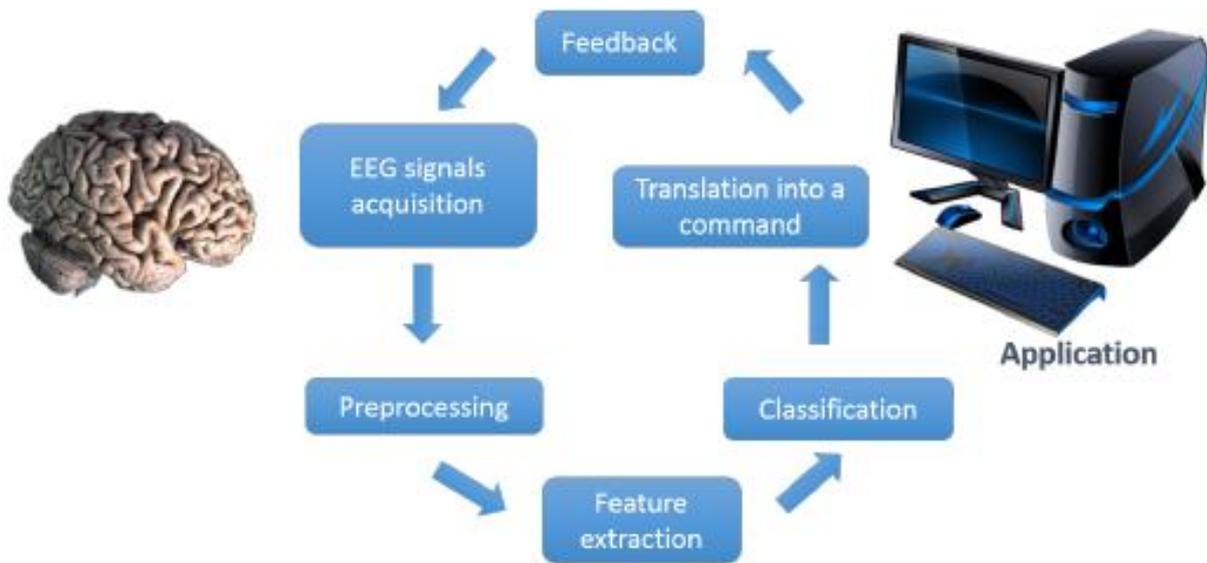


Fig. 1 The standard design of EEG-based BCI. A subject's brain activity is recorded using the EEG system; the EEG data is then analysed in order to reveal some features, then the features are classified and translated to actions.

C. Neural Activity Acquisition

Fig. 2 displays the main steps for designing a BCI. In an experiment, the brain's signals (BCI's inputs) are acquired from the participant's brain using one of the brain imaging techniques mentioned earlier. A BCI receives two types of inputs: *evoked input* or *spontaneous input* [15]. Evoked input can be elicited from the brain in response to stimulation (e.g., EEG signals produced by focusing on a

target stimulus). Spontaneous inputs are not generated by stimulation; they reflect some aspects of the brain function during mental tasks [e.g., rhythmic activity [5]]. Different brain imaging have been developed for several purposes such as EEG, fMRI and MEG.

EEG is heavily used in BCI research due to its practical advantages: both the running and installation costs are low, portability makes it easier to conduct research and the high temporal resolution makes it ideal for fast-responding BCIs. BCIs were imagined as soon as EEG was discovered, but the computational speeds at the time were far from adequate.

fMRI is also have been used in developing BCIs. Most of the fMRI-based BCI are in research on language, emotions, vision, psychological disorders and all research fields that employ neurofeedback and require high spatial resolution [16]. fMRI can tell which part of the brain is currently more active, and a patient using it can be informed of this in real time (with delays of a few seconds, which are not critical for this type of research). First, the raw fMRI output from a specific part of the brain is analysed. The system will then abstract from the resulting information to something more understandable, which is displayed back to the user in the scanner. The user then tries to manipulate his/her brain activity by looking at the feedback, closing the loop of interaction. For example, a user could manipulate his/her perception of pain by learning how to manipulate the region of the brain involved in pain perception [13].

BCIs based on MEG are currently being investigated, as it is believed that the higher spatial resolution of MEG, when compared to EEG, can increase the final speed of a BCI [17]. Even so, the practical drawbacks of MEG allow them to be used only in specific scenarios; therefore, like fMRI, they are used purely for research purposes.

