

Optimized DNN Classification Framework based on Filter Bank Common Spatial Pattern (FBCSP) for Motor-imagery-based BCI

Zayyanu Shuaibu

Brain-Computer Interface Lab
Changchun University of Science and Technology

Li Qi

Brain-Computer Interface Lab
Changchun University of Science and Technology

ABSTRACT

The Brain-Computer Interface (BCI) is a kind of communication channel between the brain and the outside world that is not dependent on the nervous system. The recent improvement of deep neural network (DNN) in a classification task lead several researchers to apply DNN methods for BCI motor imagery (MI) classification. However, DNN based classification methods for MI classification either record low accuracy or take a long time to classify MI signal due to the high dimensional nature of the MI signal. This issue prevents the DNN based MI systems from being deployed in a real application. An ideal DNN classification framework for the MI classification was proposed in this study, which can be trained with better accuracy in a short period. To address the long DNN training time and the challenges of low classification accuracy, a new method was proposed that uses the FBCSP method to reduce the data dimension and increase class discrimination by extracting the best subject-specific features of the CSP from raw EEG data as input into the DNN. The DNN is designed with a few layers, and several DNN hyperparameters have been evaluated. A configuration that has performed better in terms of minimum training time and good classification accuracy has been selected. To investigate the accuracy of the proposed DNN methods called FBCSP-DNN; first, we use traditional approaches such as LDA, SVM, and KNN to create a baseline; second, we choose some DNN studies used in other studies that have been applied to the same dataset used in this study; third,

we look at the use of transfer learning, a network pre-training methodology for one data collection of some subjects and then fine-tuning for another to improve some subjects with low accuracy; finally, the results obtained using the proposed were compared to traditional methods and other competing DNN methods used in other studies. The performance of the proposed FBCSP-DNN method is evaluated using BCI competition IV dataset 2a. The results show that the proposed method (81.43%) is superior to the traditional classification methods (SVM: 72.37%; KNN: 61.06%; LDA: 72.07%). We also compared the proposed DNN method and other DNN methods proposed by other studies using the same dataset. The proposed method outperforms other studies in terms of accuracy of the MI classification for within-subjects and cross-subjects, which improved by 6% and 9% respectively. The FBCSP-DNN classification framework proposed in this paper has the advantages of short training time and high classification accuracy, which ensures reliability for the practical application of BCI-motor imagery systems.

Keywords

Brain-Computer Interface (BCI), Deep Neural Network (DNN), Motor Imagery (MI), Filter Bank Common Spatial Pattern (FBCSP), electroencephalography (EEG).

1 INTRODUCTION

Brain-Computer Interface (BCI) is a communication technology between the human and external devices, which measures and translates brain signals into a command for the interactive application [1]. BCI applications aim to address the inability of a patient with muscle impairment, paralyzed persons and myotrophic lateral sclerosis (ALS) patient, by providing them with an alternative way to communicate, which translates their mental state into a command for the control of external devices without any physical activity. Electroencephalography (EEG) is a non-invasive and is by far the most used to acquire the user's brain signal due to its less risk and easy setup for BCI, unlike invasive which requires brain surgery to measure the user's brain signal [2].

Motor imagery (MI) has been widely applied in non-invasive BCI as a communication approach [3]. When imagining moving some body parts such as hands, feet, and tongue it leads to a change in the power of EEG signals in some specific frequency bands [4]. The increase of EEG signal power in a given frequency band is called an Event-Related Synchronizations (ERS). In contrast, a decrease of EEG signal power is called an Event-Related Desynchronizations (ERD) [5]. As an example, the imagination of a left-hand movement leads to a contralateral ERD in the motor cortex (i.e., in the right motor cortex for left-hand movement) in the μ (about 8–12) and β (about 16–24) bands during movement imagination, and to an ERS in the β band just after the movement imagination ending. Recently [6] proposed a novel MI pattern classification based on the movement direction of both hands. They used a common spatial pattern (CSP) method for feature extraction and support vector machine (SVM) to discriminate between the hands' movement direction. [7] proposed a method to improve the convolutional neural network (CNN) models by fusing CNNs with different characteristics and architectures to improve EEG MI classification accuracy. However, these approaches either record low accuracy or take a long time to develop the motor imagery System.

