# Double-Scale Attention-Based Temporal Convolutional Network for Steady-state Visual Evoked Potential Signal Classification

Xinyi Jiang, Ruimin Wang, Yue Leng, Keiji Iramina, Wenming Zheng, *Member, IEEE*, and Sheng Ge*, *Member, IEEE*

***Abstract*—The steady-state visual evoked potential (SSVEP)-based brain computer interface (BCI) system has received significant attention due to its high transmission rate and multi-command capabilities. However, for new application systems, the lengthy and expensive process of data collection impedes the widespread use of BCI. In the past two decades, SSVEP-based decoding algorithms have made significant progress. Recently, deep-learning algorithms have been widely used in SSVEP decoding. However, limitations are still obvious in offline analysis due to the need for large amounts of training data and lengthy training processes. To achieve low-cost and rapid calibration, we propose an end-to-end SSVEP decoding model based on a Temporal Convolutional Network, with a two-step training strategy to achieve fast calibration of a subject-dependent model in new BCI experimental scenarios in which applications are displayed with insufficient calibration data. The model has good generalization ability and has achieved information transmission rates of 169.4 bits/min and 123.27 bits/min on two publicly available 40-class public SSVEP datasets with limited calibration data. In this study, multiple performance metrics of the proposed method were compared with other state-of-the-art algorithms. In addition, the feasibility of establishing an online system for building SSVEP classification models is discussed with the aim of adapting models to new application scenarios in the real world with a lack of subjects and calibration data.**

***Index Terms*—Deep Learning; Brain Computer Interface; SSVEP; TCN.**

## I. Introduction

A Brain-computer interface (BCI) system is a communication system that enables interaction with the external world through the decoding of brain signals [1]. Among them, the BCI system based on steady-state visual evoked potentials (SSVEP) has attracted substantial attention in the fields of rehabilitation and psychology because of its non-invasiveness, high signal-to-noise ratio, high transmission rate, and multi-command capabilities [2]. SSVEP is a neurophysiological phenomenon whereby a periodic visual stimulus can elicit a sustained neural response in the visual cortex of the brain that is phase-locked to the stimulus frequency, and it has been utilized in the design of online spelling systems that allow users to select letters or commands by fixating their gaze on flickering targets [3]. By modulating the frequency, phase, location, and intensity of these targets, it is possible to elicit distinct SSVEP responses that can be decoded to control a variety of computer interfaces [4]. Despite its potential as a non-invasive and reliable method for BCI applications, there are still challenges in designing robust and user-friendly systems that can adapt to individual differences [5].

The current research on SSVEP decoding algorithms is mainly divided into two categories: non-training algorithms and training algorithms. Canonical correlation analysis (CCA) [6] and filter bank CCA (FBCCA) [7] are representative non-training algorithms, utilizing the reference signal template derived from the specific visual stimulus paradigm to detect targeted frequency. In contrast, training algorithms often achieve superior identification accuracy by extracting subject-specific feature information [8]. In 2015, Chen et al. proposed the modified extended CCA (mECCA) algorithm [9], which combines a reference signal template with an individual average signal template to improve identification accuracy. In 2018, Nakanishi et al. introduced task-related component analysis (TRCA) as a novel algorithm for spatial filter calculation, and provided an advanced method called ensemble TRCA (eTRCA) by integrating the spatial filter of all stimuli, which is calculated by maximizing the inter-trial covariance of each stimulus target. [10]. Similarly, Kumer et al. proposed the sum of squared correlations (SSCOR) and its ensemble version (eSSCOR) to optimize individual SSVEP templates by learning the common SSVEP representation space in 2019 [11]. In 2020, Wong et al. improved the multiset CCA algorithm by adding a reference signal template [12]. And in 2021, Liu et al. proposed the task-discriminant component analysis (TDCA) algorithm, which enhances signals by increasing data dimensions through the use of time delays, outperforming the TRCA algorithm [13].

In addition to the above supervised algorithms, deep-learning methods have also rapidly developed and achieved notable accomplishments in the field of SSVEP decoding. In 2018, Waytowich et al. proposed a compact convolutional neural network (CompactCNN) based on deep separable convolution, achieving an accuracy of approximately 80% in the 12-class SSVEP subject-dependent classification task, making it the

This work was supported in par by the National Natural Science Foundation of China under Grant 62271141, and in part by the Fundamental Research Funds for the Central Universities under Grants 2242021k30014 and 2242021k30059.

Xinyi Jiang, Yue Leng, Wenming Zheng, and Sheng Ge are with the Key Laboratory of Child Development and Learning Science, Ministry of Education, School of Biological Science and Medical Engineering, Southeast University, Nanjing 210096, China.

Ruimin Wang is with the Research Center for Brain Communication, Kochi University of Technology, Kami 7828502, Japan.

Keiji Iramina is with the Graduate School of Systems Life Sciences, Kyushu University, Fukuoka 8190395, Japan.

*Corresponding author (e-mail: shengge@seu.edu.cn).

state-of-the-art (SOTA) SSVEP decoding model [14]. In 2019, Podmore et al. proposed PodNet and a single-subject training optimization scheme, achieving an accuracy of 77% with a 2-s time window length on the Benchmark dataset [15]. Although PodNet's performance level is slightly lower than that of FBCCA, its optimization scheme has positively influenced subsequent research. Recently, Guney et al. utilized signal sub-band filtering for signal enhancement and proposed a deep neural network-based model [8]. This model achieved a high transmission rate of 265.23 bits/minute on the Benchmark dataset, surpassing CompactCNN and becoming the most outstanding model in recent years in terms of classification performance. Besides the above models based on raw signals, Cecotti was the first to propose a deep learning model based on fast Fourier transform (FFT) [16]. Subsequently, Ravi et al. further used FFT by combining the real and imaginary parts of the complex frequency spectrum as inputs to the model, reporting an accuracy of 92.33% on a 1-s time window for the 12-class classification task [17]. In 2022, Xujin and his team reported a transformer model that combines both temporal and frequency streams, achieving an accuracy of 88% in the no-training 40-class classification task using a 2-s time window [18].

Although studies have explored the feasibility of using calibration-free models for subject-independent SSVEP classification [18, 19], these models rely on training data collected from a large number of subjects in the same experimental setting. Note that electroencephalogram (EEG) signals can vary significantly across different experimental settings due to various factors, such as differences in the data collection environment and equipment [20]. Furthermore, studies have shown that EEG signals can also vary greatly among individuals [8, 10, 13, 14]. Therefore, collecting calibration data from individual subjects is necessary for achieving high transmission rates in online BCI systems. However, the process of collecting EEG data can be time-consuming and fatiguing for the subjects, making it difficult to collect a large number of calibration data [18, 20]. These factors have resulted in the current research on deep-learning methods for SSVEP classification remaining in the offline analysis stage, as it is still a challenge to generate subject-dependent models for online systems without a sufficient number of calibration data. Therefore, it is essential to explore methods that can adapt quickly to new experimental settings, reduce computational costs, and generate individualized models rapidly to achieve an online SSVEP model generation system that is not reliant on a large number of calibration data.

In this study, we propose a temporal convolutional neural network-based SSVEP classification model (DATCNet) along with a two-step training strategy. The proposed model was evaluated on two publicly available 40-class SSVEP datasets, and its performance was measured using multiple metrics. Our aim is to investigate the feasibility of quickly calibrating subject-specific models for online SSVEP-BCI systems. To achieve this, we present a detailed training process and explore the potential of our model for online applications.