D. Neural Activity Processing

Once the brain's signals are recorded, they are processed in order to improve the accuracy of the next step, which is the classification. In fact, the brain activity recorded by a brain imaging technique usually has a low signal-to-noise ratio (SNR). Thus, the noise removal (artefact rejection) is a crucial step in processing electrical data. While designing a BCI, the term 'noise' includes all data that are not relevant for BCI work. During an experiment, noise can be generated from different sources including either physiological (e.g., eye blink, or heartbeat) or environmental sources (e.g., system equipment, electrode status) [18]. Some types of noise can be detected easily even by inspection, such as an eye blink, which is usually generated on the frontal area of the scalp. These types of noise are usually removed from the recorded data by simply removing all segments that include them. However, in some occasions, the amount of noise can be vast, especially when the data are recorded from children or patients [19]. Thus, some approaches have been developed in order to detect and remove the amount of noise properly without any effects on the relevant data. Independent component analysis (ICA) is an advanced technique used commonly to treat noisy data systematically [20].

Feature selection is another step that follows artefacts rejection. Some prominent features can be extracted from the data. In fact, different BCIs can be designed based on the features that have been selected such as mu and beat rhythm activity[15], P300 evoked potentials [2], [3], [21], [22], slow cortical potentials [23](Zhao et al., 2009), and SSVEPs component [24], [25].

E. Feature Extraction and selection

In order to develop a BCI, it is important to find a way to obtain the important characteristics of the brain signals. Different methods have been deployed and used to improve the BCI performance. This section outlines some feature extraction and selection techniques.

Principle component Analyses (PCA)

It is used to extract brain signal features by transforming a set of correlated variable into a smaller set of uncorrelated variable. In EEG-based BCI, PCA used to reduce the dimensions of the extracted features. It is also used for artefact reject in several BCI applications.

Indeapendent Component Analyses (ICA)

It is a statistical technique finds the sources of mixed signals without knowing any information about the characteristics of the signals. In BCI applications, ICA has been used to remove artefacts from noisy EEG signals [19]. It can be also deployed as a classification technique in some BCI studies [26].

F. Data Classification

Once the required features have been specified, algorithms are used to classify the features. Common techniques that have been used in developing BCIs include a non-linear algorithm; linear discriminant analysis; a support vector machine [27]; and algorithms based on neural networks. In fact, the choice of selecting the appropriate classification method depends on the nature of the BCI system that one aims to develop.

EXPERIMENT

In this paper, we analysed EEG data recorded from two BCIs paradigm designed by Srivas et al. The full details of the experiment can be found in [28]. A brief summary about the experiment procedure will be outlined in the following subsection.

G. Participants

The experiment included twelve participants, six female, six male.

H. Stimulus Presentation

The experiment consisted of two presentation modes: RSVP and Matrix.

I. RSVP Paradigm

The RSVP paradigm consisted of showing subjects six 5-letter words plus one word used for a training session. Each letter in these words was considered a target letter, so each word comprised five blocks (one per letter). Within each block, there were a number of successive RSVP runs ranging between 8 and 12. Each RSVP run consists of a stream of 25 uppercase English letters randomly chosen without repetition.

The Stimulus Onset Asynchrony (SOA) was 166ms.

J. Matrix Paradigm

The Matrix paradigm (oddball) was first presented by Farwell and Donchin (1988) and is probably the most commonly used BCI speller. Similar to the RSVP paradigm, the approach consists of participants viewing six 5-letter words, and one further word used for training. Participants were faced with a 5 x 5 matrix, i.e. 25 uppercase English letters. The letters were presented in white color on a black background. For each word, participants viewed five blocks of a certain number of trials, ranging between 8 and 12. A run was defined as the flashing of all 5 rows and all 5 columns; thus, the probability of the target item being illuminated on the grid on each flash was 2/10, or 0.2. The rows and columns were illuminated in random order. Each flash lasted 133 ms, with an SOA of 166 ms.

K. Experiment Tasks

Each participant viewed both paradigms. Before initiating the stimulus presentation, participants were given IDs from 1 to 12. Odd-numbered participants performed the RSVP paradigm first, then Matrix, and vice versa for even-numbered participants. Prior to the start of the main experiment, each participant completed a training session including one word in each paradigm. Then, each participant was presented with six words in each paradigm: *right*, *world*, *black*, *quest*, *flame*, and *heavy*. The order of

presentation of these words was randomly in both paradigms.

For the duration of the presentation, participants were instructed to:

- They were permitted to blink and relax during the questions and prior to initiating the next block.
- Mentally count how many times the target character was presented (in RSVP mode) or flashed (in Matrix mode).
- Answer the counting question at the end of each block by choosing from multiple choices.

