



PAPER • OPEN ACCESS

## Classification and Recognition of P300 Event Related Potential Based on Convolutional Neural Network

To cite this article: Mingjie Ma and Shuting Feng 2021 *J. Phys.: Conf. Ser.* **1952** 032007

View the [article online](#) for updates and enhancements.

You may also like

- [A comprehensive review of EEG-based brain-computer interface paradigms](#)  
Reza Abiri, Soheil Borhani, Eric W Sellers et al.
- [Influence of P300 latency jitter on event related potential-based brain-computer interface performance](#)  
P Aricò, F Aloise, F Schettini et al.
- [A hybrid BCI speller paradigm combining P300 potential and the SSVEP blocking feature](#)  
Minpeng Xu, Hongzhi Qi, Baikun Wan et al.



The Electrochemical Society  
Advancing solid state & electrochemical science & technology

**247th ECS Meeting**  
Montréal, Canada  
May 18-22, 2025  
*Palais des Congrès de Montréal*

**Showcase your science!**

**Abstracts  
due  
December  
6th**

# Classification and Recognition of P300 Event Related Potential Based on Convolutional Neural Network

Mingjie Ma\*, Shuting Feng

School of Electrical Engineering, Xi'an Jiaotong University, Xi'an, 710049, China

\*Corresponding author e-mail: mmj3119104140@stu.xjtu.edu.cn

**Abstract.** Electroencephalography plays an important role in brain science research and human disease diagnosis, especially the classification and recognition of P300 event related potential has been widely concerned. In view of the low accuracy of P300 potential recognition in brain computer interface system, a classification and recognition method of P300 event related potentials based on improved Convolution Neural Network (CNN) is proposed. First, we use *EEGLAB* toolkit for preprocessing operations such as filtering, power frequency removal, eye movement removal and normalization. The character multiclassification problem is then transformed into a dichotomous task of whether P300 signals existed, with classification labels manually made based on stimulus row and column numbers in the training data. Finally, contrast is made with the Support Vector Machine (SVM) algorithm. The results show that the CNN algorithm we employed can achieve 98.5% correct prediction on the testing data, with higher accuracy rate and reliability than the SVM algorithm. This is of great significance for the application of brain-computer interface.

**Keywords:** Brain-computer interface; P300 event related potential; convolutional neural network; classification recognition; deep learning

## 1. Introduction

In daily life, the human brain controls perception, thinking, movement, language and other functions, and uses peripheral nerves as a medium to issue instructions to various parts of the body. When the peripheral nerves of the human brain are damaged, the signal transmission pathway is blocked, so the ability to communicate and control with the outside world will be lost. Brain-computer interface (BCI) is a communication system that does not rely on the peripheral nerves or muscle tissue of the human but directly establishes communication and control channels between the brain and the computer or other electronic devices, so as to realize the transmission of information between the brain and the outside world [1]. With the rapid development of medical system application requirements and machine learning algorithms such as pattern recognition, especially the development of deep learning, BCI system has become one of the most important research fields in the world.

The classification and recognition of P300 event related potential (ERP) plays an important role in BCI system. It refers to a positive wave peak that appears about 300ms after the occurrence of a small probability stimulus. Usually, due to individual differences, the occurrence time is also different. As an endogenous component, P300 potential is not affected by the stimulation of physical properties, and



is related to perceptual or cognitive psychological activities. However, P300 event related potential is not easy to detect. Due to its small amplitude and low signal-to-noise ratio, it often submerges in spontaneous electroencephalography (EEG) and artifact interference [2]. Therefore, it is of great significance for the normal operation of BCI system to classify P300 ERP from EEG data efficiently and accurately.