## II. Methods and Materials

### A. SSVEP dataset description

We utilized two SSVEP public datasets, the Benchmark dataset [21] and Beta dataset [20], to evaluate our model. Both datasets employed joint frequency–phase modulation [9] to encode visual stimuli, with identical designs for target number ($N_{target} = 40$), stimulus frequencies ($8$–$15.8\,\mathrm{Hz}$, with $0.2\,\mathrm{Hz}$ intervals), and phase intervals (0.5 $\pi$). The specific stimulus configuration is presented below, where $k$ represents the index of the target stimulus, $C_A$ denotes the set of target stimuli, $f_0$ is set to $8\,\mathrm{Hz}$, and $\Phi_0$ is 0.

$$f_k = f_0 + (k-1)\Delta f, \phi_k = \phi_0 + (k-1)\Delta\phi, k \in C_A \quad (1)$$

The benchmark dataset [21] comprises data from 35 participants, each of whom completed six blocks of testing in a laboratory within an electromagnetically shielded room. Each block consisted of a single trial of all 40 targets, with a prompt time of $0.5\,\mathrm{s}$, a stimulus duration of $5\,\mathrm{s}$, and an end time of $0.5\,\mathrm{s}$, resulting in a total of $6\,\mathrm{s}$ of EEG data collected from 64 channels, which were then downsampled to $250\,\mathrm{Hz}$. The Beta dataset [20] is a supplementary dataset for the Benchmark dataset, with different target designs that simulate the use of SSVEP in real-life scenarios by using the arrangement of keys on a QWERTY virtual keyboard. Data were collected from 70 participants in a low signal-to-noise ratio non-laboratory environment. To accommodate real-world demands, the number of calibration blocks was reduced to four, and the stimulus duration was reduced from $5\,\mathrm{s}$ to $2\,\mathrm{s}$ (S1–S15) and $3\,\mathrm{s}$ (S16–S70).

In this study, we focused on the EEG data from the occipital and parietal regions [O1, O2, Oz, Pz, PO3, PO4, PO5, PO6, PO7, PO8, and POz], as SSVEP has been shown to have distinct features in the visual response areas of the brain [22, 23]. To account for visual latency [7], we selected the data segments from [$0.14\,\mathrm{s}$, $0.14\,\mathrm{s}$ + $d$] after the onset of stimulation, where $d$ is the time window length used in the analysis. Additionally, to preprocess the SSVEP signal and remove noise interference such as electromagnetic and muscle artifacts, we applied a notch filter at $50\,\mathrm{Hz}$ and a sixth-order Butterworth band-pass filter between $5\,\mathrm{Hz}$ and $70\,\mathrm{Hz}$.

The experimental paradigms and data collection for both the Benchmark and Beta datasets were approved by the Research Ethics Committee of Tsinghua University, and the protocols used in this study were approved by the ethics committee of the Affiliated Zhongda Hospital of Southeast University (No. 2016ZDSYLL002-Y01).

### B. Architecture of the proposed model: DATCNet

In this study, we propose the double-scale attention-based temporal convolutional network (DATCNet) for SSVEP signal classification. The proposed model is an end-to-end architecture that takes raw EEG signals ($X_t{\in}R[C{\times}T]$) as input and predicts the target stimuli for different SSVEP frequencies. The proposed DATCNet model consists of four modules: (1) Double-Scale Module, (2) Attention Module, (3) Temporal Convolutional Network (TCN) Module, and (4) Fully-connected

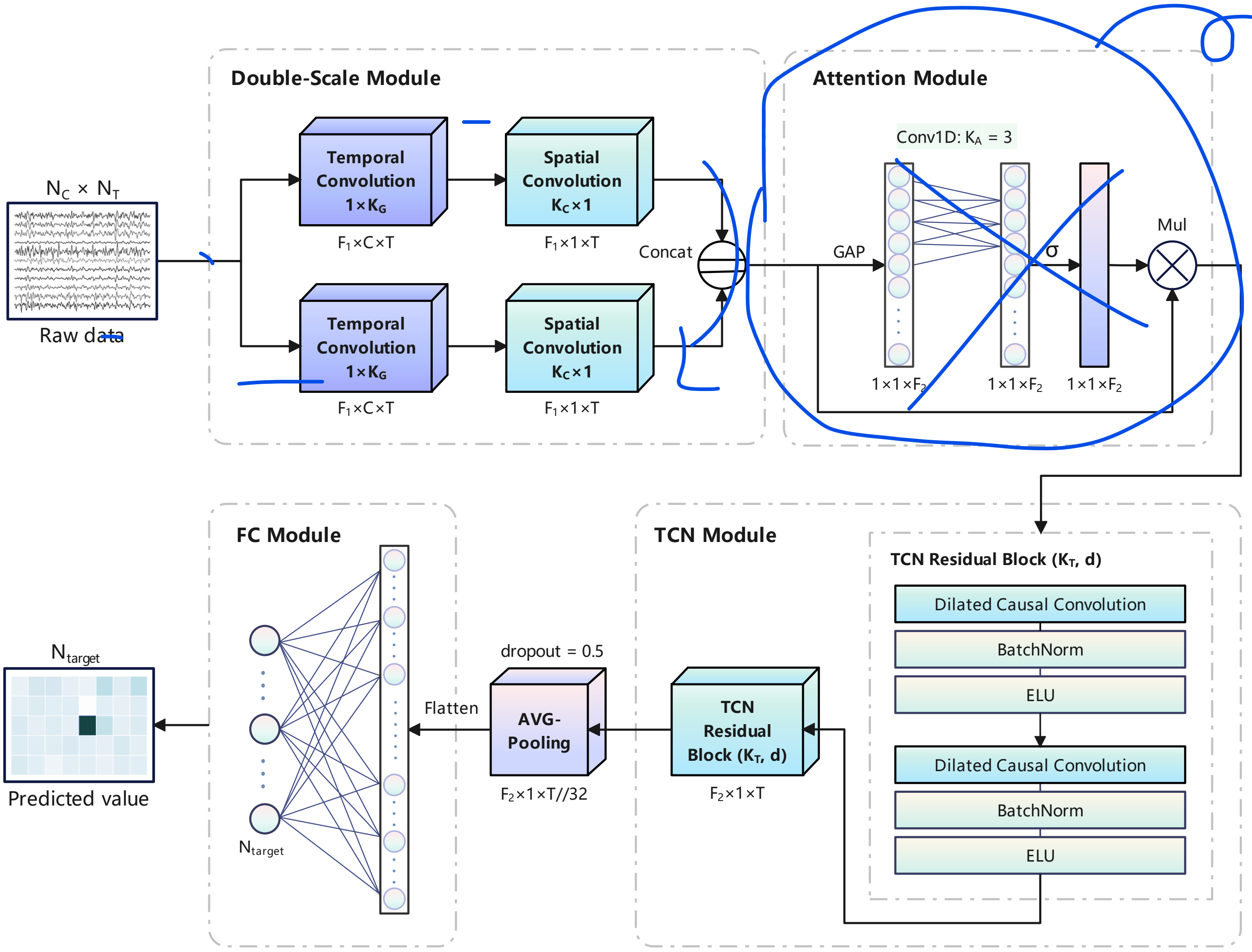


Fig. 1. Architecture of the proposed DATCNet model, which is an end-to-end model. The input of the proposed model is a signal matrix of dimensions $C\times T$, where $C$ and $T$ represent the number of selected channels and data length, respectively. The model output is the predicted values for 40 targets, with the index of the maximum prediction value selected as the final prediction result.