L. Data analyse and classification

EEG Data was filtered with a low-pass of 85Hz and a high-pass of 0.30Hz. For further analyses (such as ERP generation), EEGLAB version 9 was used under Matlab 2010a. ERPs were segmented from -100ms to 800ms post critical stimuli, baseline corrected from -100ms to 0ms from critical item onset.

In this paper, we investigate the performance of deep learning algorithms in in two different BCI applications. Deep learning algorithms have been used in various BCI studies [29]–[31]. Two deep learning algorithms will be used in this study: convolutional neural networks (CNN) and recurrent neural network (RNN). The block structure of any deep learning algorithm is present in Fig. 3.

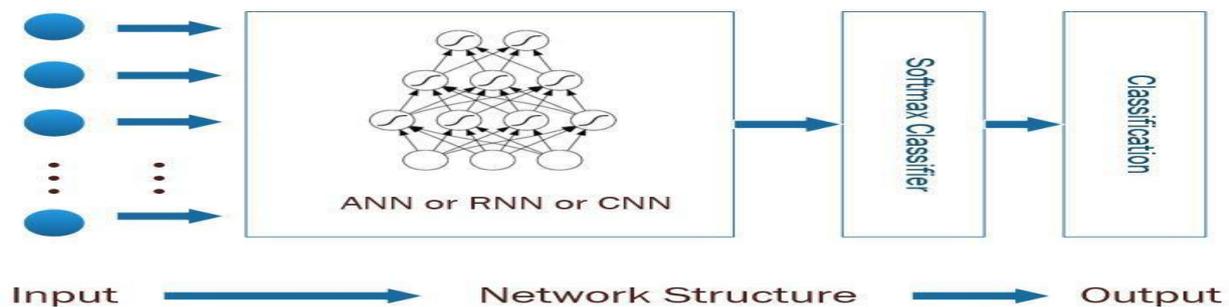


Fig. 2 The main structure of Deep learning [32].

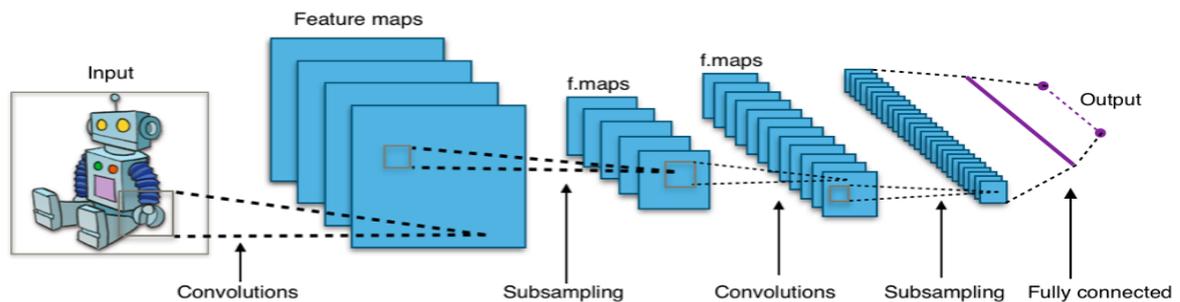


Fig. 3 The typical structure of CNN.

• CNN

CNN has been used heavily in EEG based applications. It has been applied successfully to classify SSVEPs [30]. In BCI, it was used to detect P300 signal in P300 Speller application [33][29]. Moreover, it provides a powerful tool for removing artefact for EEG signals [34]. The main characteristic of CNN is that it can extract the feature without using other feature extraction techniques. It consists of multi layers that are updated using error back

propagation. In this paper, the following layers were optimized: convolutional layer, fully connected layer, pooling layer and the filter. Table-I lists the parameters used for optimizing the neural network. Fig. 4 presents the structure of CNN algorithm.

Table -I The parameter of the CNN layers

CNN parameter	value
Convolutional layer	1
Fully connected layer	1
Pooling layer	1
Drop probability	0.7

• **RNN**

RNN has been also used in BCI applications: automated visual classification [35] and smart sensor manufacturing environment [36]. RNN learns from the previous inputs, in other words its decision is affected by the past. Unlike the CNN, RNNs learn from the training as well as from the previous inputs in order to generate outputs. The RNN model used in this paper was similar to the one applied in [29]. The typical structure of RNN is plotted in Fig. 5.

• **SVM**

The SVM is a statistical learning method used for analyzing data and classification patterns. It has been reported that the SVM performed better in EEG-based BCI. The main idea of SVM is to construct a hyperplane (or multiple hyperplanes) in a high-dimensional space that classifies two or more classes. A high level of accuracy can be achieved when the distance between the hyperplane and the nearest data point of any class is large. The selection of training and testing data was determined by adopting 4-fold cross-validation. Fifth, the SVM was applied with two different kernels: a linear kernel and a Gaussian radial basis function (RBF) kernel. The two parameters of the RBF kernel (the kernel width and the cost parameter) were adjusted using 10-fold cross-validation. Finally, the *svmclassify* Matlab function was used to evaluate the classifier performance.

