MSCNN-ADDA: A Cross-Subject P300 EEG Decoding Algorithm Based on a Multi-Scale Convolutional Neural Network and Adversarial Discriminative Domain Adaptation

Wanying He (wanying_he2000@163.com)
School of Software, South China Normal University
Guangzhou, China

Yongxi Zhou (zhouyongxi_cn@163.com)
School of Software, South China Normal University
Guangzhou, China

Jiahui Pan* (panjiahui@m.scnu.edu.cn)
School of Software, South China Normal University
Guangzhou, China

Abstract

A brain-computer interface (BCI) allows direct communication between the human brain and external devices. Despite impressive performance, previous EEG decoding algorithms still face several challenges: 1) how to shorten or eliminate the calibration process in cross-subject BCI scenarios; 2) how to capture more characteristic features from different scales in EEG data; and 3) how to extract subject-independent EEG features more effectively. To address these problems, we propose a cross-subject EEG decoding algorithm based on a multiscale convolutional neural network (MSCNN) and domain adaptation for P300-based BCIs. The MSCNN was trained on a large-scale EEG dataset to extract subject-independent features. Subsequently, we fine-tune the MSCNN through adversarial discriminative domain adaptation (ADDA) to reduce the differences among cross-subject EEG data. In offline analysis, we achieved a cross-subject average accuracy exceeding 83%, indicating that we successfully established a domain adaptation-based cross-subject EEG decoding algorithm, which can eliminate the subject-specific calibration process for new subjects.

Keywords: Brain-computer interface (BCI), P300, cross-subject, multiple scale convolutional neural network, adversarial discriminative domain adaptation

Introduction

A brain-computer interface (BCI) enables direct communication between the brain and an external device(He, Yuan, Meng, & Gao, 2020). Electroencephalograms (EEGs), commonly used as signal sources in BCI technology, reflect brain activity by putting electrodes on the scalp and measuring their potential changes. However, the significant EEG differences between individuals make it difficult to obtain a universal model applicable to all users(Jayaram, Alamgir, Altun, Scholkopf, & Grosse-Wentrup, 2016). Traditional BCIs usually require a time-consuming and inconvenient subject-specific calibration phase before application to new users to ensure its performance(Lotte et al., 2018). Therefore, proposing a EEG decoding algorithm that can shorten or eliminate the calibration time is particularly crucial.

In recent years, studies on EEG decoding algorithm have shifted from traditional machine learning(Sha'Abani, Fuad, Jamal, & Ismail, 2020) to advanced deep learning techniques(Craik, He, & Contreras-Vidal, 2019). Cecotti et al. first used CNNs as the feature extractor to detect P300 signals(Cecotti & Graser, 2010). Lawhern et al. designed the generalized EEGNet to classify EEG signals across BCI paradigms(Lawhern et al., 2018). While effective in within-subject P300 detection, these algorithms have relatively simple feature extraction modules, making the extracted features contain insufficient information for cross-subject classification. A key challenge is how to improve the feature extractor to obtain rich EEG information from different scales.

Recently, many studies have focused on using cross-subject EEG decoding algorithm based on deep learning to shorten or eliminate the BCI calibration process. Maddula et al. used a DRCNN to better extract spatial and temporal features for cross-subject P300 classification(Maddula, Stivers, Mousavi, Ravindran, & de Sa, 2017). Mijani et al. fine-tuned a pre-trained CNN model using transfer learning methods to improve cross-subject effects(Mijani, Einizade, Shamsollahi, & Beyglou, 2020). However, these algorithms still required some labeled data from target subjects to train the model, thus not achieving complete subject-independence effect. The difficulty lies in the inability to effectively extract subject-independent EEG features, possibly due to the lack of large-scale EEG data for model training(Gao et al., 2021).