Classification Module. The double-scale module employs different sizes of kernel functions to extract both temporal and spatial information from EEG signals. The attention module is designed to extract filter weights. The TCN module is a specialized type of recurrent neural network [24], which is designed to learn the temporal features of a sequence. Finally, we use global average pooling to reduce the feature dimensionality and flatten all the features to the fully-connected layer for classification. The contribution of each module is further discussed in the ablation study section, the structure and parameters of each module are summarized in Fig. 1 and described in detail as follows:

1) Double-Scale Module: The double-scale module consists of two feature extraction modules with different kernel sizes. Each feature extraction module consists of a temporal convolution block and a spatial convolution block. One temporal convolution block uses $F1$ filters with kernel size (1, $K_G$), where $K_G$ is a larger kernel size of 125, while the other uses a smaller kernel size of $K_L = 51$ with the same number of filters. The kernel sizes were determined through cross-validation. The purpose of using different kernel sizes is to capture both global and local features in the EEG signals, which is further discussed in the next section. The output feature maps of each temporal convolution block have a size of $F_1 \times 1 \times T$, where $F_1$ is the number of filters and $T$ is the length of the input signal. To extract more temporal information and ensure the output feature maps have the same shape, we set the padding of the temporal convolution block to "same." The spatial convolution block is then used to extract spatial information from the signals, with a kernel size of $(C, 1)$, where $C$ is the number of EEG channels. Each block uses BatchNorm2D function for normalization and ELU function for activation. By learning the channel correlations through cross-channel information extraction, the temporal convolution block can capture both spatial and temporal features. Finally, the feature maps of the two modules are concatenated to obtain an output size of $F_2 \times 1 \times T$, where $F_2$ is twice the size of $F_1$.

2) Attention Module: The kernel size of the attention Module $K_A$ is calculated by Eq. (2). This module is known as the

efficient channel attention module [25]. The global average pooling (GAP) method is used to downsample the input feature map, and we apply the sigmoid function to the output of the 1D convolution layer to normalize the weights. $w_A$ represent the importance of each channel of the input feature map, which can be calculated by Eq. (3). Finally, we multiply the input feature map by the attention weights $w_A$ to obtain the attended feature map.

$$K_A = \psi(F_2) = \left|\frac{\log_2(F_2)}{\gamma} + \frac{b}{\gamma}\right|_{odd}, b = 1, \gamma = 2 \tag{2}$$

$$\omega_A = \sigma\left(\text{Conv}\,1D_K(GAP(X))\right) \tag{3}$$

3) TCN Module: In contrast to a recurrent neural network, the TCN module uses causal convolutions to prevent leakage of temporal information and uses dilated convolutions to capture long-term dependencies in the input sequence [24]. In this module, the number of residual blocks ($L$) is 2, which are used to increase network depth and the size of the receptive field [26]. The receptive field size of the TCN module is defined by Eq. (4), where $K_T$ represents the kernel length of all convolutional layers and is set to 16. In our implementation, we modified the original TCN module proposed in [24] by replacing the activation function with ELU, replacing the normalization method with BatchNorm1D, and setting the dropout to 0.25. Additionally, the dialation size $d$ is set to 2. These modifications were found to improve the performance of the TCN module in our network architecture. According to Eq. (4), the receptive field size of our TCN module is 91.

$$S_{RF} = 1 + 2\left(K_T - 1\right)\left(2^L - 1\right) \tag{4}$$

4) Fully-connected Classification Module: We use global average pooling with a size of 8 to downsample the feature map. The resulting flattened features are then fed into a linear layer for final classification, with a dropout rate of 0.5 to prevent overfitting. The model produces predictions for 40 classification targets, and the target corresponding to the highest predicted value is selected as the final prediction, Eq. (5), where $y_{\text{pre}}$ refers to the predicted value of the model for 40 targets, and $\hat{y}$ refers to the final predicted label.

$$\hat{y} = \operatorname{argmax}\left(y_{\text{pre}}(j)\right), y_{\text{pre}} \in \mathbb{R}^{N\times 1} \tag{5}$$

### C. Training strategy

For the Benchmark dataset, which has sufficient calibration data, we propose a two-step training strategy to obtain the model while ensuring its generalization ability. Our aim was to have the model learn reference signal templates in the first stage and individual signal templates in the second stage. In the first stage, we used the leave-one-subject-out (LOSO) method to obtain a subject-independent model. The model obtained in this stage is based on a training-free method and can achieve cross-subject prediction of SSVEP signal classification results. In the second stage, we used the subject-independent model obtained in the first stage as the pre-trained model for subject-dependent calibration. The leave-one-block-out (LOBO) method was applied to fine-tune and obtain subject-dependent models for each of the 35 subjects. To intuitively test the model's generalization performance and verify its ability for quick calibration, we used the first four blocks of EEG data from the Benchmark dataset as the training set and the last two blocks as the test set to evaluate the model's performance on this dataset. Specifically, we used the data from all subjects in the first four blocks to train a subject-independent model in the first stage using the LOSO method, and used the LOBO method in the second stage to train subject-dependent models for the 35 subjects, which were then tested and evaluated on the test set.

For the Beta dataset, which has limited calibration data, we employed the pre-trained base model from the first stage of training on the Benchmark dataset as a template for transfer learning, and trained subject-dependent models directly in the second stage. The aim of this process was to evaluate the rapid calibration and online generation capabilities of the subject-dependent model. For each subject, we used one or two blocks of data as training data and tested the model's accuracy on the remaining two blocks. Specifically, there was only one or two calibration data per target available for each individual.

We constructed the model using the PyTorch framework and accelerated training using a GTX 3060Ti GPU. The entire network was trained by minimizing the cross-entropy loss function and optimizing with the Adam optimizer. The training batch size was set to 64, and the maximum number of training epochs was set to 100. During the subject-independent training stage, the learning rate was set to 0.001, and the CosineAnnealingLR method [27] was used to dynamically adjust the learning rate. In the subject-dependent training stage, the learning rate was adjusted to 0.0005. Additionally, we utilized an early stopping algorithm to prevent overfitting and reduce training cost, with the early stopping step set to 10.

### D. Evaluation metrics

For the performance evaluation, three advanced deep learning models in the EEG decoding field were used for comparison, and the training strategy was the same as described in the previous section. In addition, we compared and analyzed the proposed online generative model training method with other advanced SSVEP supervised algorithms and training-free SOTA algorithms to comprehensively assess the performance of the model in terms of both accuracy and efficiency. The comparison methods are as follows:

1) Deep-learning methods: CompactCNN [14], EEGTCN [26], GuneyNet [8]
2) Supervised methods: mECCA [9], eTRCA [10], eSSCOR [11], TDCA [13]
3) Training-free method: FBCCA [7]

Basically, we use the average accuracy, information transfer rate (ITR) [21, 28], macro-averaged F1-score, and costing time to evaluate the performance of each algorithm. Additionally, for the deep-learning methods, to measure their time and space complexity, we also calculated the multiply-accumulate operations (MACs) and parameters (# Params) of each model using the computation method proposed in [29].

TABLE I: Performance results of the advanced deep learning models on the Benchmark test dataset with a time window of 1.0 s.

| Model | Subject-independent Accuracy (%) | Subject-dependent Accuracy (%) | Macro Fl-score | MACs | Params |
|---|---|---|---|---|---|
| DATCNet (proposed) | ~~68.04±20.90~~ | ~~87.57±15.10~~ | ~~0.8754~~ | 22.58M | 0.0558M |
| CompactCNN | 63.57±18.93 ** | 74.82±18.68 *** | 0.7477 | 68.42M | 0.0639M |
| EEGTCN | 48.43±19.65 *** | 65.96±20.23 *** | 0.6579 | 1.10M | 0.0140M |
| GuneyNet | 69.99±17.22 | 88.50±13.04 | 0.8774 | 22.54M | 0.7745M |

The paired t-test was used to compare the performance of DATCNet with that of other models.
Statistical significance is indicated by asterisks: * $p_{corrected} < 0.05$, ** for $p_{corrected} < 0.01$, and *** for $p_{corrected} < 0.001$.