Previous researches used traditional classifiers and feature extraction methods to improve the motor-imagery-based BCI system. Methods such as Principle Component Analysis (PCA), Independent Component Analysis (ICA), autoregression, and other feature extraction methods [7][8][10][11] were used to lower the data dimension and extract features. Classifiers can easily use this lower representation of the data. The classifiers used were Gaussian classifiers [12], logistic regression [13], regularized linear discriminant analysis (RLDA) [14][15], support vector machines (SVM) [16] and linear discriminant analysis (LDA) [17].

Recently, Deep Neural Network (DNN) has improved the ability to classify images, recognize speech, identify objects, and described contents. This improvement led several researchers to apply DNN methods for motor imagery classification [11], [18]–[24]. Restricted Boltzmann machine (RBM), convolutional layer, and autoencoders are the basic building blocks to construct deep neural networks (DNN). RBMs, convolutional layer, and autoencoders can be stacked and train in a layer-by-layer manner to form DBN (Deep Belief Network), CNN (Convolutional Neural Network), and stacked autoencoders (SAE) respectively which learn to extract a deep hierarchical representation of the training data. These DNN methods could be a solution for motor imagery classification. However, DNN networks have a large number of parameters. Consequently, it requires a large number of training data and the user cannot perform several long pieces of training before actually using a BCI application. It is difficult to gather many motor imagery data for DNN training [25]. Hence, there is less utilization of DNN in motor imagery classification. However, the Transfer learning method can enable to reduce long training by pre-training DNN models on large BCI data from other multiple subjects for the target BCI user or other periodic training for a known user [26]. A DNN will be able to learn from all subjects and use the learned data for new subjects without being retrained or by training on small data from the new data. This method could increase motor imagery-based application for DNN models.

Generally, The DNNs can be applied directly to the time-domain feature and frequency-domain feature of the raw EEG data. For example, A DBN method has been used in [27] on the time-domain feature of the raw EEG data signals for the classification of left and right-hand motor imagery based on a certain single channel. A weak classifier was trained by DBN, then copied the idea of the Ada-boost algorithm to joined the trained weak classifiers as a more powerful one. Several RBMs were stacked by putting the hidden layer of the bottom layer RBM as the visible layer of the next RBM, and the Contrastive Divergence (CD) algorithm was also used to train multilayered DBN. The performance of this DBN method was observed using different combinations of hidden units and hidden layers on multiple subjects. The new results showed that the method performs better with eight hidden layers, an improvement of 4-6% in certain cases in comparison with SVM was reported. In another work [28], the frequency-domain features were computed via fast Fourier transform (FFT) and wavelet packet decomposition (WPD). The obtained frequency domain of motor imagery signal was then fed into the DNN for training. Three RBMs were stacked together with an output layer to perform a classification task. Only band-pass filtering and data normalization were applied to the raw motor imagery data for preprocessing. The result shows that converting the motor imagery signal from the time domain to the frequency domain led to a significant increase in performance.

However, training the above DNNs on raw EEG data results in long DNN training and testing time due to the high dimensionality of the training data and a large number of DNN parameters [28], which is not feasible for real application. In BCI, it is desired to reduce the time necessary to record sufficient data, train, and test a model. The model has to be trained and tested in a few minutes or seconds to enable the practical application. Thus, to design an ideal DNN framework that can be used for a real application with good performance, there is a need to preprocess the raw EEG data

to lower the dimension and extract relevant features to feed the DNN. To address long DNN training and testing time, the DNN should be carefully designed with few parameters while keeping good accuracy.

Common Spatial Pattern (CSP) is by far the most used feature extraction method in the current Motor-imagery-based systems due to the ERD nature of the motor-imagery signal. CSP finds spatial filters that maximize the distance between multiple classes after which CSP features were obtained as the representation of the data. Some studies showed that these CSP features were inputted into linear classifiers such as SVM to yield good accuracy [29].

Some methods were proposed to improve the performance of the conventional CSP such as Common Spatial-spectral Pattern (CSSP) which optimizes a simple filter that employed one time-delayed sample with the CSP algorithm [30]; Common Sparse Spatial-spectral Pattern (CSSSP) was also used to improve the CSSP methods, which find the spectral pattern of all the channels instead of finding a spectral pattern for each channel in CSSP [31]; and Sub-band Common Spatial Pattern (SBCSP) filtered motor imagery signal at multiple bands and features are extracted from each band [32]. However, not all bands contain relevant information, using features from all bands decrease the performance of the motor imagery system. Due to the nature of motor imagery which is subject-specific, FBCSP has often used to select the relevant features from the bands [33]. To the best of our knowledge, no study used the FBCSP method combined with DNN for motor imagery classification.