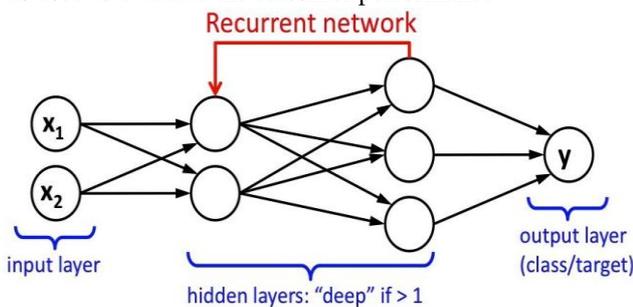


Fig. 4 The block structure of RNN model.

• **K-NN**

The K-NN algorithm is a non-parametric technique in which the proximity is used to form the grouping [37]. The input is the K closest training cases among the feature sets. The k-NN was examined on different k values from 1 to 12. In order to measure the distance, the Euclidean distance was used.

M. Classification approach of the P300

The aim was to discriminate between targets and non-targets based on the P300 of each single trial. In order to generate feature for the classification phase, Event Related Spectral Perturbation (ERSP) was applied to the raw EEG data. The ERSP calculates the mean changes in the power over

different frequencies for single trials that are time locked to the same stimulus. The output of the ERSP is a 2D Matrix of log spectral difference from the baseline (in dB). EEGLAB software was used to calculate the ERSP [38], [39]. Fig. A.1 plots the out of ERSP analyses for subject 2 EEG data. In order to compute the ERSP, the following formula was used:

$$ERSP(f, t) = 1/n \sum_{k=1}^n |F_k(f, t)|^2$$

where *t*, *f*, and *k* refer to time point, frequency and trial. The number of all trials is defined by *n*.

Each classifier was applied to a window from 300 to 800 ms from stimulus onset. This window was then sampled down to 100 epochs. The average of each segment was computed and considered as a feature. Thus, for each condition we have a set of N*100 data points, where N is the number of trials. Ten-fold cross-validation was applied. The steps of EEG data classification are shown in Fig. 7. We have also used ROC curve analyses to get a better insight about the performance of each classifier. ROC curves of subject 7 in Matrix and RSVP are plotted in Fig. 7 and Fig.8 respectively.

RESULTS AND DISCUSSION

Results of the different algorithms applied to Matrix and RSVP data are presented in table-II and table-III respectively. As can be seen from the tables, CNN has outperformed all the other algorithms in both data sets. Linear SVM also showed high performance comparing to the other traditional algorithms. Interestingly, linear SVM has also produced higher accuracies comparing to the deep learning algorithm of RNN.

Gaussian SVM has the worst performance when applied to the both data sets. It has been reported that the SVM performed better in EEG-based BCI. The main idea of SVM is to construct a hyperplane (or multiple hyperplanes) in a high-dimensional space that classifies two or more classes. A high level of accuracy can be achieved when the distance between the hyperplane and the nearest data point of any class is large. In our experiment, The selection of training and testing data was determined by adopting 4-fold cross-validation. Fifth, the SVM was applied with two different kernels: a linear kernel and a Gaussian radial basis function (RBF) kernel. The two parameters of the RBF kernel (the kernel width and the cost parameter) were adjusted using 10-fold cross-validation. Finally, the *svmclassify* Matlab function was used to evaluate the classifier performance.

It is worth to note that Matrix data has higher accuracies with all the different algorithms when compared to the RSVP data.

Our finding shows that Matrix- based BCI achieved higher performance in terms of detecting the P300. This finding comes in agreement with another study that compares between the performance of the two paradigms [28]. In [28], it has been shown that RSVP paradigm achieved less performance in terms of speller speed. Their study used Stepwise linear discriminant analyses to classify EEG data.

Improving the Performance of Brain-Computer Interface using Deep Learning Algorithms and Event-Related Spectral Perturbation

The deep learning of CNN obtained the highest accuracies compared to the other machine learning algorithms. The linear SVM comes second in terms of achieving high accuracies. This result is also in line with [29]. The aim of their study was to explore the visibility of applying deep learning algorithms in classifying EEG data of BCI applications. Their results approved the superiority of deep learning technique comparing to other machine learning algorithms.