## III. Results

### A. Classification results of the Benchmark dataset

On the Benchmark test dataset (Blocks 5 and 6), we evaluated our proposed model and the other advanced deep-learning models using a time window of 0.5 to 1.2 s. FBCCA [7], which is a SOTA training-free algorithm for SSVEP, was used as a baseline algorithm for comparison. The CCA used in this experiment had five harmonics ($N_h$). The average classification accuracies and ITRs for each method are depicted in Fig. 2 (a, b). According to the result of one-way repeated measures ANOVA, there was a significant difference among models for all data lengths in terms of accuracies and ITRs ($p < 0.001$). To further evaluate the performance of each model under the two-step training strategy, we calculated multiple evaluation metrics for each model in a 1-s time window (Table I). For the subject-independent model obtained in the first training stage, our proposed model achieved an average accuracy of 68.04%, which is significantly higher than that of the CompactCNN model (paired t-test, $p < 0.01$) and the EEGTCN model (paired t-test, $p < 0.001$). For the subject-dependent model obtained in the second training stage, the DATCNet model achieved an average accuracy of 87.57%, which is significantly higher than those of the CompactCNN model (paired t-test, p ¡ 0.001) and the EEGTCN model (paired t-test, $p < 0.001$). The SOTA model GuneyNet performed slightly better than the DATCNet model in both stages, but the difference was not significant (paired t-test, $p > 0.1$).

In addition, the time and space complexity of the proposed and comparison models were evaluated (Table I). For time complexity, the MACs of DATCNet and GuneyNet are similar, at 22.58M and 22.54M, respectively, which are both lower than the MAC of CompactCNN (68.42M) but higher than the MAC of EEGTCN (1.10M). For space complexity, the parameter size of DATCNet for each input signal is 0.0558M, which is slightly smaller than that of the CompactCNN model (0.0639M), higher than that of the EEGTCN model (0.0140M), and significantly smaller than that of GuneyNet (0.7745M). Although our proposed model has a slightly lower classification accuracy than GuneyNet, the model size is significantly smaller, being only about 1/14 of the size of GuneyNet. This implies that under the same conditions, storing a subject-dependent model of GuneyNet would be equivalent to storing approximately 14 subject-dependent models generated by our proposed model.

### B. Classification results for the Beta dataset

Fig. 3 presents the subject-dependent classification results of the first 15 subjects from the Beta dataset after one-shot and two-shot calibration. In contrast to our experiments for the Benchmark dataset, we directly conducted the second phase of subject-dependent model training on the Beta dataset and compared the results with other advanced supervised methods in a 1-s time window, as presented in Table II. According to the one-way repeated measures ANOVA, there is a significant difference among these supervised methods in terms of accuracy ($p < 0.001$). All supervised methods show a significant improvement in accuracy as the number of calibration data increases ($p < 0.001$). For each subject-dependent model, DATCNet achieved an accuracy of 63.17%, an average F1-score of 0.63, and an average ITR of 104.23 bits/min on the test set with only one set of calibration data, which was higher than FBCCA (59.84%, paired t-test: $p > 0.1$) and the other trained algorithms. The accuracies of the other trained algorithms were as follows: (mECCA: 63.08% (paired t-test, $p > 0.1$); ensemble eTRCA: 9.92% (paired t-test, $p < 0.001$); eSSCOR: 8.33% (paired t-test, $p < 0.001$); TDCA: 12.33% (paired t-test, $p < 0.001$). With two sets of calibration data, DATCNet achieved an accuracy of 70.75% and an average ITR of 123.27 bits/min on the testing set. Moreover, the accuracies of other algorithms were as follows: mECCA: 70.67% (paired t-test, $p > 0.1$); eTRCA: 70.08% (paired t-test, $p < 0.001$); eSSCOR: 70.25% (paired t-test, $p < 0.001$); and TDCA: 74.08% (paired t-test, $p > 0.1$).

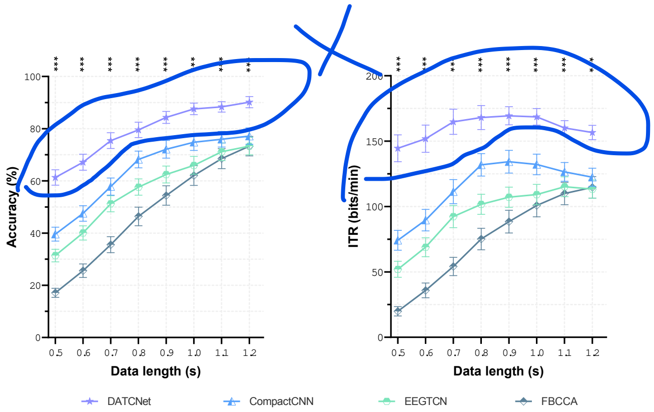


Fig. 2. Average classification results of the Benchmark test dataset in the time window of 0.5 s to 1.2 s. (a) Average accuracies across 35 subjects; (b) average ITRs across 35 subjects. The error bars indicate the standard error of the mean. One-way repeated measures ANOVA was conducted for all data lengths to test the differences in accuracies and ITRs. Statistical significance is indicated by asterisks: * for $p_{corrected} < 0.05$, ** for $p_{corrected} < 0.01$, and *** for $p_{corrected} < 0.001$.

TABLE II: Results for the Beta dataset.

| Method | Number of calibration trials | | | | | | Testing time (s/block) |
|---|---|---|---|---|---|---|---|
| | 1 | | | 2 | | | |
| | Accuracy (%) | Fl-score | Training time (s) | Accuracy (%) | Fl-score | Training time (s) | |
| DATCNet (proposed) | ~~63.17±23.56~~ | ~~0.6311~~ | ~~1.7930~~ | ~~70.73±22.90~~ | ~~0.7071~~ | ~~2.1375~~ | 0.0020 |
| MeCCA | 63.08±24.58 | 0.6290 | 0.0561 | 70.67±23.39 | 0.7054 | 0.0578 | 2.2635 |
| eTRCA | 9.92±5.31 *** | 0.0977 | 0.0274 | 70.08±23.06 | 0.7004 | 0.0380 | 0.3345 |
| eSSCOR | 8.33±6.84 *** | 0.0829 | 0.0275 | 70.25±23.43 | 0.7018 | 0.0305 | 0.3513 |
| MsetCCA | 5.50±3.40 *** | 0.0538 | 0.0701 | 47.42±26.96 * | 0.4727 | 0.2440 | 0.6008 |
| TDCA | 12.33±10.44 *** | 0.1255 | 0.0774 | 74.08±21.12 | 0.7422 | 0.0994 | 0.6793 |
| FBCCA | 59.84±19.69 | 0.6058 | NaN | 59.84±19.69 * | 0.6058 | NaN | 1.8150 |

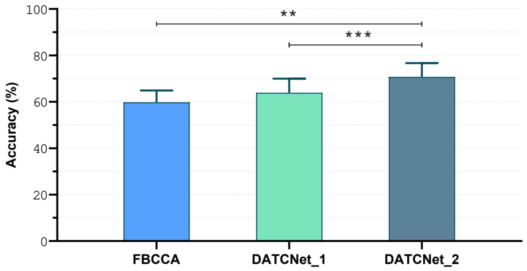


Fig. 3. The mean classification accuracy with standard error of the mean (SEM) of Beta testing dataset, comparing the DATCNet model after using one-shot or two-shot calibrations to the SOTA algorithm FBCCA. Significant differences were determined using paired t-test, with ** indicating $p_{corrected} < 0.01$, *** indicating $p_{corrected} < 0.001$.