In recent years, scholars at home and abroad have carried out the research on P300 ERP classification and recognition. The early classification and recognition methods are relatively simple, mainly based on the time domain characteristics of the signal processing, ignoring the frequency domain and other aspects of information, so the accuracy of P300 signal recognition is not very high [3]. Rakotomamonjy and Guigue proposed a P300 recognition method based on integrated support vector machine (SVM). Character recognition is realized by summing up all the scores of SVM. The highest score row and column are considered to be the coordinates of the characters to be detected [4]. Li Yandong et al from Tsinghua University removed the eye movement artifacts by using independent component analysis (ICA). The classification is based on SVM and bagging with patterns selected with a time window of 100-850ms after a flashing light [5]. Lin Zhonglin et al [6] considered bagging with component classifier linear discriminant analysis (LDA). Hubert Cecotti realized the classification of P300 ERP by deep learning for the first time [7]. On this basis, Liu Mingfei combined with the technique of batch normalization in training, which greatly improved the recognition accuracy of P300 ERP [8]. However, when the number of experiments is less, the recognition results still need to be improved.

In conclusion, the classification accuracy of P300 ERP needs to be improved. We present a research improvement on the structure of the network based on classical convolutional neural networks. Translating the character multiclassification problem into a dichotomous task of whether P300 signals exist, classification labels are manually made based on stimulus row and column numbers in the training data. In addition, improved preprocessing of the data leads to some improvement in the correct rate of character recognition.

## 2. The P300 BCI Experiment

### 2.1. Experimental Process

P300 ERP was usually evoked by oddball paradigm. Each subject could observe a character matrix composed of 36 characters, as shown in Figure 1. The character matrix was in rows or columns (6×6). First, the subjects were prompted to look at the target character. Secondly, entered the flashing mode of the character matrix, and flashed one row or one column of the character matrix in random order each time. Finally, when all rows or columns were flicker once, a round of experiments would be ended. In this process, when the row or column of the target character was flashing, P300 ERP would appear in the EEG signal of the subjects; when other rows or columns were flashing, P300 potential would not appear. The total number of flashes in each experiment was 12 times, and the target character was blinked twice. In the process of character recognition, if we could determine that the current EEG signal was P300 ERP, and then according to the row and column of the current EEG signal, we could determine the target character, and then completed the spelling of characters, so as to achieve the purpose of communication between the human body and the outside world.

>A

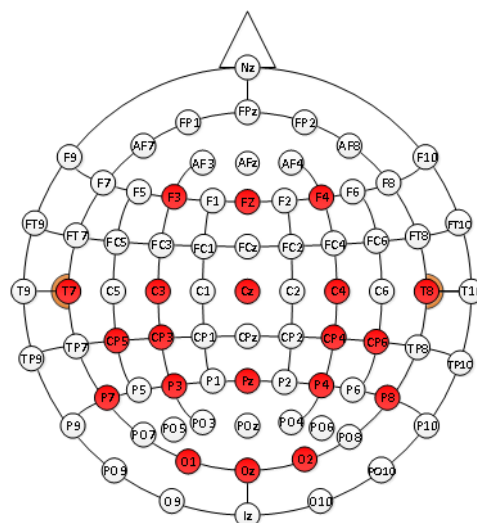
A	B	C	D	E	F
G	H	I	J	K	L
M	N	O	P	Q	R
S	T	U	V	W	X
Y	Z	1	2	3	4
5	6	7	8	9	0

**Fig. 1** Character matrix interface

## 2.2. P300 EEG Data

The experimental data were derived from the C group questions of “Huawei Cup” the 17<sup>th</sup> China Post-Graduate Mathematical Contest in Modeling. The stimulus paradigm used in this experiment is summarized as follows: The visual stimulator flickered its row or column randomly at the frequency of  $6.25\text{Hz}$ , that is, each flicker lasted for  $80\text{ms}$ , with an interval of  $80\text{ms}$  to enter the next flicker, and the experiment flickered 12 times. Each experiment corresponded to a target character, and each target character was repeated for 5 rounds. All of the rows/columns were intensified 5 times. Therefore, 10 possible P300 responses should be detected for the character recognition.

