Exploring Artificial Neural Network Models for c-VEP Decoding in a Brain-Artificial Intelligence Interface

Zhengqing Miao*[1,5], Anja Meunier[1,2], Michal Robert Žák[1], Moritz Grosse-Wentrup[1,3,4]
[1]Research Group Neuroinformatics, Faculty of Computer Science, University of Vienna, Vienna, Austria
[2]Doctoral School Computer Science, Faculty of Computer Science, University of Vienna, Vienna, Austria
[3]Vienna Cognitive Science Hub, University of Vienna, Vienna, Austria
[4]Research Network Data Science, University of Vienna, Vienna, Austria
[5]Department of Psychology, Justus Liebig University Giessen, Giessen, Germany
Email: mzqolive@gmail.com

***Abstract*—The Conversational Brain-Artificial Intelligence Interface (BAI) is a novel brain-computer interface (BCI) that uses artificial intelligence (AI) to help individuals with severe language impairments communicate. It translates users' broad intentions into coherent, context-specific responses through an advanced AI conversational agent. A critical aspect of intention translation in BAI is the decoding of code-modulated visual evoked potentials (c-VEP) signals. This study evaluates five different artificial neural network (ANN) architectures for decoding c-VEP-based EEG signals in the BAI system, highlighting the efficacy of lightweight, shallow ANN models and pre-training strategies using data from other participants to enhance classification performance. These results provide valuable insights for the application of ANN models in decoding c-VEP-based EEG signals and may benefit other c-VEP-based BCI systems.**

***Index Terms*—Brain-Artificial Intelligence Interface (BAI), c-VEP, EEG, chatgpt, artificial neural network (ANN).**

I. Introduction

The Conversational Brain-Artificial Intelligence Interface (BAI), a new type of brain-computer interface (BCI), leverages AI to enable users with severe language impairments to communicate effectively [1]. It operates by translating users' high-level intentions into articulate, contextually appropriate responses using a sophisticated AI-driven conversational agent. The operation of BAIs begins with the acquisition of contextual data tailored to the user's immediate environment, followed by probing for user intentions, often facilitated through conversational agents. These intentions are decoded from the brain's signals and converted into actionable commands by the AI, enabling interaction with external environments. The BAI system consists of critical components like contextual input, cognitive probing, intention decoding, and action generation.

The workflow of the BAI system is shown in Figure 1, which incorporates several key technologies, such as speech-to-text, keyword and sentence generation via ChatGPT [2], and c-VEP-based EEG decoding. Among these, the decoding accuracy of c-VEP is crucial for the BAI system to accurately capture the user's intentions, manifested as the user's ability to select the keywords on the screen according to their own will (as shown in Figure 1). The decoding of c-VEP is not only a key component of the BAI system but also the core of c-VEP-based BCI spellers. Canonical Correlation Analysis (CCA) is a commonly used method to generate class-specific spatial filters [3]. Some studies [4], [5] also employ task-discriminant component analysis (TDCA) to derive class-generic spatial filters for all calibration data. Recent research has shown that artificial neural network (ANN) models have achieved better decoding accuracy when decoding EEG signals in paradigms such as motor imagery (MI) based EEG [6], [7], P300-EEG [8], [9], and emotion recognition based EEG [10], [11]. However, research related to ANNs in decoding c-VEP based EEG is still relatively scarce [3]. One main reason is that c-VEP based EEG often can only collect a small number of training samples for system calibration, which poses challenges for training decoding models based on ANNs.

EEG2Code [12] is a convolutional neural network (CNN) that is currently influential in decoding c-VEP signals. It innovatively uses sliding window approaches to continuously predict stimulation patterns. To address the issue of insufficient sample size, EEG2Code predicts the visual stimulation pattern represented as binary sequences from the EEG signals. In the BAI system, EEG2Code was also utilized to decode the c-VEP-based EEG. However, the network architecture of EEG2Code is significantly different from classic ANN architectures for EEG decoding like ConvNet [6]. While both use CNNs for feature extraction and fully connected networks for classification, ConvNet [6] and EEGNet [9] first employ temporal convolution followed by spatial convolution, whereas EEG2Code, to reduce the number of parameters, first uses spatial convolution and then temporal convolution. In the design of the classifier, ConvNet and EEGNet both use a single fully connected layer, while EEG2Code, drawing from traditional image recognition CNNs, uses a two-layer fully connected network. It is worth noting that the approach of applying temporal convolution followed by spatial convolution in ConvNet has had a profound impact on subsequent EEG decoding work based on ANNs [8], [9], [13], [14]. Therefore, a