The training and test time of various algorithms were calculated and are listed in Table II. Although the deep-learning method did not have an advantage in terms of training time, under our proposed subject-dependent calibration mode, the model only needs to be trained for about 2 s for each participant (1.7930 s when one set of calibration data is used and 2.1375 s when two sets of calibration data are used). With respect to test time, the mECCA algorithm, which has similar accuracy under one- and two-shot calibration conditions, had a longer testing time (2.2635 s/block) because of the complexity of its computation. In contrast, our proposed deep learning model can complete the classification task of 40 targets in just 0.002 s, which is almost negligible and much faster than all other algorithms.

### C. Ablation study and visualization

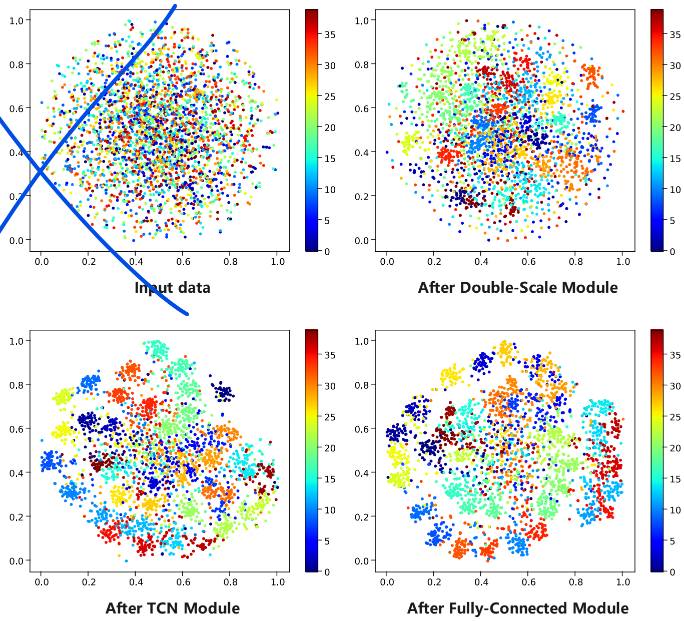
To further evaluate the contributions of the proposed DATCNet model structure, an ablation study was performed on the Benchmark dataset and the results are presented in Table III. Each of the four important components of the model was removed separately to obtain five models for comparison, i.e., All (with all modules), No-Atten (without the attention module), No-TCN (without the TCN module), No-Scale $K_G$ (without the feature extraction module at scale $K_G$), and No-Scale $K_L$ (without the feature extraction module at scale $K_L$). The contribution of each component was compared by evaluating the classification accuracy of the individual models trained in the second stage using a one-shot calibration trial on the test set. For different calibration trial numbers, the models with a specific component removed were compared with the complete model using paired t-tests.

For the subject-independent model, the classification accuracy of the All model was 68.04%. The No-Atten model showed a slightly higher accuracy than the complete model, with an accuracy of 68.32%, but the difference was not significant ($p > 0.1$). The No-TCN, No-Scale $K_G$, and No-Scale $K_L$ models had accuracies of 63.96%, 59.71%, and 64.36%, respectively, which were all significantly lower than the All model ($p < 0.001$). When only one calibration trial was used, the classification accuracy of the All model was 81.39%, which is an increase of 13.35% when compared with the results of the initial subject-independent model. The No-Atten model had an accuracy of 79.39%, and the All model outperformed it significantly ($p < 0.05$). Similarly, the No-TCN model had an accuracy of 78.39% ($p < 0.01$), the No-Scale $K_G$ model had an accuracy of 75.93% ($p < 0.001$), and the No-Scale $K_L$ model had an accuracy of 78.39% ($p < 0.01$). After completing the second stage of training using LOBO, the classification accuracy of the All model was 87.57%, whereas the No-Atten model had an accuracy of 85.82% ($p < 0.05$), the No-TCN model had an accuracy of 84.18% ($p < 0.001$), the No-Scale $K_G$ model had an accuracy of 83.00% ($p < 0.001$), and the No-Scale $K_L$ model had an accuracy of 85.14% ($p < 0.01$). In general, the classification accuracy increased as the number of calibration trials increased, and the complete DATCNet model achieved the optimal level of performance.

Fig. 4 visualizes the proposed models to illustrate the specific classification process. In detail, we used the subject-independent model trained in the first stage and applied it to 2800 test data (35 subjects × 2 test blocks × 40 trials) as input. The feature maps of each module were reduced using a two-dimensional t-stochastic neighborhood embedding (t-SNE) [30] approach (Fig. 4. A–D). It can be observed that as the features move through the model hierarchy, the signals gradually cluster together to form 40 target clusters, and the cluster centers gradually disperse. Fig. 5 presents the confusion matrices of models obtained from the two-step training strategy. A confusion matrix is an $M \times M$ matrix, where $M$ represents the number of unique classes, the horizontal axis represents the stimulus frequency of the predicted target label, and the vertical axis represents the stimulus frequency of the actual label. From the confusion matrices before and after calibration, it can be seen that the false

TABLE III: RESULTS OF THE ABLATION STUDY.

| Method | Number of calibration trials | | | |
|---|---|---|---|---|
| | 0 | 1 | 2 | 3 |
| DATCNet | ~~68.04±20.90~~ | ~~81.39±15.84~~ | ~~85.21±14.10~~ | ~~87.57±13.10~~ |
| ~~No-Atten~~ | 68.32±20.46 | 79.39±17.94 * | 83.85±15.39 * | 85.82±13.84* |
| No-TCN | 63.96±21.62 *** | 78.39±17.71 ** | 80.75±17.18 *** | 84.18±14.79 *** |
| No-Globalscale | 59.71±22.20 *** | 75.93±18.40 *** | 78.75±18.00 *** | 83.00±15.51 *** |
| No-Localscale | 64.36±21.53 *** | 78.39±17.94 ** | 81.79±17.65 ** | 85.14±14.53 ** |



Input data

After Double-Scale Module

After TCN Module

After Fully-Connected Module

Fig. 4. Two-dimensional t-SNE visualization of the outputs of DATCNet layers. (a) Input data of the Benchmark test dataset; (b) feature map after the double-scale module; (c) feature map after the TCN module; (d) feature map after the fully connected module. Dots of different colors represent the categories of real stimulus targets.

positive rate of the subject-dependent model is greatly reduced after calibration, and the misjudgment is mainly distributed in the ±0.6 Hz range of the true target. According to [9], this error pattern can be explained as the confusion that arises from the design of the JFPM method. We use Eq. (7) to generate the stimulus sequence, where $f_k$ and $\Phi_k$ are respectively the stimulus frequency and phase in Eq. (1). Fig. 6. (a) visualizes the waveforms of the stimulation signals of 11.8 Hz, 12.4 Hz, and 13.0 Hz in a 1-s time window, and a period of overlapping waveforms can be observed. Fig. 6. (b) illustrates the correlation coefficients between the 12.4 Hz stimuli and all stimuli of 8 Hz to 15.8 Hz, where the correlation coefficient is calculated using Eq. (8). According to Fig. 6. (b), the 12.4-Hz stimulation signal has the highest correlation with the stimulation signals of 11.8 Hz and 13.0 Hz (0.48 and 0.46, respectively), and the correlation is moderately positive, whereas the correlation with the two nearest neighboring stimulation signal is negative (12.2 Hz: −0.56, 12.6 Hz: −0.55). When using frequency phase modulation with an interval of $0.5\pi$, the correlation between stimuli with an interval of 0.6 Hz is maximized.