The contest provided the experimental data of P300 BCI of five healthy adult subjects with an average age of 20 years (S1~S5). In each character matrix scintillation experiment, the EEG data table contained 20 columns, each of which represented a recording channel. The recording channels were numbered in turn. Table 1 showed the identifier of the record channel, and Figure 2 showed the location of the record channel. The row of EEG data table represented the sample point data, and the sampling frequency was  $250\text{Hz}$ . The signal acquisition equipment was equipped with reference electrode and grounding electrode, and the signal of recording channel was the difference between them. The P300 EEG data of each subject were divided into training data and testing data. The training data included 12 known target characters (char01~char12), and the test data includes 10 target characters to be recognized (char13~char22). Therefore, the number of P300 to be detected was  $12 \times 2 \times 5$  and  $10 \times 2 \times 5$  respectively. A detailed introduction to the data can be used for reference [9].



**Fig. 2** EEG signal acquisition channel diagram

**Table 1.** Identifier of the acquisition channel

Identifier	Channel name	Identifier	Channel name
1	Fz	11	CP5
2	F3	12	CP6
3	F4	13	Pz
4	Cz	14	P3
5	C3	15	P4
6	C4	16	P7
7	T7	17	P8
8	T8	18	Oz
9	CP3	19	O1
10	CP4	20	O2

### 2.3. EEG Data Preprocessing

By analyzing the experimental data of BCI provided in the contest, we could see that the training data and testing data were raw data that had not been processed. If they were classified directly, the accuracy of the model would be affected. In order to recognize P300 ERP better, we preprocessed the data in frequency domain and time domain. The specific operation steps were as follows:

1) The original data was imported into *MATLAB-EEGLAB* toolkit to load and display the data, and the information of 20 channels of EEG signal could be browsed to view the voltage time waveform drawn by the original data. EEG scalp map was drawn for 20 EEG acquisition channels, and the channel positions were visualized.

2) Wave filtering: Each signal of the original data was processed by a Butterworth band-pass filter with frequency ranging from 1Hz to 30Hz. In order to eliminate the influence of the power frequency of the commercial power, the 50Hz power frequency notch is removed.

3) ICA processing: The filtered data were analyzed by independent component analysis. In this process, eye movement, blink artifacts and other noises could be removed as needed.

4) Mean normalization: In order to improve the recognition accuracy of P300 evoked potential, zero mean and unit variance of EEG signals were normalized, and then got:

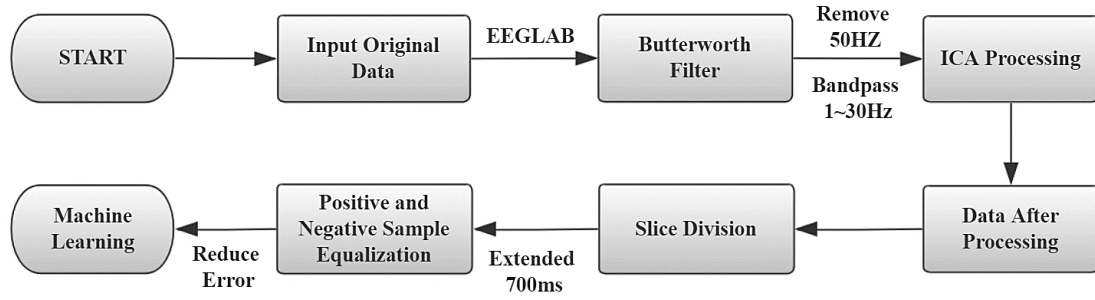
$$x^* = \frac{x - \mu}{\sigma} \quad (1)$$

Where  $\mu$  represented the mean value of all sample data,  $\sigma$  represented the standard deviation of all sample data.

5) Time domain processing: Sliced the above data and found out the sample sequence corresponding to each participant in the flash test by associating the event file information. By extending the time sequence 700ms, that is to extend 175 time series samples, so as to cover the possible P300 signals.

6) Because the same subject in the process of five rounds of experiment, might appear at the beginning of the experiment, the subject could not effectively produce P300. In order to reduce the error, we calculated the mean value of the corresponding sample sequence in five rounds of test.