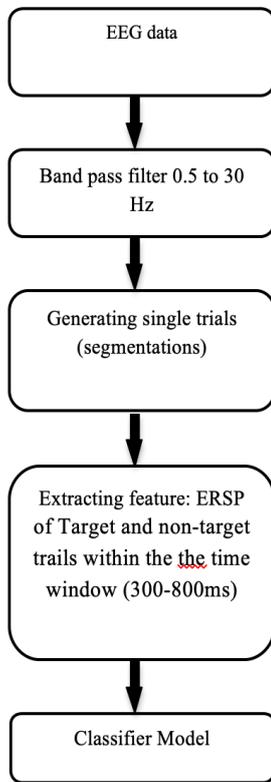


Fig. 5 The main steps for classifying EEG data.

Table-II Classification accuracies of different algorithms on Matrix data set.

Matrix-based BCI					
Subjects	Traditional Methods			Deep learning algorithms	
	Linear SVM	SVM-G	KNN	CNN	RNN
Subject 1	71.31	47.21	60.01	75.99	59.86
Subject 2	63.52	53.66	50.22	68.03	55.90
Subject 3	58.24	33.87	42.67	59.77	54.93
Subject 4	79.05	59.55	60.24	77.09	73.80
Subject 5	78.64	60.34	74.85	78.44	69.76
Subject 6	64.45	48.62	56.82	73.87	67.89
Subject 7	61.55	57.49	66.20	67.90	61.53
Subject 8	68.84	59.63	63.01	64.8	60.45
Subject 9	73.02	62.45	56.67	58.67	59.03
Subject 10	72.36	51.54	57.04	68.05	55.67
Subject 11	59.78	49.66	53.84	69.75	58.45
Average Accuracy	68.25	52.84	58.57	69.30	61.43

Table-III Classification accuracies of different algorithms on RSVP data sets.

RSVP-based BCI					
Subjects	Traditional Methods			Deep learning algorithms	
	Linear SVM	SVM G	KNN	CNN	RNN
Subject 1	61.31	42.56	58.01	71.78	57.16
Subject 2	44.54	51.78	45.02	63.43	60.04
Subject 3	52.23	39.81	49.65	55.77	54.96
Subject 4	55.05	44.71	59.24	74.16	63.68
Subject 5	73.64	58.88	72.35	72.64	58.83
Subject 6	62.43	58.67	51.32	60.97	50.37
Subject 7	76.01	59.82	63.91	79.50	72.03
Subject 8	50.84	56.88	57.44	60.92	59.84
Subject 9	61.67	55.45	55.07	54.07	56.26
Subject 10	69.36	57.54	51.54	58.55	53.01
Subject 11	55.67	52.65	59.64	63.82	52.19
Average Accuracy	60.25	52.61	56.65	65.05	58.03

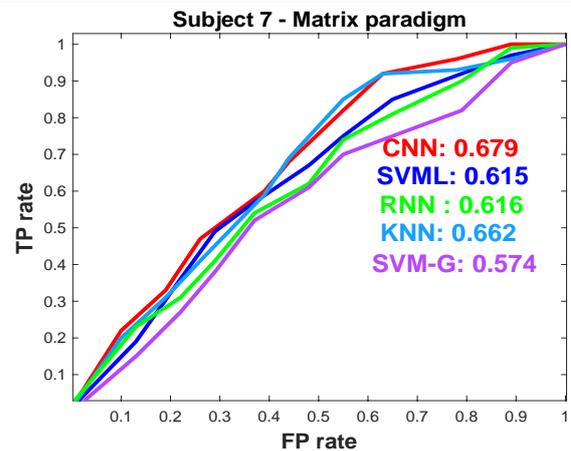


Fig. 6 ROC curve of all classifiers in Matrix data for subject 7.

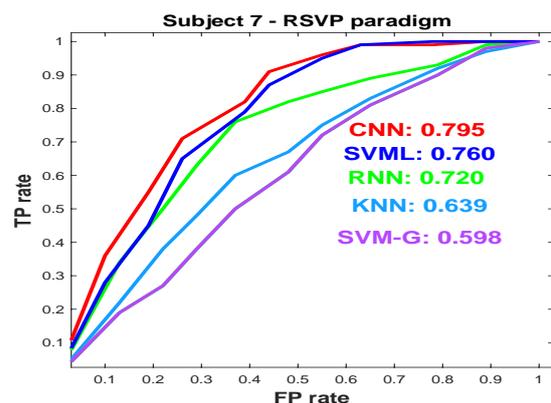


Fig. 7 ROC curve of all classifiers in RSVP paradigm for subject 7.