In this study, a novel DNN framework was proposed using FBCSP for improving the MI-based BCI system. An FBCSP method was used to lower the data dimension and increase the discrimination between classes by extracting the best subject-specific CSP features from the raw EEG data as an input into the DNN. The DNN also trains on the CSP feature to extract high-level features for higher classification accuracy. To address the long DNN training time, the DNN was designed with a few layers, and several DNN hyperparameters were evaluated, and a configuration that performed better in terms of minimum training time was selected.

2 METHODS AND DATA DESCRIPTION

2.1 Dataset Description

The data set [34], [35] consisted of four different motor imaging tasks, namely the movement of the left hand (Class 1), the right hand (Class 2), the two legs (Class 3) and the tongue (Class 4). Two sessions on different days have been recorded for each subject. Each session consists of six runs which are separated by short breaks. A run consists of 48 trials (12 for each of the four possible classes) resulting in a total of 288 trials per session. The subjects were sitting in a comfortable chair in front of a computer screen. At the beginning of the trial ($t = 0$ s), a cross was shown on the black screen. After two seconds ($t = 2$ s), a sign in the form of an arrow pointing either to the left, right, down or up (corresponding to one of the four classes left hand, right hand, foot or tongue) appeared and remained on the screen for 1.25 s. This prompted the subjects to perform the desired task of motor imagery. No feedback has been provided. The subjects were asked to perform the task of motor imagery until the cross-fixing disappeared from the screen at $t = 6$ s. There followed a short break where the screen was black again.

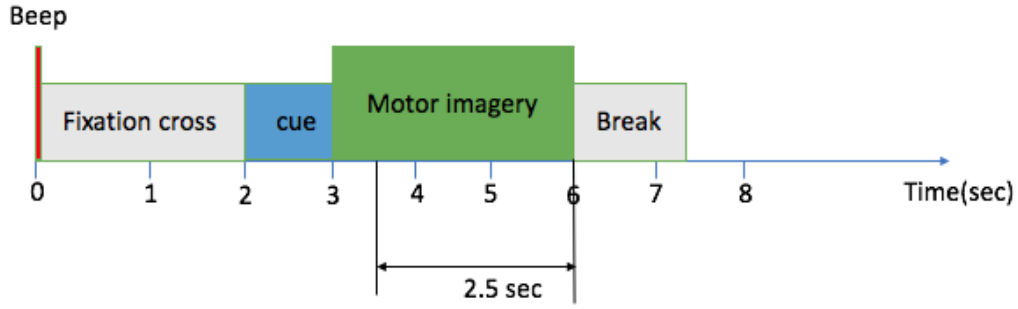


Figure 1: Timing scheme of the paradigm

Twenty-two electrodes were used to acquire each motor imagery data. They were recorded using bipolar recording with a sampling frequency of 250 Hz. Filter was used to band-pass filter between 0.5 Hz and 100 Hz, and a notch filter at 50 Hz. As shown in Figure 1, we extracted the EEG segment of 2.5-seconds lengths from 3.5-6.0-seconds of each trial MI EEG recording. EEG recordings are sampled as 250 samples per second (Each EEG recording contains 626 samples)

2.2 Filter Bank CSP (FBCSP)

Filter Bank CSP (FBCSP) was applied to the raw EEG data for every 2.5 seconds long trial. In the case of the 250 Hz signal, this is corresponding to 626 samples. FBCSP filter EEG in multiple frequency bands (4-8, 6-10, 12-14... 26-30) before the computation of CSP, this used to select the most discriminative subject-specific features from the best band (see Figure 2) FBCSP was indeed the algorithm used in the winning-entries of all EEG data sets from the last BCI competition[2]. Moreover, For each frequency band, spatial filters are optimized using the CSP algorithm; this selects both the best spectral and spatial filters since each feature

corresponds to a single frequency band and CSP spatial filter[36]. CSP uses the spatial filters w which extremize the following function:

$$J_{csp}(w) = \frac{wX_iX_i^Tw^T}{wX_iX_i^Tw^T} = \frac{wC_iw^T}{wC_iw^T} \quad (1)$$

Where T denotes transpose, X_i is the training band-pass filtered signal matrix for class i . $J_{csp}(w)$ is a Rayleigh quotient. Therefore, Generalized Eigen Value Decomposition (GEVD) can solve extremizing it. The spatial filters w that maximizes or minimizes $J_{csp}(w)$ is thus the eigenvectors corresponding to the largest and lowest eigenvalues, respectively, of the GEVD of matrices $C1$ and $C2$. Typically, 44 filters (i.e., 22 pairs), corresponding to the 22 largest and 22 lowest eigenvalues are used. Once these filters obtained, a CSP feature f is described as in equation ((2). Each trial data is spatially filtered using equation (3), WCSP is the spatial filter, and Z_m is the spatially projected data.