979-8-3503-8622-6/24/$31.00 ©2024 IEEE

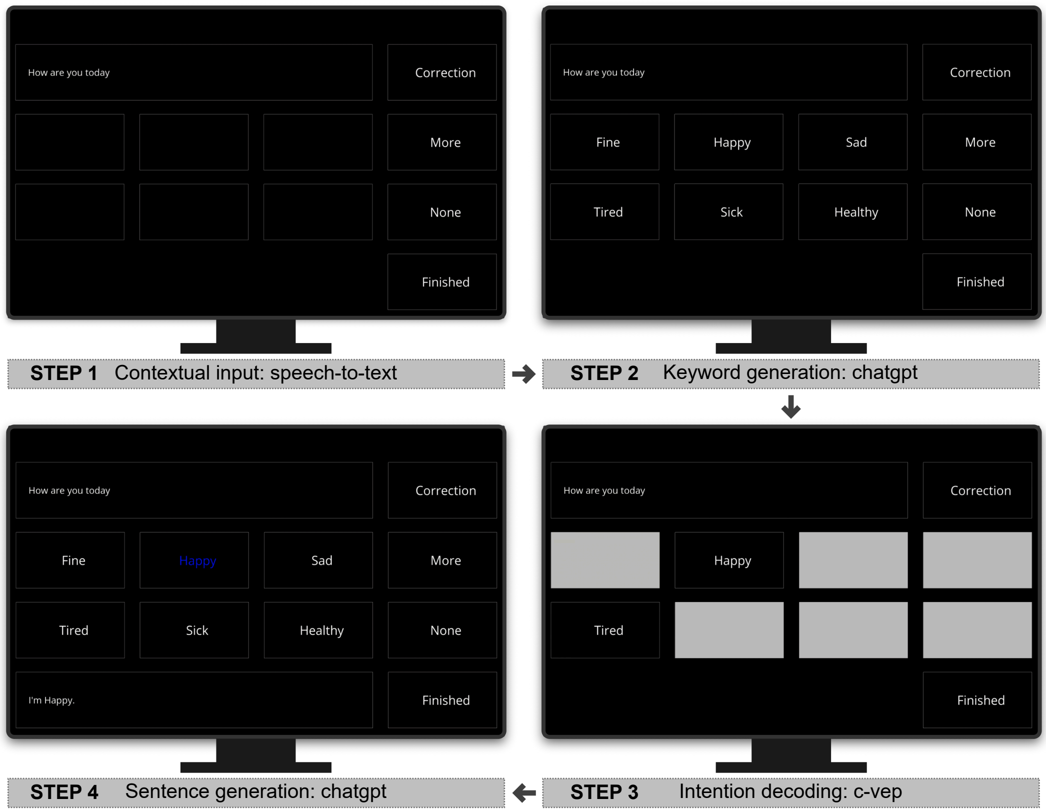

Fig. 1. The workflow of the BAI system

question worth exploring is: compared to EEG2Code, how do classic EEG decoding networks like ConvNet [6] perform in classifying c-VEP? This study builds on the EEG2Code model, which predicts each 2-bit frame, by exploring the performance of five different ANN architectures in c-VEP decoding. It aims to enhance classification performance in the BAI system and lay the groundwork for future ANN-based c-VEP decoding.

II. Methods and Materials

A. Methods

This study investigates the decoding performance of five distinct ANN architectures as outlined in Table I. Among them, EEG2Code [12] is an artificial neural network architecture specifically designed for c-VEP-based EEG and serves as a baseline network in this study. DeepConvNet [6] and ShallowConvNet [6], initially designed for MI-based EEG decoding, have been adapted in recent works for decoding emotion-based EEG. Both EEGNet [9] and ShallowNet [8] are lightweight ANN models and have demonstrated robust performance in decoding MI and event-related potentials (ERP).