CONCLUSION

Brain Computer Interface allows disabled people to communicate with the external world by using their brain signals.

In this paper, the aim was to evaluate the performance of the traditional BCI paradigm which is based on the Matrix platform against the new paradigm that uses the RSVP technique. In Matrix, the paradigm consists of presenting a participant with a matrix $n \times n$ of specific items (usually alphabet, digits, or alphanumeric characters). On the other hand, RSVP paradigm consists of presenting a series of items in a specific position on a screen with a fixed rate.

The aim of the paper was to apply deep learning algorithms in order to compare the performance of the Matrix-based BCI and RSVP-based BCI. Our finding shows that Matrix-based BCI achieved higher performance in terms of detecting the P300. This finding comes in agreement with another study that compares between the performance of the two paradigms [28]. In [28], it has been shown that RSVP paradigm achieved less performance in terms of speller speed. Their study used Stepwise linear discriminant analyses to classify EEG data.

The deep learning of CNN obtained the highest accuracies compared to the other machine learning algorithms. The linear SVM comes second in terms of achieving high accuracies. This result is also in line with [29]. The aim of their study was to explore the visibility of applying deep learning algorithms in classifying EEG data of BCI applications. Their results approved the superiority of deep learning technique comparing to other machine learning algorithms.

Despite the advantages of applying deep learning algorithms to EEG data, their performance is still not good enough to obtain higher accuracies. In our results, the highest accuracy was obtained in Matrix data with subject 4 (79.05%). This weak performance among all the algorithms suggests that the EEG data used in our paper contains noise, which might prevent the algorithms from detecting the P300 component. Another reason is due to limited number of EEG single trials used in our experiment. In the future, it is recommended to design a BCI with recording EEG data from several channels and over different locations at the scalp. This would provide enough EEG data sets that allows better training for machine learning algorithms.

APPENDIX

- I. Fig. A.1 presents the ERSF transform of Target and non-target trials for subject 2. The ERSF value at each time-frequency and time point is shown by a colour value. The red colour represents an increase of the power, while the green colour indicates no change in the power. As can be seen, the target trials show a power increase between 250ms to 500ms and over 0.5Hz to 10Hz frequencies. On the other hand, non-target trial produce no changes in the power during the sam time and frequencies. The most right plot in the figure Depicts the ERSF of the difference between the ERSF of target trial and the ERSF of the non-target trials.

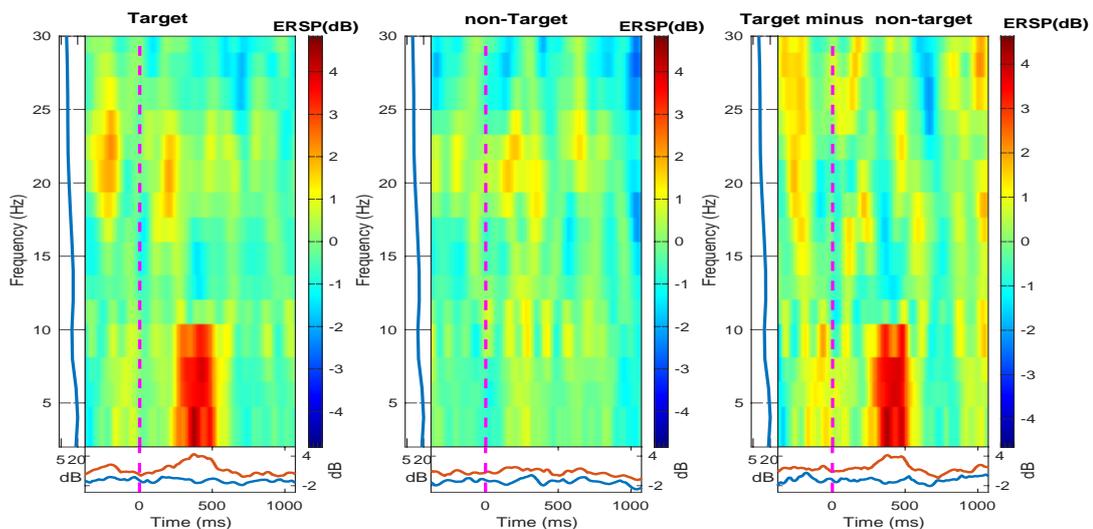


Fig. A.1 An example of ERSF transform that was calculated from the EEG data of Subject 2.