$$x_k(t) = \sin\left(2\pi f_k t + \phi_k\right), k \in C_A \tag{6}$$

$$r = \frac{\sum_{i=1}^{T}\left(x_i - \bar{x}\right)\left(y_i - \bar{y}\right)}{\sqrt{\sum_{i=1}^{T}\left(x_i - \bar{x}\right)^2 \sum_{i=1}^{T}\left(y_i - \bar{y}\right)^2}} \tag{7}$$

To further analyze the impact of the proposed double-scale feature extraction module, we compared the differences in feature selection between different scales of convolutional layers by plotting waveform features of the filter outputs at two different scales in both the time and frequency domains (Fig. 7). According to Fig. 7, the signals extracted by convolutional layers with different scales show clear differences in the waveform features among each result from the eight distinct kernels. After the features have been processed with an FFT, almost every filter can effectively capture the frequency of the stimulus target and its harmonics. Through temporal convolution blocks, signals are filtered into different frequency bands by convolution filters, and different convolution filters generate different feature mappings. For example, in kernel 5, the larger kernel mainly captures the second harmonic of the stimuli and its smaller kernel mainly captures the fundamental stimuli, whereas kernel 7 behaves in the opposite way. When compared with the smaller kernels ($K_L$, orange line), the larger kernels ($K_G$, blue line) are more sensitive to the second harmonic, with the amplitude being highest in that frequency band. The smaller kernels capture information in the lower amplitudes but capture the fundamental frequency more prominently and preserve more harmonic information.

## IV. DISCUSSION

In this section, we provide a thorough analysis of our proposed model, focusing on its characteristics and composition, that is based on the classification results obtained from two 40-class SSVEP public datasets. Then, we summarize the comparison with other advanced SSVEP classification algorithms to analyze the suitable application scenarios of the model. Finally, we discuss its potential directions for future research.

### A. Contribution of the modules

According to the results of the ablation experiments, the double-scale feature extraction module extracts more useful information from the signals, and the attention module is advantageous for the fast calibration of subject-dependent models. Studies have shown that the kernel size plays a critical role in the model design and training process [31, 32]. The

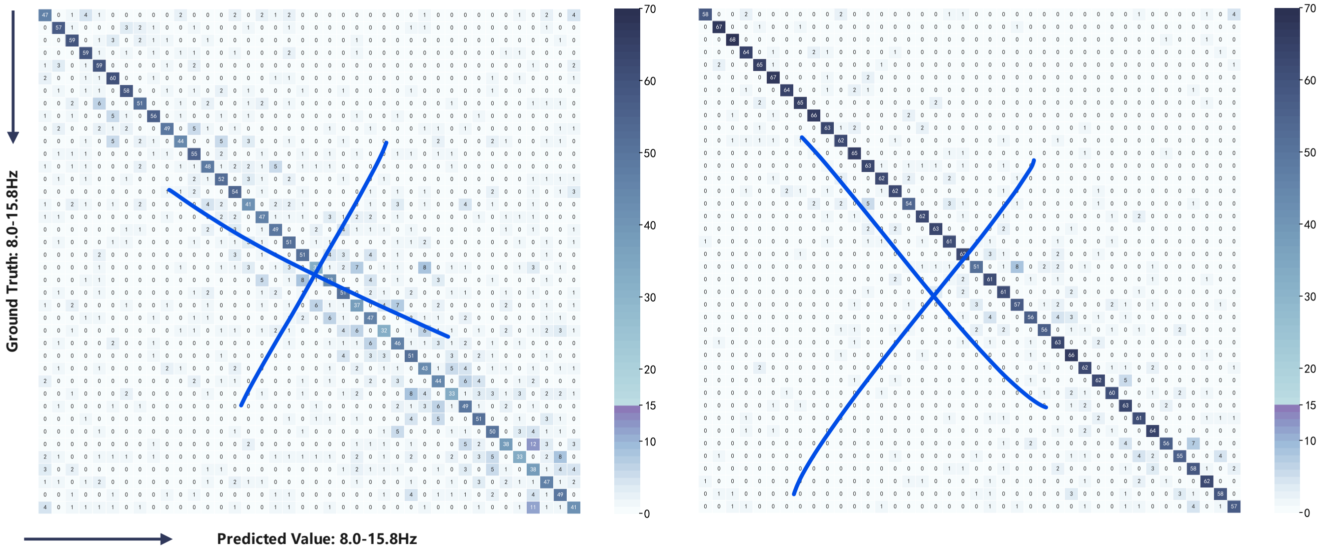


Fig. 5. Forty-target confusion matrices before and after calibration. (a) Confusion matrix of the subject-independent model; (b) confusion matrix of the subject-dependent model.

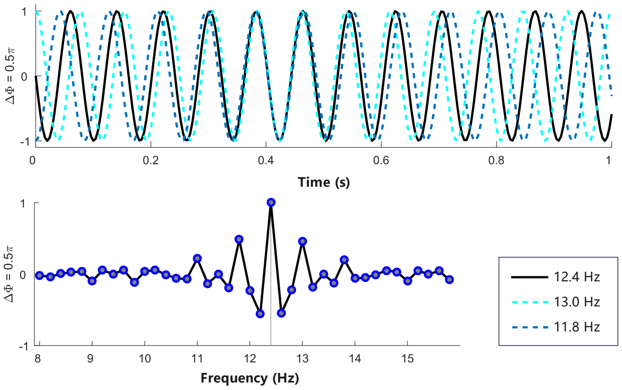


Fig. 6. Visualization of the stimulus sequence generated by the JFPM method of the Benchmark and Beta datasets. (a) Waveform of three stimulation signals with a frequency interval of 0.6 Hz and phase interval of $0.5\pi$ in the time domain (11.8 Hz: $1.5\pi$, 12.4 Hz: $1.0\pi$, and 13.0 Hz: $0.5\pi$). (b) Pearson correlation coefficients between the 12.4-Hz stimulation signal and all stimulation signals (8–15.8 Hz).

design of the double-scale convolution module is inspired by the Inception block [32], which can obtain filters with different receptive fields by employing different kernel sizes and increase the receptive field by increasing the model depth. This allows the model to achieve high classification performance with fewer filters to capture the global and local features of the signal. Using fewer filters reduces the spatial complexity of the model, and a smaller parameter count is advantageous for reducing RAM overhead [33]. Specifically, in the SOTA model, which uses 120 filters and a dropout of 0.95 in the final classifier [8], a large number of parameters are wasted and additional RAM overhead is incurred. In this study, we use different kernel sizes to enhance the features. In the comparison models, the FBEEGNet model [34] enhances information by generating ensemble models with parallel channels by band filtering, the CompactCNN model [14] increases the number of features by increasing the number of filters and using large convolution kernels with the same signal length, the GuneyNet model [8] additionally uses filters for cross-harmonic feature extraction, and the Conv-CA model [35] utilizes the idea of CCA to train template signals and subject signals in parallel, with a correlation analysis layer to calculate the correlation coefficient of tensors. Although they all show good performance in classification accuracy, they all exhibit computational and storage resource consumption issues and rely on powerful GPU support.