To solve the problems above, we proposed a cross-subject EEG decoding algorithm based on a multiscale convolutional neural network (MSCNN) and adversarial discriminative domain adaptation (ADDA) for P300-based BCIs and trained the model on a large-scale dataset, which achieved good cross-subject effects without calibration process in offline experiments. The contributions of this paper are as follows:

- We employ an adversarial training method (ADDA) to significantly reduce the individual differences in the EEG features among subjects, which does not require labeled data from the target subjects, thus increasing the cross-subject effects of the model.
- We adopt the MSCNN as the feature extractor, which is equipped with small convolutional kernels of different sizes and can extract multi-scale P300 features from EEG

In D. Barner, N.R. Bramley, A. Ruggeri and C.M. Walker (Eds.), *Proceedings of the 47th Annual Conference of the Cognitive Science Society*

data at different scales. This allows the extracted features to contain more contextual characteristic information.

- We propose utilizing a large amount of EEG data for model training, enabling the feature extractor to effectively extract subject-independent features to enhance cross-subject EEG classification performance.

Methods

Equipment and P300 speller system

In this experiment, we used a 64-channel SynAmps2 amplifier (Compumedis, Neuroscan, Inc., Australia) to collect EEG signals at a 250 Hz sampling rate, employing a 30-channel EEG cap (LT 37) following the extended 10-20 system, referenced to the right mastoid. Electrode impedances were kept below 5k when collecting the EEG data to ensure the data reliability.

The P300 spelling system in our study can identify the characters users intend to input by detecting the P300 component in the human brain caused by the flash of corresponding letters on the screen. The graphical user interface (GUI), as shown in Fig. 1, displays a 4×10 matrix of character buttons on the screen. When stimulus began, all 40 buttons flashed sequentially and randomly, with each flash lasting 100 ms and a 30 ms interval between the start of two flashes. Each trial consists of 10 rounds, with 40 flashes per round (each button flashes once), so it requires 400 flashes (12070 ms) to identify one character. Subjects are required to face the screen directly and focus on one of the 40 buttons when the character button matrix starts flashing to input a character(Gao et al., 2023).

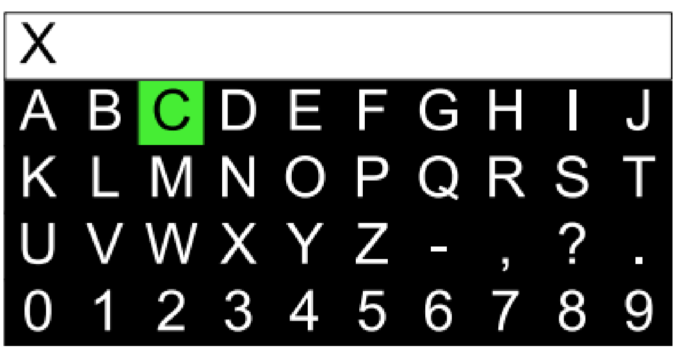

Figure 1: GUI of the P300 speller.

Data acquisition and processing

Subjects This study, approved by the Ethics Committee of Sichuan Provincial Rehabilitation Hospital, China, involved 130 healthy participants aged from 18 to 32 years who signed informed consent forms. During experiment, participants need to face the screen and focus on the flashing target character in each trial, mentally counting while their EEG signals were recorded. Each participant spelled 120 randomly generated characters, corresponding to 120 trials, divided into four sessions(Gao et al., 2021).

Data Partitions The collected EEG data were used for offline analysis of cross-subject EEG decoding algorithms. We constructed an EEG dataset of 130 participants, with 100 for training and validation (denoted as “TV-100”) while the remaining 30 for independent testing (denoted as “Te-30”).

Algorithm description

The key components of our algorithm are the feature extractor MSCNN and the transfer learning method ADDA. The MSCNN is used to extract features containing more contextual characteristic information using multi-scale convolution, while ADDA is used to reduce individual EEG differences through adversarial training. The overall MSCNN-ADDA structure is shown in Fig. 2. The input sample size is 78×30, based on a 30-channel EEG cap with 78 sampling points per channel. The specific details of the network are as follows.

MSCNN The feature extractor of this network employs a multiscale convolutional neural network. MSCNNs are able to learn features at different scales from different receptive fields, thereby capturing more hidden information in EEG signals(Kundu & Ari, 2019). The detailed framework of the MSCNN, which consists of six layers, labeled L1 to L6, is illustrated in Table 1.

Table 1: The detailed structure of the MSCNN feature extractor.