7) We took the samples with P300 signal as positive samples and the samples without P300 signal as negative samples. The ratio of positive and negative samples is 1:5. For the neural network, the learning effect was not good when the samples were too unbalanced. Therefore, we had done the equalization of positive and negative samples, that is, we copied the positive sample sequence in step 6) above, so that the positive and negative sample ratio reached 1:1[10].



**Fig. 3** Data processing flow chart

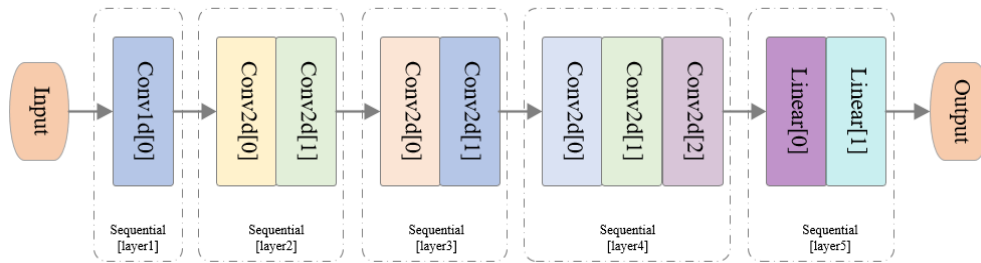
### 3. Convolutional Neural Networks

Convolutional Neural Networks (CNN) is a kind of feedforward neural network with depth structure including convolution computation. It is one of the representative algorithms of deep learning. The network has characterization learning capabilities, and can classify input information according to its hierarchical structure. Therefore, it has attracted people's eager attention in many fields such as image processing, character recognition and so on.

In view of the high continuity and poor manual recognizability of P300 data, it is difficult for humans to design features manually. Therefore, we chose a deep learning method to identify the P300 signals [11-13].

#### 3.1. Network Topology

The network structure we used is shown in Figure 4. The network consists of a 5-layer structure (excluding the input layer). We denote it as  $L_i$ , where  $i \in [0, 5]$ .



**Fig. 4** Network topology

$L_0$  is the input layer of the network model. The input data size is  $B \times C \times N_{elec} \times N_t$ , where  $B$  represents the size of batch size. The P300 signal is generated about 300ms after the stimulation, but the time is not fixed due to individual differences. Therefore, we extended the consideration time to 700ms after the stimulus: 175 consecutive samples after each stimulus are selected as a batch of data. Considering that the five rounds of experimental data of a subject may have various fluctuations or errors, we use the average of the five rounds of data as the final training data:

$$X_{i,j} = \frac{1}{K} \sum_{k=1}^k x_{i,j} \quad (2)$$

where  $i$  and  $j$  represent the  $i_{th}$  electrode channel and the  $j_{th}$  sample, and  $K$  is the test round of each subject.



$L_1$  layer performs a one-dimensional spatial convolution filter, to find the relationship between the electrode channels and the best combination of the channels. The non-linear activation function uses the non-linear rectification unit (Rectified Linear Unit, ReLU) with a simple derivative form, which is suitable for fast back propagation algorithm. At the same time, ReLU is a non-saturated activation function, which can effectively prevent the problem of gradient disappearance. The calculation expression is:

$$a^{(i)}(j) = \text{RELU}(\sum_i^{N_{elec}} a_i^{(0)}(j)w^{(1)}(j) + b^{(1)}) \quad (3)$$

where  $w^{(1)}$  and  $b^{(1)}$  represents the weight and bias of the convolution kernel of the first layer.  $i \in [1, 20]$  is electrode channel number.  $j \in [1, 175]$  is the  $j_{th}$  sample after the stimulus.

$L_2$  layer performs convolution filtering on the time domain and the space domain at the same time. The main purpose is to find the internal relationship between the EEG signal and the electrode channel in the time domain. We use Dropout function at the end of layer, which can prevent over-fitting problems caused by strong network learning ability.