### B. Necessity of calibration data

Calibration data are essential for improving the complex classification accuracy of subject-dependent models. From Tables II and III, it can be seen that the accuracy of the model increases as the calibration data increases. Subject-specific calibration is quick and efficient, with just one-shot calibration improving the accuracy by more than 13%. In new application scenarios, where there is a lack of sufficient subject data, a model that can be quickly calibrated to adapt to new environments has practical significance for online systems [36]. Although collecting calibration data takes a long time, our subject-dependent model can be generated in about 2 s, and this can be completed during the calibration data collection process, enabling the model to be immediately put into use in online systems.

### C. Comparison with advanced algorithms

Compared with the latest SOTA model GuneyNet, although our model's classification accuracy is slightly lower, its complexity has significantly decreased. In addition, for the training method proposed in [8], the training time for a subject-dependent model is about 30 minutes with powerful GPU

用小波

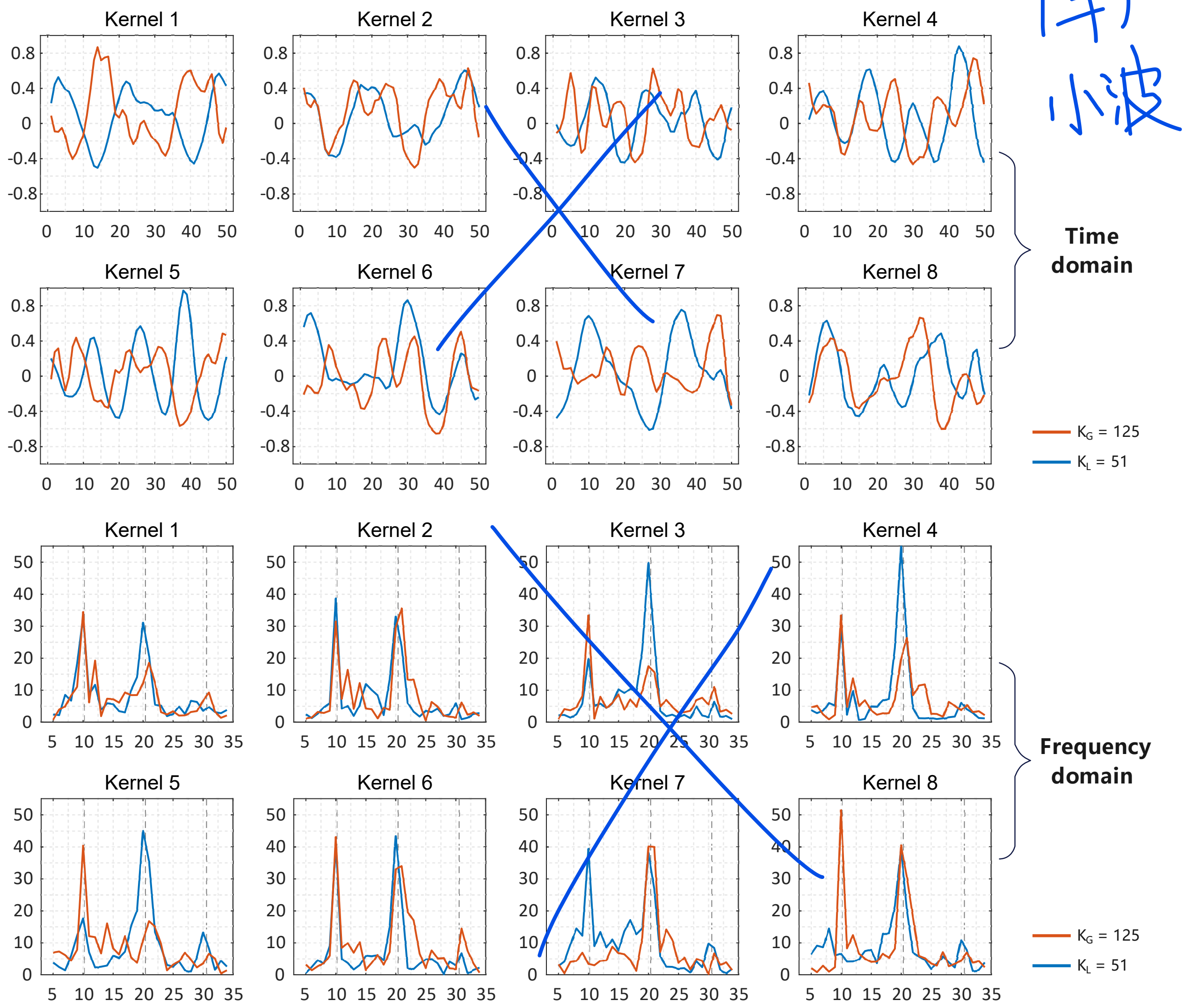


Fig. 7. Visualization of DATCNet temporal kernels. Input data were obtained from Subject 3's EEG signal for the stimuli of the letter "K" (10.2 Hz) in the Benchmark dataset, and the waveform representing the averaged kernel output, which comes from the double-scale module's temporal convolution blocks. The blue line represents $K_G$ and the orange line represents $K_L$. (a) Each filter's waveform characteristics of the feature maps from the two scales of temporal convolution blocks in the time domain. The horizontal axis represents the time point and the vertical axis represents the amplitude. (b) Each filter's waveform characteristics of the feature maps from two scales of temporal convolution blocks in the frequency domain. The horizontal axis represents the frequency and the vertical axis represents the magnitude.

support (Tesla V100). Although there have been recent studies proposing the training-free prediction of ensemble models based on other subjects [19], the premise is that a large number of subject data are available from the same scenario, which is difficult for real-world online BCI systems in new scenarios [37]. Our proposed training strategy, using the results of the first training stage, can greatly reduce the model's training time and resources and has the feasibility of deployment in online systems. Compared with other supervised SSVEP algorithms, our deep learning model has a competitive advantage in scenarios with a lack of calibration data, and its accuracy and testing speed show outstanding performance among various algorithms.

## V. CONCLUSION

In this study, we proposed a temporal convolutional neural network-based model (DATCNet) for SSVEP classification and validated it on two publicly available datasets. The experimental results demonstrate that the proposed DATCNet model achieves outstanding performance on multitasking SSVEP classification, with a high classification accuracy, lower training cost, and higher training speed than other advanced deep-learning methods. The proposed training strategy can solve the dilemma of a lack of subjects and calibration data in new application scenarios, and suggest the potential for building an online model generation system for more SSVEP-BCI applications in the real world.

## References

[1] J. J. Daly and J. E. Huggins, "Brain-computer interface: current and emerging rehabilitation applications," *Arch. Phys. Med. Rehabil.*, vol. 96 3 Suppl, pp. S1-7, 2015.

[2] N. Shi, L. Wang, Y. Chen, X. Yan, C. Yang, Y. Wang, and X. Gao, "Steady-state visual evoked potential (SSVEP)-based brain–computer interface (BCI) of Chinese speller for a patient with amyotrophic lateral sclerosis: A case report," *J. Neurorestoratology*, vol. 8, no. 1, pp. 40–52, 2020.

[3] M. Nakanishi, Y. Wang, W. Yu-Te, and T.-P. Jung, "A comparison study of canonical correlation analysis based methods for detecting steady-state visual evoked potentials," *PloS One*, vol. 10, no. 10, p. e0140703, 2015.

[4] X. Chen, Y. Wang, M. Nakanishi, T.-P. Jung, and X. Gao, "Hybrid frequency and phase coding for a high-speed SSVEP-based BCI speller," presented at the 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Chicago, IL, USA, Aug. 26–30, 2014, pp. 3993–3996.

[5] M. Nunez, R. Srinivasan, and J. Vandekerckhove, "Individual differences in attention influence perceptual decision making," *Front. Psychol.*, vol. 8, p. 18, Feb. 2015.