Layers	Input Size	Kernel/Operator	Output Size
L1	1×78×30	20×1×30	20×78×1
L2	20×78×1	20×4×1 20×2×1	20×38×1 20×39×1
L3	20×38×1 20×39×1	Integration	20×77×1
L4	20×77×1	Pooling	20×38×1
L5	20×38×1	Flatten	760×1
L6	760×1	Fully Connected	300×1

L1—Spatial Convolution Layer: This layer uses a (1,30) convolutional kernel with 20 channels and a stride of [1,1]. To improve the S/R and remove redundant spatial information of the signal, we use weighted superposition averaging as processing techniques. The calculation formula is as followed:

$$x^{(1)} = f(I \times k^{(2)} + b^{(2)}) \tag{1}$$

where I represents the data inputted into the network, $b^{(2)}$ denotes the additive bias, $k^{(2)}$ means the convolutional kernel matrix, and f means the activation function (tanh).

L2—Temporal Convolution Layer: This layer contains two separate convolutional layers with kernel sizes of [(2, 1), (4, 1)], each having 20 channels and a stride of [2, 1]. This special structure helps in extracting more differentiated temporal information(Pan, Cai, Huang, He, & Li, 2023). The

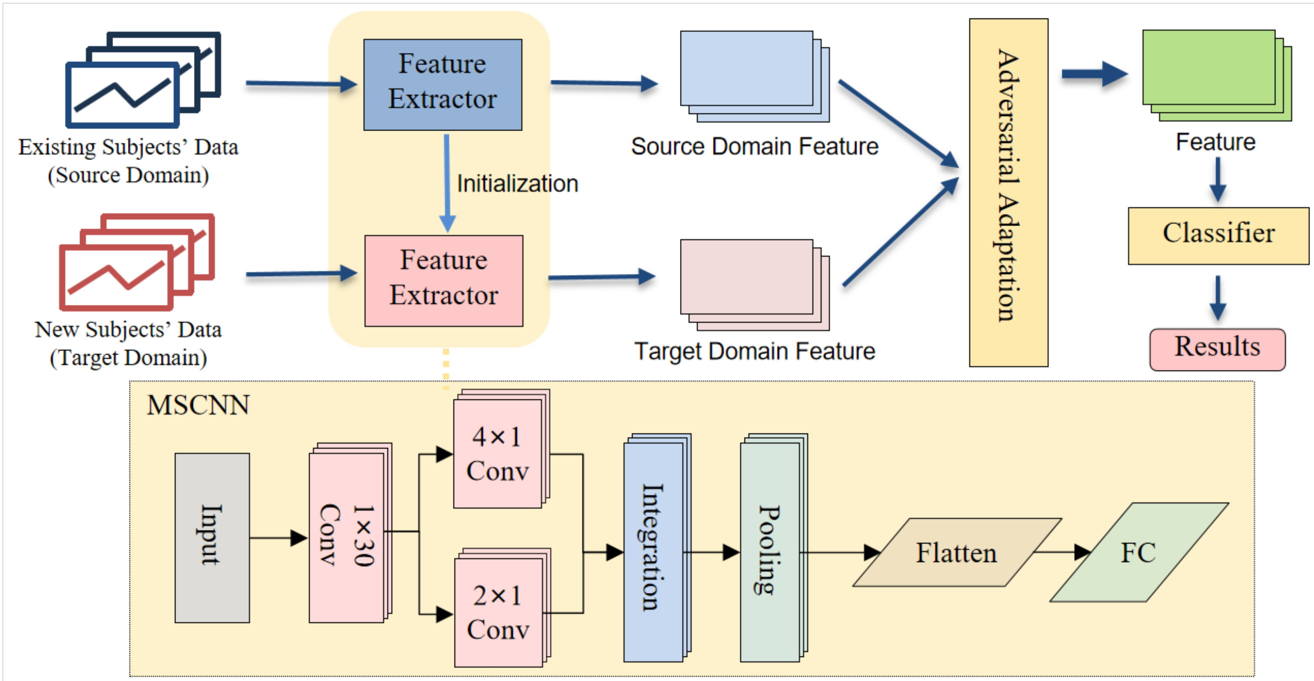

Figure 2: The structure of the proposed MSCNN-ADDA.