$$f = \log(wXX^Tw^T) = \log(wCw^T) = \log(\text{var}(wX)) \quad (2)$$

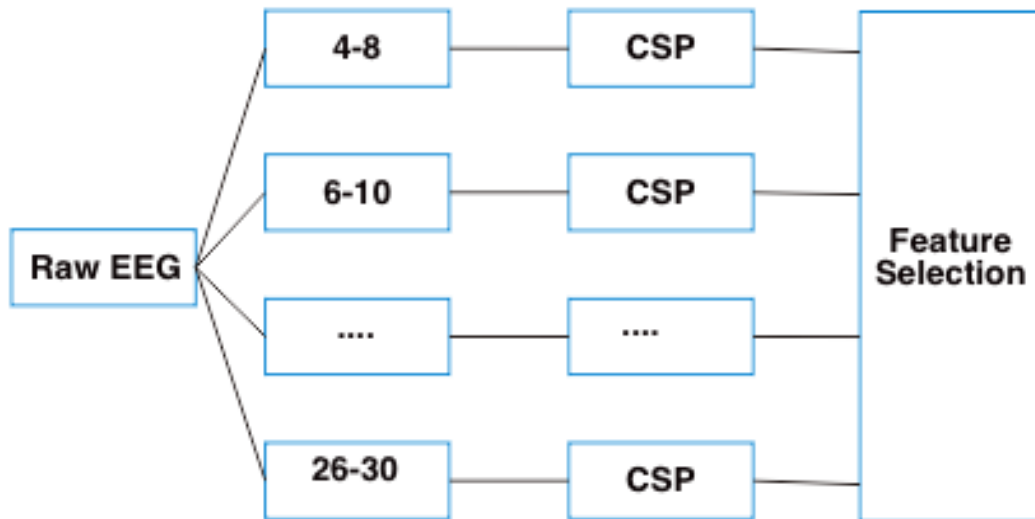


Figure 2: FBCSP Method

$$z_m = w^TX \quad (3)$$

The CSP variance-based features are then extracted from each trial using (4)

$$f_i = \log \left(\frac{\text{var}(Z_m)}{\sum (\text{var}(Z_m))} \right) \quad (4)$$

Where f_i is the i -th feature, and $\text{var}(Z_m)$ denotes the variance

of the m-th row of Z. The features f_i is then combined to form the feature vector of each trial. The features f_i is then combined to form the feature vector of each trial. The FBCSP procedure can be summarized as the following:

1. The EEG signals from all recorded channels are filtered using a filter bank with nine subsequent band-pass filters, starting at 4 Hz and with a bandwidth of 4 Hz (4–8 Hz, 8–12 Hz, . . .). All filters are type II Chebyshev filters.
2. Spatial filters for each output of the filter bank are computed using CSP. This is done by equation (1).
3. Spatial filters corresponding to the $2 \times$ extreme eigenvalues (largest and smallest eigenvalues) are selected. Each of the extreme spatial filters is then paired with each other correspondingly.
4. Energy (log variance) of the spatially filtered channels are obtained based on the equation (4).

5. CSP is designed for a two-class problem; in the case of multiclass tasks like this in this study, a one-versus-rest or one-versus-one strategy must be employed. In FBCSP, the former is chosen, and it will lead to a maximum of class number $\times 2 \times$ features. The values 2 and are selected using cross-validation. With the competition data containing four classes, the maximum number of features used for classification will be 32.

2.3 FBCSP-DNN framework

Multiple band-pass filters were used to filter the raw EEG data. Then, Spatial filtering using CSP is applied on each band-pass filtered data and the CSP variance-based features are extracted. The DNN is then trained using the selected features from session 1 and the trained DNN is used to classify features from session 2. The summary of the FBCSP-DNN framework is shown in Figure 2.