REFERENCES

1. J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain-computer interfaces for communication and control.," *Clinical neurophysiology: official journal of the International Federation of Clinical Neurophysiology*, vol. 113, no. 6, pp. 767–91, 2002.
2. E. W. Sellers and E. Donchin, "A P300-based brain-computer interface: Initial tests by ALS patients," *Clinical Neurophysiology*, vol. 117, no. 3, pp. 538–548, 2006.
3. E. Donchin, K. M. Spencer, and R. Wijesinghe, "The mental prosthesis: Assessing the speed of a P300-based brain-computer interface.," *IEEE Transactions on Rehabilitation Engineering*, vol. 8, no. 2, pp. 174–179, Jun. 2000.
4. P. R. Kennedy, R. A. E. Bakay, M. M. Moore, K. Adams, and J. Goldwithe, "Direct control of a computer from the human central nervous system.," *IEEE Transactions on Rehabilitation Engineering*, vol. 8, no. 2, pp. 198–202, Jun. 2000.
5. J. D. R. Millán, P. W. Ferrez, F. Galán, E. Lew, and R. Chavarriaga, "Non-invasive brain-actuated interaction," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2007, vol. 4729 LNCS, pp. 438–447.
6. D. J. Krusienski, E. W. Sellers, D. J. McFarland, T. M. Vaughan, and J. R. Wolpaw, "Toward enhanced P300 speller performance.," *Journal of Neuroscience Methods*, vol. 167, no. 1, pp. 15–21, 2008.
7. P. Brunner, S. Joshi, S. Briskin, J. R. Wolpaw, H. Bischof, and G. Schalk, "Does the 'P300' speller depend on eye gaze?," *Journal of neural engineering*, vol. 7, no. 5, p. 056013, 2010.

Improving the Performance of Brain-Computer Interface using Deep Learning Algorithms and Event-Related Spectral Perturbation