formulas are as follows:

$$x^{(2,1)} = f(x^{(1)} \times k^{(2,1)} + b^{(2,1)}) \tag{2}$$

$$x^{(2,2)} = f(x^{(1)} \times k^{(2,2)} + b^{(2,2)}) \tag{3}$$

where $x^{(2,1)}$ and $x^{(2,2)}$ denote the feature matrix extracted from different receptive field scales by two different kernels, containing various EEG information. $k^{(2,1)}$ and $k^{(2,2)}$ represent the convolution kernel matrix for the two different kernels, and $b^{(2,1)}$ and $b^{(2,2)}$ are the corresponding additive biases.

L3—Integration Layer: This layer integrates the two feature maps from L2. After concatenation, the features retain local EEG characteristics and capture more global information.

L4—Feature Pooling Layer: In this layer, we use a pooling operation to reduce redundant information to avoid overfitting during training, and the size of pooling filter is (2,1).

L5—Flatten Layer: The flatten layer is used to transform the input data from multi-dimensional to one-dimensional for the next fully connected layer.

L6—Fully Connected Layer: The output of this layer comprises all the extracted features from the input data. The calculation process is as follows:

$$x^{(6)} = f(x^{(5)} \times k^{(6)} + b^{(6)}) \tag{4}$$

where $k^{(6)}$ is the convolution kernel matrix, f is the tanh activation function and $b^{(6)}$ is the bias. $x^{(6)}$ is the output feature extracted by the MSCNN with dimensions of (300, 1), also used for domain adaptation in the ADDA module.

ADDA The adversarial discriminative domain adaptation (ADDA) is a domain adaptation method commonly applied in transfer learning. ADDA can reduce the feature differences between two different domains through adversarial training, allowing the model trained in the source domain can be transferred for use to the target domain(Tzeng, Hoffman, Saenko, & Darrell, 2017), which can address cross-subject issues in the BCI field. In our study, the source domain represents EEG data from existing subjects and the target domain represents EEG data from new subjects. After adversarial training between the domain discriminator and the target domain feature extractor, the domain discriminator cannot accurately distinguish between the source domain and the target domain, so the feature differences between the two domains are reduced. Therefore, the classification model trained on existing EEG data can be transferred to new subjects' EEG data to obtain the desired results.

Model training and testing

The experiments were conducted on an NVIDIA GeForce RTX 3090 GPU, with networks implemented in PyTorch 3.6. The MSCNN-ADDA algorithm framework comprises three parts: pre-training, adversarial adaptation, and testing. The complete learning process is shown in Fig. 3.

Pre-training During pre-training, the source domain feature extractor and classifier are first trained on the data from source domain. The loss is calculated as follows:

$$loss = E(X_s, Y_s) \tag{5}$$

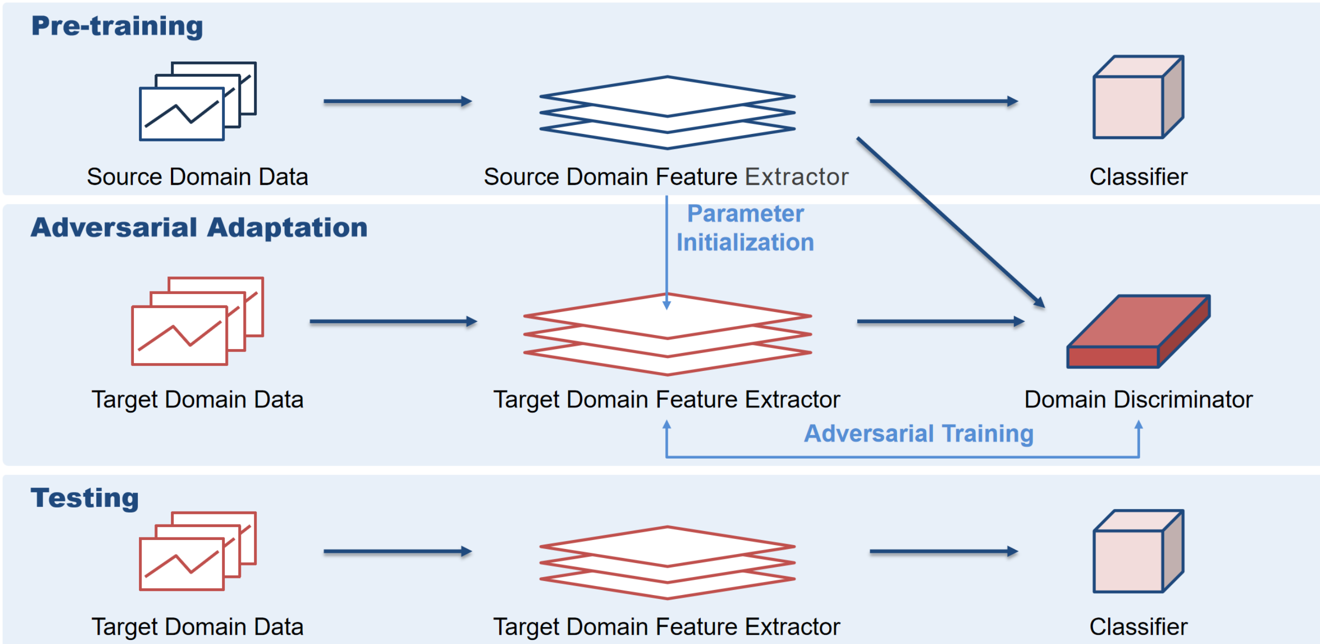

Figure 3: The learning steps of the proposed MSCNN-ADDA.

where E is the error function, X_s is the samples from source domain, and Y_s is the set of source domain labels. The loss function used here is the cross-entropy loss.

Adversarial adaptation The proposed algorithm adopts an adversarial approach for domain adaptation. Before adversarial training, the parameters of target domain feature extractor are initialized by the pre-trained source domain feature extractor. During the adversarial training between the target domain feature extractor and the domain discriminator, the data distributions of the two domains are gradually approximated through loss gradient transfer. The loss of this discriminator can be defined as (binary cross-entropy loss):

$$loss = E(X_s, X_t, M_s, M_t) \tag{6}$$

where X_s is the samples from source domain, X_t is the samples from target domain, M_s is the set of source domain features, and M_t is the set of target domain features.

Testing After training the model on the large-scale dataset, we use the target domain data (test set) to validate the cross-subject classification performance. The testing is conducted offline on the Te-30 test set. In this experiment, we adopt character spelling accuracy (the number of correctly predicted characters to the total number of input characters) as metric to evaluate the performance of the proposed algorithm.

Results and discussion

To eliminate the calibration process and improve P300 speller accuracy, we propose the cross-subject EEG decoding algorithm MSCNN-ADDA, achieving an 83.6% cross-subject average accuracy, with nearly half of the test participants reaching an accuracy of 85%, demonstrating great cross-subject performance. We validated the effectiveness of the MSCNN-ADDA on the test set Te-30, and separately verified and analyzed the contributions of the following three modules of the proposed algorithm: the feature extractor MSCNN, the large-scale dataset, and the ADDA module.

Table 2: Comparison of cross-subject classification accuracy (%) on a large-scale dataset.

Accuracy (%) \ Model / Flashing rounds	CNN-1-ADDA	EEGNet-ADDA	MSCNN-ADDA
1	23.6±12.4	20.5±10.4	**23.8±7.8**
2	31.5±11.3	28.4±9.4	**30.4±7.5**
3	36.6±11.4	37.2±9.5	**43.9±6.5**
4	43.4±11.5	48.3±9.8	**51.7±6.8**
5	52.1±13.5	53.3±9.2	**62.1±9.2**
6	58.7±10.6	57.4±10.7	**65.2±7.6**
7	61.2±10.4	62.1±8.8	**69.7±8.8**
8	63.5±10.2	67.4±11.9	**74.3±6.7**
9	66.8±10.4	70.3±8.4	**78.2±6.5**
10	70.6±12.1	76.4±9.5	**83.7±6.7**
Average	**50.8**	**52.1**	**58.3**