Compared with traditional ANN architectures related to natural image and natural language processing, ANN architectures used for EEG decoding have significantly fewer layers. Among the network architectures mentioned above, DeepConvNet contains six layers and is the ANN architecture with the most layers in Table I. EEG2Code is the ANN architecture with the most parameters in Table I. This is because EEG2Code uses a two-layer fully connected network for classification, while other ANNs use only one layer. More parameters imply that more training data may be needed to train the model, and there is a higher risk of overfitting. Floating Point Operations (FLOPs) represent the number of floating-point operations required during the execution of the model. A smaller FLOPs value indicates lower computational cost under the same software and hardware conditions, which is advantageous in real-time online systems. As shown in Table I, the FLOPs of ShallowNet and EEGNet are both smaller than that of EEG2Code, indicating that ShallowNet and EEGNet are more efficient than EEG2Code in real-time online systems.

B. Datasets

The experimental data utilized in this study were collected from six healthy participants, labeled S01 to S06. EEG data were recorded with six channels: POz, PO3, PO4, Oz, O9, and O10, at a sampling rate of 1,000 Hz using a Brain Products EasyCap and Bittium NeurOne Tesla amplifiers. The study protocol was evaluated and approved by the Ethics Committee of the University of Vienna. In the experiment, each participant underwent three different stages of EEG data collection: system calibration stage, scenario simulation dialogue stage, and evaluation stage. In the system calibration stage, participants labeled the collected EEG data in each trial

TABLE I
FEATURES OF MODELS UTILIZED IN THIS STUDY

Features	EEGNet [9]	DeepConvNet [6]	ShallowConvNet [6]	ShallowNet [8]	EEG2Code [12]
Parameters	1,506	141,427	12,722	1,146	144,138
FLOPs(M)	1.65	11.14	6.17	1.35	2.48
Layers	4	6	3	3	5

"Layers" includes only feature extraction layers and classification layers.

TABLE II
NUMBER OF TRIALS COLLECTED FROM PARTICIPANTS

Participant ID	S01	S02	S03	S04	S05	S06
# of training trials	7	9	7	7	8	20
# of test trials	11	20	20	20	20	20

by clicking the mouse; that is, during this stage, the EEG data of each participant contained labels. This part of the data was also used as training data, as shown in the number of training trials in Table II. In this experiment, a trial recorded a complete flashing process of visual stimulation. By decoding the EEG corresponding to the entire trial, the user's choice in the current context can be determined. In this study, the EEG data collected during the scenario simulation dialogue were not analyzed because this part of the data did not have real labels.

To accurately and objectively reflect each participant's performance in the BAI system, each participant was required to answer 20 simple, predetermined questions with obvious answers, such as "What color is an elephant?" In this study, the EEG data collected in the evaluation stage were used as test data, as shown in Table II.[1] Participants S01 to S05 used the GPT-3.5 model to generate keywords during the experiment, while S06, a newly added participant, utilized the GPT-4 model. Using different ChatGPT models may affect the position of the labels in the calibration stage but does not have a substantial impact on the decoding model. It is important to note that during training, each trial's EEG data corresponding to the m-sequence stimuli was flashed seven times. However, during the evaluation stage, an early-stopping strategy based on EEG2Code was employed, resulting in variable test data lengths across participants and a potential bias in favor of the EEG2Code model during testing. For more experimental settings, interested readers can refer to [1].

C. Experimental Environment

This study employs offline analysis, but the purpose of the offline analysis is to provide a basis for improving the decoding performance of the online BAI system. Therefore, in the offline analysis, we completely simulated the data flow in the online BAI system. Although the ANN models used in this study employ sliding window methods to predict each flashing bit, in the online BAI system, the decoding model aims to determine the user's real intention; that is, the basic unit of analysis is the accuracy of each trial. Therefore, in this study, we did not discuss the decoding accuracy of each bit in detail. The results predicted by the ANN are a 2-bit output sequence of the same length as the m-sequence of visual stimulation. By calculating the correlation coefficient between this output and the m-sequence corresponding to each keyword, the real keyword selected by the user can be determined. From the interface layout in Figure 1, it can be seen that each trial's EEG label corresponds to one of 10 keywords; therefore, this study addresses a 10-class classification problem, with a chance level of 10%.

[1]Due to technical reasons, S01 only answered 11 questions in the evaluation stage.