[6] Z. Lin, C. Zhang, W. Wu, and X. Gao, "Frequency recognition based on canonical correlation analysis for SSVEP-based BCIs," *IEEE Trans. Biomed. Eng.*, vol. 54, no. 6, pp. 1172–1176, Jun. 2007.

[7] X. Chen, Y. Wang, S. Gao, T.-P. Jung, and X. Gao, "Filter bank canonical correlation analysis for implementing a high-speed SSVEP-based brain-computer interface," *J. Neural Eng.*, vol. 12, no. 4, pp. 046008–046008, 2015.

[8] O. B. Guney, M. Oblokulov, and H. Ozkan, "A deep neural network for SSVEP-based brain-computer interfaces," *IEEE Trans. Biomed. Eng.*, vol. 69, no. 2, pp. 932–944, 2022.

[9] X. Chen, Y. Wang, M. Nakanishi, X. Gao, T.-P. Jung, and S. Gao, "High-speed spelling with a noninvasive brain–computer interface," *Proc. Natl. Acad. Sci.*, vol. 112, no. 44, pp. E6058–E6067, 2015.

[10] M. Nakanishi, Y. Wang, X. Chen, Y.-T. Wang, X. Gao, and T.-P. Jung, "Enhancing detection of SSVEPs for a high-speed brain speller using task-related component analysis," *IEEE Trans. Biomed. Eng.*, vol. 65, no. 1, pp. 104–112, Jan. 2018.

[11] G. R. Kiran Kumar and M. Ramasubba Reddy, "Designing a sum of squared correlations framework for enhancing SSVEP-based BCIs," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 10, pp. 2044–2050, 2019.

[12] C. M. Wong, B. Wang, Z. Wang, K. F. Lao, A. Rosa, and F. Wan, "Spatial filtering in SSVEP-based BCIs: unified framework and new improvements," *IEEE Trans. Biomed. Eng.*, vol. 67, no. 11, pp. 3057–3072, 2020.

[13] B. Liu, X. Chen, N. Shi, Y. Wang, S. Gao, and X. Gao, "Improving the performance of individually calibrated SSVEP-BCI by task-discriminant component analysis," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. PP, pp. 1–1, Sep. 202.

[14] N. Waytowich, V. J. Lawhern, J. O. Garcia, J. Cummings, J. Faller, P. Sajda, and J. M. Vettel, "Compact convolutional neural networks for classification of asynchronous steady-state visual evoked potentials," *J. Neural Eng.*, vol. 15, no. 6, p. 066031, Oct. 2018.

[15] J. J. Podmore, T. P. Breckon, N. K. N. Aznan, and J. D. Connolly, "On the relative contribution of deep convolutional neural networks for SSVEP-based bio-signal decoding in BCI speller applications," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 4, pp. 611–618, 2019.

[16] H. Cecotti, "A time–frequency convolutional neural network for the offline classification of steady-state visual evoked potential responses," *Pattern Recogn. Lett.*, vol. 32, no. 8, pp. 1145–1153, Jun. 2011, doi: 10.1016/j.patrec.2011.02.022.

[17] A. Ravi, N. H. Beni, J. Manuel, and N. Jiang, "Comparing user-dependent and user-independent training of CNN for SSVEP BCI," *J. Neural Eng.*, vol. 17, no. 2, p. 026028, Apr. 2020.

[18] X. Li, W. Wei, S. Qiu, and H. He, "TFF-Former: Temporal-frequency fusion transformer for zero-training decoding of two BCI tasks," in *Proc. 30th ACM Int. Conf. Multimed.*, MM '22, NY, USA: Association for Computing Machinery, 2022, pp. 51–59.

[19] O. B. Guney and H. Ozkan, "Transfer learning of an ensemble of DNNs for SSVEP BCI spellers without user-specific training," *J. Neural Eng.*, vol. 20, p. 016013, 2022.

[20] B. Liu, X. Huang, Y. Wang, X. Chen, and X. Gao, "BETA: A large benchmark database toward SSVEP-BCI application," *Front. Neurosci.*, vol. 14, p. 627, 2020.

[21] Y. Wang, X. Chen, X. Gao, and S. Gao, "A benchmark dataset for SSVEP–based brain–computer interfaces," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 10, pp. 1746–1752, Oct. 2017.

[22] Y. Wang, X. Gao, B. Hong, C. Jia, and S. Gao, "Brain-computer interfaces based on visual evoked potentials," *IEEE Eng. Med. Biol. Mag.*, vol. 27, no. 5, pp. 64–71, 2008.

[23] Y. Pan, J. Chen, Y. Zhang, and Y. Zhang, "An efficient CNN-LSTM Network with spectral normalization and label smoothing technologies for SSVEP frequency recognition," *J. Neural Eng.*, vol. 19, Aug. 2022.

[24] S. Bai, J. Z. Kolter, and V. Koltun. (2018). An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv:1803.01271. [Online]. Available: http://arxiv.org/abs/1803.01271.

[25] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "ECA-Net: Efficient channel attention for deep convolutional neural networks," In *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11534–11542, .

[26] T. M. Ingolfsson, M. Hersche, X. Wang, N. Kobayashi, L. Cavigelli, and L. Benini, "EEG-TCNet: An accurate temporal convolutional network for embedded motor-imagery brain–machine interfaces," presented at the 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Toronto, ON, Canada, Oct. 11–14, 2020, pp. 2958–2965.

[27] I. Loshchilov and F. Hutter. (2017). SGDR: Stochastic gra-

dient descent with warm restarts. arXiv:1608.03983. [Online]. Available: https://doi.org/10.48550/arXiv.1608.03983.

[28] X. Lin, Z. Chen, K. Xu, and S. Zhang, "Development of a high-speed mental spelling system combining eye tracking and SSVEP-based BCI with high scalability," presented at 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Berlin, Germany, Jul. 23–27, 2019, pp. 6318–6322.

[29] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz. (2016). Pruning convolutional neural networks for resource efficient inference. arXiv: 1611.06440. [Online]. Available: https://doi.org/10.48550/arXiv.1611.06440.

[30] L. van der Maaten and G. Hinton, "Viualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

[31] F. Yu and V. Koltun. (2015). Multi-scale context aggregation by dilated convolutions. arXiv:1511.07122. 2015. [Online]. Available: https://doi.org/10.48550/arXiv.1511.07122.

[32] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, and A. Rabinovich, "Going deeper with convolutions," In *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1–9.

[33] G. Menghani, "Efficient deep learning: A survey on making deep learning models smaller, faster, and better," *ACM Comput. Surv.*, vol. 55, no. 12, pp. 1–37, 2021.

[34] H. Yao, K. Liu, X. Deng, X. Tang, and H. Yu, "FB-EEGNet: A fusion neural network across multi-stimulus for SSVEP target detection," *J. Neurosci. Methods*, vol. 379, p. 109674, 2022.

[35] Y. Li, J. Xiang, and T. Kesavadas, "Convolutional correlation analysis for enhancing the performance of SSVEP-based brain-computer interface," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 28, no. 12, pp. 2681–2690, 2020.

[36] R. Abiri, S. Borhani, E. Sellers, Y. Jiang, and X. Zhao, "A comprehensive review of EEG-based brain-computer interface paradigms," *J. Neural Eng.*, vol. 16, Nov. 2018.

[37] R. Zerafa, T. Camileeri, O. Falzon, and K. Camilleri, "To train or not to train? A survey on training of feature extraction methods for SSVEP-based BCIs," *J. Neural Eng.*, vol. 15, Jun. 2018.