Comparison with other algorithms

To verify the effectiveness of MSCNN as the feature extractor, we compared it with two other commonly used EEG decoding algorithms (CNN-1 (Cecotti & Graser, 2010) and EEGNet (Lawhern et al., 2018)) on the same large-scale dataset. Table 2 shows the cross-subject average accuracies with standard deviations on Te-30 in ten flashing rounds when using different feature extractors. MSCNN achieved the highest average accuracy of 83.7%, compared to 70.6% for CNN-1 and 76.4% for EEGNet. As shown in Fig. 4, MSCNN consistently performed the best across ten flashing rounds in terms of average prediction accuracy.

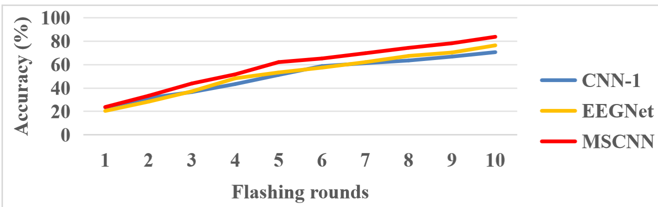

Figure 4: The average classification accuracy with respect to the number of flashing rounds obtained with different networks.

Results above show that using MSCNN as the feature extractor can achieve higher accuracy with fewer flash rounds. It is because that compared to the single-scale convolutional kernel used by the other two networks, the MSCNN employs two different convolutional kernels with the size of [(4,1),(2,1)]. The multi-scale convolution helps the network extract features containing more contextual characteristic information, enhancing its representational capability, thus improving the P300 classification performance(Han, Wang, Yang, Xue, & Hu, 2020). However, selecting an appropriate multiple-scale strategy is crucial, as too many kernels can lead to redundant multi-scale features after concatenation, thereby diminishing classification performance(Agrawal & Mittal, 2020).

The effect of large-scale data

Unlike previous works, we trained our model on a large-scale dataset with extensive subjects' EEG data. To explore the correlation between the size of training set and the cross-subject classification performance, we trained the model on training sets of different sizes and compared the cross-subject performance on the same test set. As shown in Fig. 5, we find that when the training set is relatively small, the classification accuracy is also low but increases when the training set contains more data from different subjects. When the training set included 100 participants, the average spelling accuracy of the test set reaches a relatively high level of 83.6%.

It can be observed that training the model on larger-scale training sets can lead to higher classification accuracy. The reason is that within a certain range, training the model on larger-scale training sets can help in extracting more representative subject-independent features from different subjects' EEG data and effectively enhancing the classifier performance.

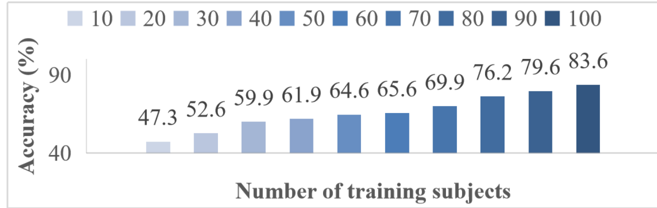

Figure 5: The trend of average accuracy with respect to the size of the training set.

Ablation study for the ADDA module

The ADDA model is incorporated into the network to enhance cross-subject performance. We validated its contribution by an ablation study on a large-scale dataset. As shown in Table 3, after adversarial training with ADDA, the average classification accuracy improved from 79.3% to 83.6% (4.3%).

Table 3: Ablation study for the ADDA module.

Model	Accuracy (%)	Variations (%)
ADDA-removed	79.3±8.6	-4.3
MSCNN-ADDA	83.6±7.1	0.0

The results show that removing the ADDA module leads to a decrease in the cross-subject classification accuracy of the algorithm, indicating that the ADDA can effectively reduce individual EEG differences by adversarial training, increasing the cross-subject classification performance of the model.

Conclusion

In this paper, we proposed MSCNN-ADDA, a cross-subject EEG decoding algorithm for P300-based BCIs. The contributions include a feature extractor (MSCNN), a domain adaptation module (ADDA), and a large-scale dataset. MSCNN is used to extract rich EEG information from multiple scales, while the ADDA module and the large-scale dataset are used to solve the problem of inconsistent EEG features across subjects. The experimental results show that the MSCNN-ADDA can eliminate the calibration phase for new users in traditional BCIs and achieve high cross-subject classification accuracy.