III. EXPERIMENTS

A. From Zero Training to Full Calibration of Different Models

This subsection explores the decoding performance of various models during the calibration phase of the BAI system, ranging from zero training to using half of the calibration trials for a specific participant. As shown in Table III, the "zerotraining" scenario utilizes transfer learning with EEG data from other participants. For example, when using S01's test data as the test dataset, the training data from S02 to S05 are used to train the model. The "finetune_half" scenario builds upon "zerotraining" by further collecting half of S01's training data to fine-tune the model. Since both "zerotraining" and "finetune_half" use training data from non-target users, they can significantly expand the number of training samples and can be considered methods of transfer learning. However, due to differences in EEG data among different participants, using data from non-target users may also introduce negative transfer issues.

The method corresponding to "finetune_half" is "online_half," which simulates the data collection method of the online BAI system by using only half of the target participant's training data for training. Taking S01 as an example, "online_half" means using only the first 3 (7 // 2) trials of EEG data collected for training. Compared with "finetune_half," the amount of the target user's training data used is the same, but since "finetune_half" also includes training data from other participants, it can be used to examine whether EEG data among different subjects are transferable, the robustness of different models in coping with such transfer learning scenarios, and the performance of different ANN models when using EEG data from different participants for data augmentation.

It should be noted that, to simulate the online BAI system, this part of the experiment did not divide an additional validation set from the training set. We trained all ANN models for 30 epochs and then recorded the classification performance on the test set. The specific results are shown in Table III.

TABLE III
CLASSIFICATION ACCURACIES (%) OF DIFFERENT MODELS UNDER VARIOUS TRAINING SCENARIOS

Methods	S01	S02	S03	S04	S05	S06	Mean±Std
EEG2Code_zerotraining	82	55	65	20	55	20	49.5 ± 24.9
EEG2Code_finetune_half	91	90	75	95	60	90	83.5 ± 13.4
EEG2Code_online_half	100	90	90	65	50	95	81.7 ± 19.7
EEG2Code_online_full	100	90	95	95	60	100	90.0 ± 15.2
ShallowConvNet_zerotraining	55	30	40	50	40	15	38.3 ± 14.4
ShallowConvNet_finetune_half	82	100	90	60	60	85	79.5 ± 16.3
ShallowConvNet_online_half	82	60	70	20	35	85	58.7 ± 26.2
ShallowConvNet_online_full	100	75	70	45	50	100	73.3 ± 23.6
DeepConvNet_zerotraining	64	45	55	60	60	25	51.5 ± 14.5
DeepConvNet_finetune_half	91	95	95	85	60	95	86.8 ± 13.7
DeepConvNet_online_half	82	75	60	65	60	90	72.0 ± 12.4
DeepConvNet_online_full	100	80	90	65	60	95	81.7 ± 16.3
ShallowNet_zerotraining	64	55	55	65	60	15	52.3 ± 18.8
ShallowNet_finetune_half	91	100	90	90	60	95	87.7 ± 14.1
ShallowNet_online_half	100	90	90	85	60	100	87.5 ± 14.7
ShallowNet_online_full	100	100	95	95	60	100	91.7 ± 15.7
EEGNet_zerotraining	82	25	60	55	55	15	48.7 ± 24.6
EEGNet_finetune_half	100	100	90	100	60	90	90.0 ± 15.5
EEGNet_online_half	100	95	90	90	60	100	89.2 ± 15.0
EEGNet_online_full	100	100	95	100	60	100	92.5 ± 16.0

"Mean±Std" represents the mean and standard deviation across all participants. "zerotraining" indicates that the model was only pre-trained using training data from other participants. "finetune_half" refers to pre-training the model with other participants' data, then fine-tuning it using half of this participant's training data. "online_half" denotes that only half of the training data was collected during the online training phase. "online_full" means that all the training data was used to train the model during the online training phase (as shown in Table II).

Compared to a 10% chance level, although there are differences in average accuracies among different ANN models under the "zerotraining" scenario, ShallowConvNet has the lowest average accuracy among all participants at 38.3%, while ShallowNet achieves the highest at 52.3%. In this scenario, S01 achieves a test accuracy of 82% using both EEG2Code and EEGNet models, but the highest test accuracy for S06 is only 25%. This indicates that for S06, training using only other participants' data is insufficient.