8. J. Cohen and J. Polich, "On the number of trials needed for P300.," *International journal of psychophysiology: official journal of the International Organization of Psychophysiology*, vol. 25, no. 3, pp. 249–55, Apr. 1997.
9. B. F. Yuksel, M. Donnerer, J. Tompkin, and A. Steed, "A novel brain-computer interface using a multi-touch surface," in *Conference on Human Factors in Computing Systems - Proceedings, 2010*, vol. 2, pp. 855–858.
10. A. Kübler, A. Furdea, S. Halder, E. M. Hammer, F. Nijboer, and B. Kotchoubey, "A brain-computer interface controlled auditory event-related potential (p300) spelling system for locked-in patients," *Annals of the New York Academy of Sciences*, vol. 1157, pp. 90–100, 2009.
11. L. A. 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, no. 6, pp. 510–523, 1988.
12. L. A. Farwell and E. Donchin, "The Truth Will Out: Interrogative Polygraphy ('Lie Detection') With Event-Related Brain Potentials," *Psychophysiology*, vol. 28, no. 5, pp. 531–547, Sep. 1991.
13. P. Brunner, A. L. Ritaccio, J. F. Emrich, H. Bischof, and G. Schalk, "Rapid communication with a 'P300' matrix speller using electrocorticographic signals (ecog)," *Frontiers in Neuroscience*, no. FEB, 2011.
14. J. E. Raymond, K. L. Shapiro, and K. M. Arnell, "Temporary suppression of visual processing in an RSVP task: an attentional blink? .," *Journal of experimental psychology. Human perception and performance*, vol. 18, no. 3, pp. 849–860, 1992.
15. J. R. Wolpaw, D. J. Mcfarland, and T. M. Vaughan, "Brain-Computer Interface Research at the Wadsworth Center," 2000.
16. G. Avidan, U. Hasson, T. Hendler, E. Zohary, and R. Malach, "Analysis of the neuronal selectivity underlying low fMRI signals," *Current Biology*, vol. 12, no. 12, pp. 964–972, 2002.
17. J. Mellinger et al., "An MEG-based brain-computer interface (BCI)," *NeuroImage*, vol. 36, no. 3, pp. 581–593, Jul. 2007.
18. M. R. N. Kousarizi, A. A. Ghanbari, M. Teshnehlab, M. Aliyari, and A. Gharaviri, "Feature extraction and classification of EEG signals using wavelet transform, SVM and artificial neural networks for brain computer interfaces," in *Proceedings - 2009 International Joint Conference on Bioinformatics, Systems Biology and Intelligent Computing, IICBS 2009, 2009*, pp. 352–355.
19. T. P. Jung, S. Makeig, M. Westerfield, J. Townsend, E. Courchesne, and T. J. Sejnowski, "Removal of eye activity artifacts from visual event-related potentials in normal and clinical subjects," *Clinical Neurophysiology*, vol. 111, no. 10, pp. 1745–1758, 2000.
20. T. P. Jung et al., "Removing electroencephalographic artifacts by blind source separation.," 2000.
21. C. C. Duncan-Johnson and E. Donchin, "On Quantifying Surprise: The Variation of Event-Related Potentials With Subjective Probability," *Psychophysiology*, vol. 14, no. 5, pp. 456–467, 1977.
22. R. Li, A. Keil, J. P.-J. of neuroscience methods, and undefined 2009, "Single-trial P300 estimation with a spatiotemporal filtering method," Elsevier.
23. M. Zhao, C. Zhao, and C. Zheng, "Identifying Concealed Information Using Wavelet Feature Extraction and Support Vector Machine," *Procedia Environmental Sciences*, vol. 8, pp. 337–343, Jan. 2011.
24. M. Cheng, X. Gao, S. Gao, and D. Xu, "Design and Implementation of a Brain-Computer Interface With High Transfer Rates," *IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING*, vol. 49, no. 10, 2002.
25. Y. Wang, R. Wang, X. Gao, B. Hong, and S. Gao, "A practical VEP-based brain-computer interface," in *IEEE Transactions on Neural Systems and Rehabilitation Engineering, 2006*, vol. 14, no. 2, pp. 234–239.
26. G. Rosas-Cholula, J. M. Ramirez-Cortes, V. Alarcón-Aquino, J. Martinez-Carballido, and P. Gomez-Gil, "On Signal P-300 Detection for BCI Applications Based on Wavelet Analysis and ICA Preprocessing. IEEE, 2010, pp. 360–365.
27. M. Kaper, P. Meinicke, U. Grossekhoefer, T. Lingner, and H. Ritter, "BCI competition 2003 - Data set Iib: Support vector machines for the P300 speller paradigm," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 6, pp. 1073–1076, Jun. 2004.
28. S. Chennu, A. Alsufyani, M. Filetti, A. M. Owen, and H. Bowman, "The cost of space independence in P300-BCI spellers.," *Journal of neuroengineering and rehabilitation*, vol. 10, no. 1, p. 82, Jan. 2013.
29. J. Thomas, T. Maszczyk, N. Sinha, T. Kluge, and J. Dauwels, "Deep learning-based classification for brain-computer interfaces," in *2017 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2017, 2017*, vol. 2017-January, pp. 234–239.
30. T. H. Nguyen and W. Y. Chung, "A single-channel SSVEP-based BCI speller using deep learning," *IEEE Access*, vol. 7, pp. 1752–1763, 2019.
31. T. Ma et al., "The extraction of motion-onset VEP BCI features based on deep learning and compressed sensing," *Journal of Neuroscience Methods*, vol. 275, pp. 80–92, Jan. 2017.
32. L. Chu, R. Qiu, H. Liu, Z. Ling, and X. Shi, "Individual Recognition in Schizophrenia using Deep Learning Methods with Random Forest and Voting Classifiers: Insights from Resting State EEG Streams," 2017.
33. H. Cecotti and A. Gräser, "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.
34. P. Nejedly et al., "Intracerebral EEG Artifact Identification Using Convolutional Neural Networks," *Neuroinformatics*, vol. 17, no. 2, pp. 225–234, Apr. 2019.
35. C. Spampinato, S. Palazzo, I. Kavasidis, D. Giordano, N. Souly, and M. Shah, "Deep Learning Human Mind for Automated Visual Classification," 2017.
36. F. Ordóñez and D. Roggen, "Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition," *Sensors*, vol. 16, no. 1, p. 115, Jan. 2016.
37. P. Soucy and G. W. Mineau, "A simple KNN algorithm for text categorization," in *Proceedings - IEEE International Conference on Data Mining, ICDM, 2001*, pp. 647–648.
38. S. Makeig, "Auditory event-related dynamics of the EEG spectrum and effects of exposure to tones *," 1993.
39. D. Ming, X. An, B. Wan, H. Qi, Z. Zhang, and Y. Hu, "A P300-speller based on event-related spectral perturbation (ERSP)," in *2012 IEEE International Conference on Signal Processing, Communications and Computing, ICSPCC 2012, 2012*, pp. 63–66.

AUTHORS PROFILE



Abdulmajeed Alsufyani. Abdulmajeed Alsufyani received the bachelor's degree (Hons.) in computer science from Taif University, Saudi Arabia, in 2006, and the master's degree in Computer Science and the Ph.D. degree in computer science from the University of Kent, U.K., in 2010 and 2015, respectively. He is currently an Assistant Professor of Computer Science at the College of Computers and Information Technology, Taif University, Saudi Arabia. His research interests include Computational Intelligence, Computational Neuroscience, Machine Learning Algorithms.