Acknowledgments

This work was partially supported by the STI 2030-Major Projects under grant 2022ZD0208900, and the Guangdong Basic and Applied Basic Research Foundation under grant 2024A1515010524.

References

Agrawal, A., & Mittal, N. (2020). Using cnn for facial expression recognition: a study of the effects of kernel size and number of filters on accuracy. *The Visual Computer*, *36*(2), 405–412.

Cecotti, H., & Graser, A. (2010). Convolutional neural networks for p300 detection with application to brain-computer interfaces. *IEEE transactions on pattern analysis and machine intelligence*, *33*(3), 433–445.

Craik, A., He, Y., & Contreras-Vidal, J. L. (2019). Deep learning for electroencephalogram (eeg) classification tasks: a review. *Journal of neural engineering*, *16*(3), 031001.

Gao, W., Huang, W., Li, M., Gu, Z., Pan, J., Yu, T., … Li, Y. (2023). Eliminating or shortening the calibration for a p300 brain–computer interface based on a convolutional neural network and big electroencephalography data: An online study. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *31*, 1754–1763.

Gao, W., Yu, T., Yu, J.-G., Gu, Z., Li, K., Huang, Y., … Li, Y. (2021). Learning invariant patterns based on a convolutional neural network and big electroencephalography data for subject-independent p300 brain-computer interfaces. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *29*, 1047–1057.

Han, M., Wang, R., Yang, J., Xue, L., & Hu, M. (2020). Multi-scale feature network for few-shot learning. *Multimedia Tools and Applications*, *79*, 11617–11637.

He, B., Yuan, H., Meng, J., & Gao, S. (2020). Brain–computer interfaces. *Neural engineering*, 131–183.

Jayaram, V., Alamgir, M., Altun, Y., Scholkopf, B., & Grosse-Wentrup, M. (2016). Transfer learning in brain-computer interfaces. *IEEE Computational Intelligence Magazine*, *11*(1), 20–31.

Kundu, S., & Ari, S. (2019). Mscnn: a deep learning framework for p300-based brain–computer interface speller. *IEEE Transactions on Medical Robotics and Bionics*, *2*(1), 86–93.

Lawhern, V. J., Solon, A. J., Waytowich, N. R., Gordon, S. M., Hung, C. P., & Lance, B. J. (2018). Eegnet: a compact convolutional neural network for eeg-based brain–computer interfaces. *Journal of neural engineering*, *15*(5), 056013.

Lotte, F., Bougrain, L., Cichocki, A., Clerc, M., Congedo, M., Rakotomamonjy, A., & Yger, F. (2018). A review of classification algorithms for eeg-based brain–computer interfaces: a 10 year update. *Journal of neural engineering*, *15*(3), 031005.

Maddula, R., Stivers, J., Mousavi, M., Ravindran, S., & de Sa, V. (2017). Deep recurrent convolutional neural networks for classifying p300 bci signals. *GBCIC*, *201*, 18–22.

Mijani, A., Einizade, A., Shamsollahi, M., & Beyglou, B. T. (2020). Cross-subject and crossparadigmlearning using convolutional neural network for p300 event-related potential detection. *J. Neurol. Neurosci*, *11*(5), 329.

Pan, J., Cai, H., Huang, H., He, Y., & Li, Y. (2023). Multiple scale convolutional few shot learning networks for online p300-based brain-computer interface and its application to patients with disorder of consciousness. *IEEE Transactions on Instrumentation and Measurement*.

Sha'Abani, M., Fuad, N., Jamal, N., & Ismail, M. (2020). knn and svm classification for eeg: a review. In *Inecce2019: Proceedings of the 5th international conference on electrical, control & computer engineering, kuantan, pahang, malaysia, 29th july 2019* (pp. 555–565).

Tzeng, E., Hoffman, J., Saenko, K., & Darrell, T. (2017). Adversarial discriminative domain adaptation. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 7167–7176).