Comparing the performances under the "finetune_half" and "online_half" scenarios, we can observe that the average accuracies of the five different ANN architectures are all higher in the "finetune_half" scenario than in the "online_half" scenario. This conclusion is more pronounced for ShallowConvNet and DeepConvNet, which have a larger number of parameters. Moreover, the average accuracies of ShallowConvNet and DeepConvNet in the "finetune_half" scenario exceed those in the "online_full" scenario, indicating that these two ANN architectures require more training data.

Compared to EEG2Code, ShallowNet and EEGNet achieve higher average test accuracies in almost all scenarios, suggesting that lightweight and shallow ANNs have advantages in decoding c-VEP-based EEG. Additionally, ShallowNet and EEGNet exhibit smaller performance differences across the "finetune_half," "online_half," and "online_full" scenarios, indicating that lightweight and shallow ANNs require less training data. In the "online_full" scenario, we find that the differences among different ANN models in S01, S02, S05, and S06 are not significant, suggesting that improvements in ANN decoding models have limited effect on enhancing decoding accuracy for some participants. Comparing the decoding accuracies of S05 with other participants, we can see that S05's test accuracies are lower than those of other participants across all decoding architectures, which may be due to the nature of the EEG signals themselves.

Considering the classification performance across different scenarios and ANN models, lightweight and shallow network architectures like ShallowNet and EEGNet may have advantages in decoding within the online BAI system.

B. *Volatility of Different Models*

In deploying the online BAI system, the robustness of ANN models during training is crucial, as it is challenging to reasonably partition a validation set from the training set that can represent the training data distribution. Without a validation set, the number of training epochs becomes a hyperparameter for ANN models. However, in a multi-user BAI system, it is difficult to optimize this parameter for different users. An ideal ANN model should have stable classification performance after a certain number of training epochs, so that the model's performance does not significantly differ if trained for one more or one fewer epoch.

Figure 2 illustrates that volatility varies across different ANN models. As can be seen from the figure, ANN models do not require hundreds or thousands of training epochs; most ANN models can achieve relatively stable classification performance on the test set after training for no more than 15 epochs on the training set. ShallowConvNet, DeepConvNet, and EEG2Code exhibit greater fluctuations, especially after ten training epochs, compared to ShallowNet and EEGNet. This

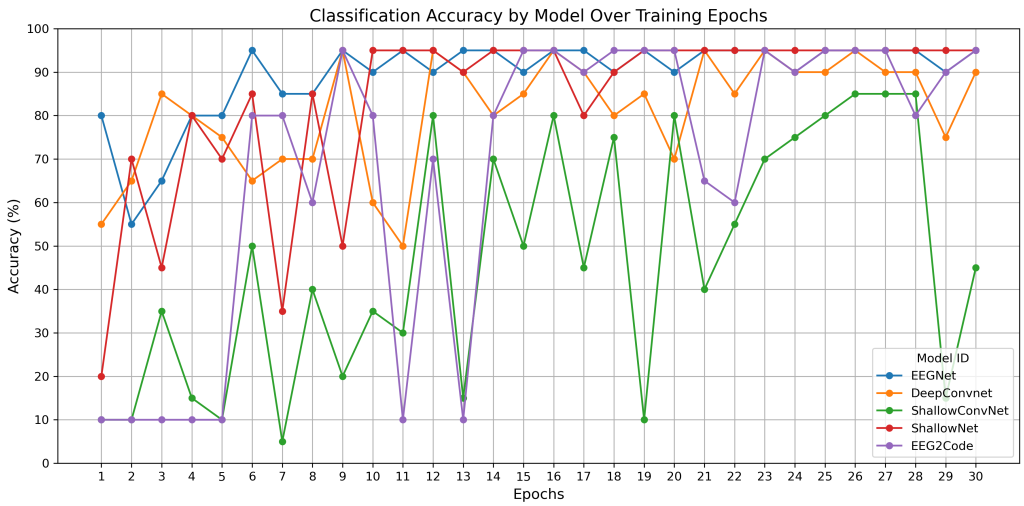

Fig. 2. Classification accuracy on the test dataset of different ANN models across 1 to 30 training epochs using all training data, illustrated with S03.

indicates that lightweight and shallow ANN models have better performance in handling volatility during training, possibly due to the small amount of training data.