$L_3$  layer adopts convolutional layer with kernel size of 5, which further improved the receptive field of neural network and extracted information from a larger area. Also simplify the number of electrode channels and the number of timings, and use the dropout layer to prevent overfitting.

$L_4$  layer is a time-domain filtering convolutional layer, whose purpose is to further filter out the time-domain related information through the feature extraction of the first few layers and prevent over-fitting.

$L_5$  layer is the output layer, which is constructed in a fully connected way. Its purpose is to use the high-level information extracted from all the previous layers for classification. The calculation expression is:

$$a^{(5)} = \text{softmax}(\sum_i^{128} a^{(4)} w^{(5)} + b^{(5)}) \quad (4)$$

where  $w^{(5)}$  is the weight of fully connected layer.  $b^{(5)}$  is the bias term of fully connected layer.

### 3.2. Training Parameters

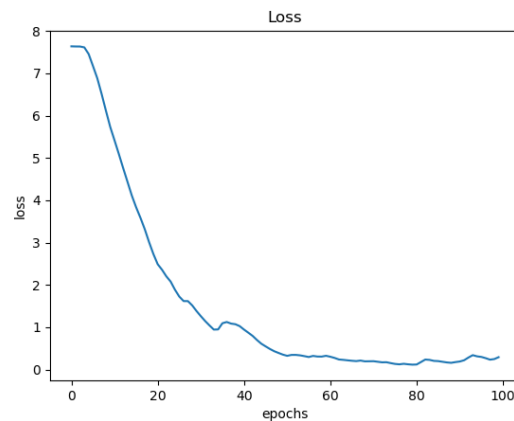
In this experiment, we set epoch=100. The initial learning rate is 1e-3. When the epoch is 40, 60, and 80, the learning rate is multiplied by the attenuation factor. The attenuation factor is set to 0.5.

We use a cross-entropy loss function, which can measure the degree of difference between different probability distributions in the same random variable. The smaller the value of cross entropy, the better the prediction effect of the model.

In order to detect whether the P300 stimulus happen is a binary classification problem. So, the expression is:

$$L = \frac{1}{N} \sum_i^N L_i = -\frac{1}{N} \sum_i^N [y_i \log(p_i) + (1 - y_i) \log(1 - p_i)] \quad (5)$$

where  $y_i$  is the label of sample  $i$ .  $p_i$  is the probability that the predicted sample  $i$  is a positive sample. The change of the loss function over time is shown in Figure 5:



**Fig. 5** Loss changes with epoch

### 3.3. Experimental Results

We used the training data of the S1, S2, and S3 subjects. After balancing the data, the ratio of the training set to the test set was 7:3. During the training process, our accuracy in detecting the presence of P300 signals can be as high as 98.5%. S1 to S5 testing data are used for the prediction results, and the final prediction results are shown in the following table 2:

**Table 2.** Prediction results based on CNN

Char 13	Char 14	Char 15	Char 16	Char 17	Char 18	Char 19	Char 20	Char 21	Char 22
M	F	5	2	I	6	K	N	A	0

### 3.4. Comparison with SVM Algorithm

Support Vector Machine (SVM) is a kind of generalized linear classifier which classifies data by supervised learning. To compare the P300 detection algorithm based on deep learning, we use SVM algorithm as a comparative verification, and further illustrate the rationality of our design method.

We use the SVM in the sklearn library to implement the algorithm, the accuracy of the SVM on the manually divided test set during the training process can only reach 89% keeping the same data division as in Section 3.1. The test set characters in the attachment are performed and results are shown in Table 3 below:

**Table 3.** Prediction results based on SVM method

Ch ar13	Char14	Ch ar15	Ch ar16	Ch ar17	Ch ar18	Ch ar19	Char20	Ch ar21	Ch ar22
M	unpredi cted	5	F	I	6	V	unpredi cted	A	0

After training, testing and predicting the data set through two different algorithms, the following table is obtained. The table records the comparison results of two different algorithms in terms of training time, test time and accuracy. Through comparison, it can be found that the two algorithms have their own advantages and disadvantages. Because the SVM algorithm will converge ahead of time, accurate time statistics cannot be obtained. For CNN, the data and model are located on the GPU during training, which requires some additional overheads such as data reading, writing and starting the GPU, so we only use the subjective time perception comparison.