C. Dilemma of online c-VEP decoding via ANNs

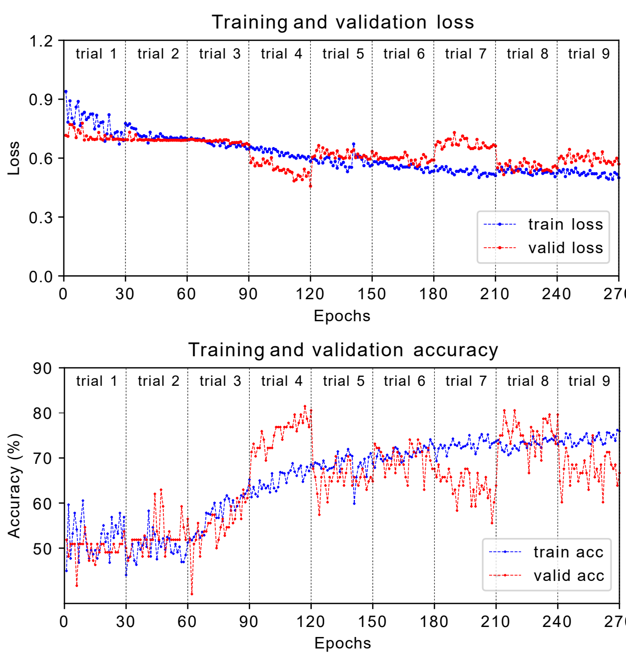

Fig. 3. Variations in validation accuracy and loss in bit-prediction with increasing training data during the online calibration of the BAI system (illustrated with S03 and EEG2Code, similar performances are also observed in other participants and models).

One significant challenge faced by ANN models in online deployment within the BAI system is determining when the model has been adequately calibrated. This is also the reason why, in Section III-B, a validation set was not used to monitor the training process of the ANN model; a detailed explanation is provided below.

In machine learning, model selection is traditionally based on the loss or accuracy of a validation dataset. However, in the online BAI system, raw EEG data is collected continuously as a data stream. The question then arises: how many trials of EEG data need to be collected from each participant to meet the system's calibration requirements? Paper [1] follows the method from EEG2Code [12] , which employs a sliding window prediction approach. Specifically, it associates the EEG data within the sliding window with the visual stimuli's black (binary 0) and white (binary 1) flickers. The accuracy of predicting the 2-bit frames indirectly reflects the ANN model's ability to decode c-VEPs, that is, whether the EEG signal collected in each trial can accurately reflect the user's intended selection. Given the abundance of training data, it is plausible to divide the data into a training set and a validation set to assess whether the BAI system has been calibrated by observing the validation set's accuracy or loss. An intuitive approach involves using a new trial of EEG data collected during the training phase as the validation set, with the EEG data collected prior to that trial serving as the training set. This ensures that both the training and validation sets have labels and aligns with the real usage scenario of an online BAI system. The results corresponding to this method are shown in Figure 3.

Ideally, an increase in training data should lead to higher validation accuracy, and when the validation accuracy no longer improves or reaches a certain threshold, it indicates that the ANN model is well-trained and the BAI system is adequately calibrated. However, as Figure 3 reveals, validation accuracy does not consistently increase with more training data but instead depends on the selection of trials for validation. In some cases, using just four trials for training can cause the validation accuracy of the ANN model to jump from 60% to approximately 80% when the fourth trial is used for validation.

This spike does not necessarily indicate adequate training, as adding more training samples can subsequently decrease validation accuracy. This phenomenon complicates the assessment of ANN training levels by monitoring validation loss or accuracy, as is commonly done in computer vision. Therefore, collecting extensive training data and selecting robust ANN models, as demonstrated in Figure 2, is critical for the stable operation of the BAI system.

IV. Conclusions

Calibration is a critical preparatory step in the BAI system, primarily involving the decoding of c-VEP based EEG signals. This study examined five different ANN architectures, providing valuable insights for BAI systems and other c-VEP based BCI applications. Generally, lightweight ANN models like EEGNet and ShallowNet outperformed more parameter-heavy models such as ShallowConvNet, DeepConvNet, and EEG2Code, particularly when calibration data was reduced by half. Leveraging data from other participants for auxiliary training proved effective, with some achieving as high as 82% classification accuracy under zero training conditions. This indicates the efficacy of pre-training ANNs with other participants' data, especially for parameter-heavy models. Additionally, the volatility of different ANN models during training was observed to vary, with lightweight models like ShallowNet and EEGNet showing more stable performance across various training epochs. Results from Table III and Figure 3 suggest that ShallowNet and EEGNet are more suitable for decoding in BAI systems, a finding that may also be relevant to other c-VEP based BCI systems. Finally, this research highlighted the challenges of online calibration using frame prediction c-VEP decoding methods, where it is difficult to gauge model training adequately by monitoring validation metrics due to the high variability in performance across trials. A potential solution is to use auxiliary training with data from other participants while collecting as many calibration samples as possible.

References

[1] Anja Meunier, Michal Robert Žák, Lucas Munz, Sofiya Garkot, Manuel Eder, Jiachen Xu, and Moritz Grosse-Wentrup. A conversational brain-artificial intelligence interface. *arXiv preprint arXiv:2402.15011*, 2024.

[2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[3] Víctor Martínez-Cagigal, Jordy Thielen, Eduardo Santamaria-Vazquez, Sergio Pérez-Velasco, Peter Desain, and Roberto Hornero. Brain–computer interfaces based on code-modulated visual evoked potentials (c-vep): a literature review. *Journal of Neural Engineering*, 18(6):061002, 2021.

[4] Bingchuan Liu, Xiaogang Chen, Nanlin Shi, Yijun Wang, Shangkai Gao, and Xiaorong Gao. Improving the performance of individually calibrated ssvep-bci by task-discriminant component analysis. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 29:1998–2007, 2021.

[5] Yining Miao, Nanlin Shi, Changxing Huang, Yonghao Song, Xiaogang Chen, Yijun Wang, and Xiaorong Gao. High-performance c-vep-bci under minimal calibration. *Expert Systems with Applications*, 249:123679, 2024.

[6] Robin Tibor Schirrmeister, Jost Tobias Springenberg, Lukas Dominique Josef Fiederer, Martin Glasstetter, Katharina Eggensperger, Michael Tangermann, Frank Hutter, Wolfram Burgard, and Tonio Ball. Deep learning with convolutional neural networks for eeg decoding and visualization. *Human brain mapping*, 38(11):5391–5420, 2017.

[7] Zhengqing Miao and Meirong Zhao. Time–space–frequency feature fusion for 3-channel motor imagery classification. *Biomedical Signal Processing and Control*, 90:105867, 2024.

[8] Zhengqing Miao, Meirong Zhao, Xin Zhang, and Dong Ming. Lmda-net: A lightweight multi-dimensional attention network for general eeg-based brain-computer interfaces and interpretability. *NeuroImage*, 276:120209, 2023.

[9] Vernon J Lawhern, Amelia J Solon, Nicholas R Waytowich, Stephen M Gordon, Chou P Hung, and Brent J Lance. Eegnet: a compact convolutional neural network for eeg-based brain–computer interfaces. *Journal of neural engineering*, 15(5):056013, 2018.

[10] Nazmi Sofian Suhaimi, James Mountstephens, and Jason Teo. Eeg-based emotion recognition: a state-of-the-art review of current trends and opportunities. *Computational intelligence and neuroscience*, 2020(1):8875426, 2020.

[11] Tengfei Song, Wenming Zheng, Peng Song, and Zhen Cui. Eeg emotion recognition using dynamical graph convolutional neural networks. *IEEE Transactions on Affective Computing*, 11(3):532–541, 2018.

[12] Sebastian Nagel and Martin Spüler. World's fastest brain-computer interface: Combining eeg2code with deep learning. *PLOS ONE*, 14(9):1–15, 09 2019.

[13] He Zhao, Qingqing Zheng, Kai Ma, Huiqi Li, and Yefeng Zheng. Deep representation-based domain adaptation for nonstationary eeg classification. *IEEE Transactions on Neural Networks and Learning Systems*, 32(2):535–545, 2020.

[14] Yonghao Song, Qingqing Zheng, Bingchuan Liu, and Xiaorong Gao. Eeg conformer: Convolutional transformer for eeg decoding and visualization. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 31:710–719, 2022.