**Table 4.** Comparison of two algorithms

Algorithm	Training speed	Test speed	Accuracy	Forecast reasonable
CNN	slightly slower	slightly slower	98.5%	reasonable
SVM	fast	fast	89.0%	unreasonable

Based on the machine learning SVM algorithm, the training process only needs to rely on the CPU to complete, so the training time is shorter. For relatively simple tasks, the neural network model based on deep learning can achieve high accuracy and high reliability, but the disadvantage is that the training time is long and the inference speed is slow. With the gradual popularity of computing resources, models based on deep learning can trade time for higher accuracy, which is very worthwhile.

#### 4. Conclusion

Aiming at the problem of having a low rate of correct P300 potential identification in BCI systems, we propose a P300 event related potential classification and recognition method based on improved Convolutional Neural Network (CNN). The raw data were first processed based on *MATLAB-EEGLAB* toolkit: a Butterworth power frequency notch and a 1~30Hz band-pass filter were designed, and then Independent Component Analysis (ICA) was used to remove eye movements and other interfering signals. The processed data were sliced and data set was partitioned, then used Python to implement P300 detection based on the improved CNN algorithm. Results showed that the identification accuracy of the validation set was as high as 98.5%. In addition, SVM method was used as contrast: the CNN based classification identification model had higher prediction accuracy and reliability, and gave a better fit to nonlinear problems. Therefore, the design method was validated, and opened new horizons for exploring optimizing the classification of P300 EEG signals in brain computer interface systems.

#### References

- [1] Wolpaw J R. Brain-computer interfaces (BCIs) for communication and control[J]. Supplements to Clinical neurophysiology, 2004, 57.
- [2] Chou Yuaning, Qiu Tianshuang, Zhong Mingjun. Classification and recognition of P300 event-related potentials based on convolutional neural network[J]. Chinese Journal of Biomedical Engineering, 2018, 37 (06): 657-664
- [3] Fa Rwell L A, Donchin E. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials[J]. Electroencephalography & Clinical Neurophysiology, 1988, 70 (6): 510-523.
- [4] Rakotomananjy A, Guige V. BCI competition III: Dataset II-ensemble of SVMs for BCI P300 speller [J]. IEEE Transactions on Biomedical Engineering, 2008, 55(3): 1147-1154.
- [5] Li Yandong, Ma Zhongwei, Lu Wenkai, et al. Automatic removal of the eye blink artifact from EEG using an ICA-based template matching approach[J]. Physiological Measurement, 2006, 27 (4):425-436.
- [6] L. Breiman, "Bagging Predictors," Machine Learning, vol. 26, no. 2pp. 123-140, 1996.
- [7] Cecotti H, Gr ser A. Convolutional neural networks for P300 detection with application to brain-computer interfaces [J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2011, 3(33) : 433-445.
- [8] Liu Mingfei, Wu Wei, Gu Zhenghui, et al. Deep learning based on batch normalization for P300 signal detection [J]. Neurocomputing, 2017, 275: 288-297.
- [9] <https://cpipc.chinadegrees.cn>
- [10] He H, Garcia E A. Learning from Imbalanced Data [J]. IEEE Transactions on Knowledge and Data Engineering, 2009, 21(9): 1263-1284.
- [11] Huang Anhu. Classification and recognition of P300 evoked potentials and its application in

- BCI[D]. Jinan:Shandong University,2008.
- [12] Robin, Tibor, Schirrmeister, et al. Deep learning with convolutional neural networks for EEG decoding and visualization[J]. Human Brain Mapping, 2017.
- [13] Zhang X, Wu D. On the Vulnerability of CNN Classifiers in EEG-Based BCIs[J]. IEEE transactions on neural systems and rehabilitation engineering, 2019, 27(5):814-825.