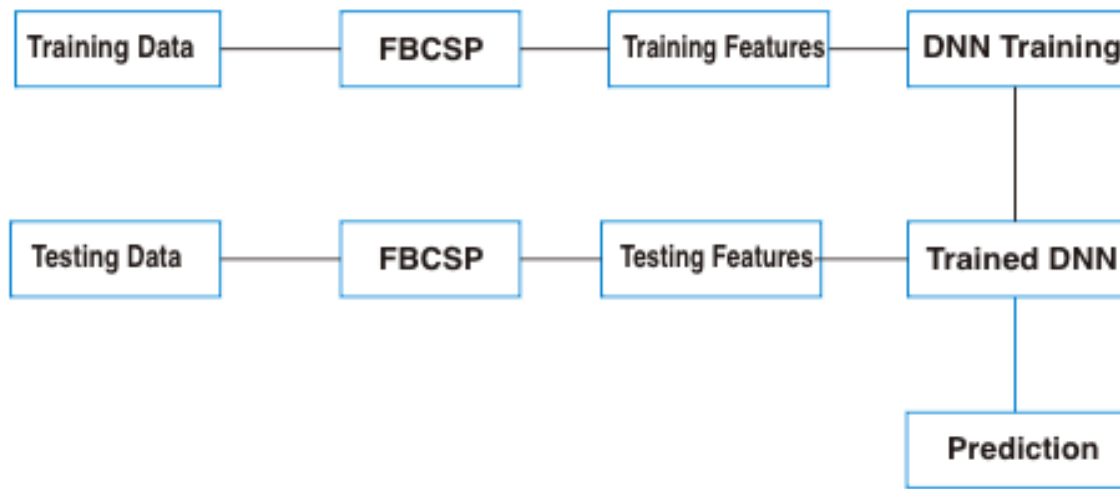


Figure 3: CSP-DNN proposed framework

2.4 DNN structure

Since the EEG signal is dynamic and varies for the same subject from subject to subject as well as for specific tests; hence a discriminative collection of features is needed to classify the data accurately. Among the DNN models, we choose the autoencoders to extract high-level features from the FBCSP features in a way to obtain a reliable and consistent representation of features across subjects. We designed four (4) layers DNN consisted of an input layer, two (2) hidden layer and an output layer. The FBCSP feature matrix is the inputs to the DNN. The second layer has 288 neurons, and the third layer has 100 neurons. The fourth layer is the Soft-max layer for classification. An auto-encoder comprises an encoder and a decoder. An auto-encoder will receive features, and "encode" them into features in a way that "decoder" can re-construct the same features as its output, the auto-encoder operation described in equation (5) and (6).

$$Y = f(w_{yx} + b_y) \quad (5)$$

$$Z = f(w_{zy} + b_z) \quad (6)$$

Where x is the input to the encoder and y is its output. Y is also an input to the decoder, z an output of an auto-encoder and f is log-sigmoid activation function, W_y and W_z are the

weights from input to hidden and hidden to output layers. b_y and b_z are bias values of hidden and output layers.

The representation learned by the encoders is useful for extracting relevant features for classification. The number of hidden neurons in an auto-encoder after training produces the same number of hidden neurons of newly learned features. Log-sigmoid function (describe in equation (7)) has been used for the auto-encoders.

$$f = f(a) = \frac{1}{(1 + \exp(-a))} \quad (7)$$

We stacked the soft-max layer with the auto-encoders for training and classification. Each Auto-encoder was trained for 200 epochs and then trained the soft-max layer for 200 epochs. A scaled conjugate gradient descent algorithm was used for training the SoftMax-layer. Cross-entropy has been used as the loss function. To improved classification performance, we retrained the whole network in a supervised fashion by performing back-propagation on the whole multi-layer network. This method used training set inputs and its corresponding target output (labels) to build a proper network behavior by computing the difference (E) using equations (9-10) to minimize the error between inputs and the target output. In this way, it adjusts the network parameters to find the

network model by minimizing the cost function E using equation (8). Here E (x, z) is the reconstruction error when the network is trained to reconstruct the output values equal to the applied input values.

$$\arg \min[E(x, z)] \quad (8)$$

The network then uses equation (11-13) to update the network parameters.

$$b_y = b_y - \alpha \frac{\partial E(x, y)}{\partial b_y} \quad (12)$$

$$b_z = b_z - \alpha \frac{\partial E(x, y)}{\partial b_z} \quad (13)$$

α is the learning rate of the algorithm.

$$w_{ij}^m(k+1) = w_{ij}^m(k) - \alpha \frac{\partial E(x, y)}{\partial w_{ij}^m(k)} \quad (9)$$

$$b_{ij}^m(k+1) = b_{ij}^m(k) - \alpha \frac{\partial E(x, y)}{\partial b_{ij}^m(k)} \quad (10)$$

$$w = w - \alpha \frac{\partial E(x, y)}{\partial w} \quad (11)$$

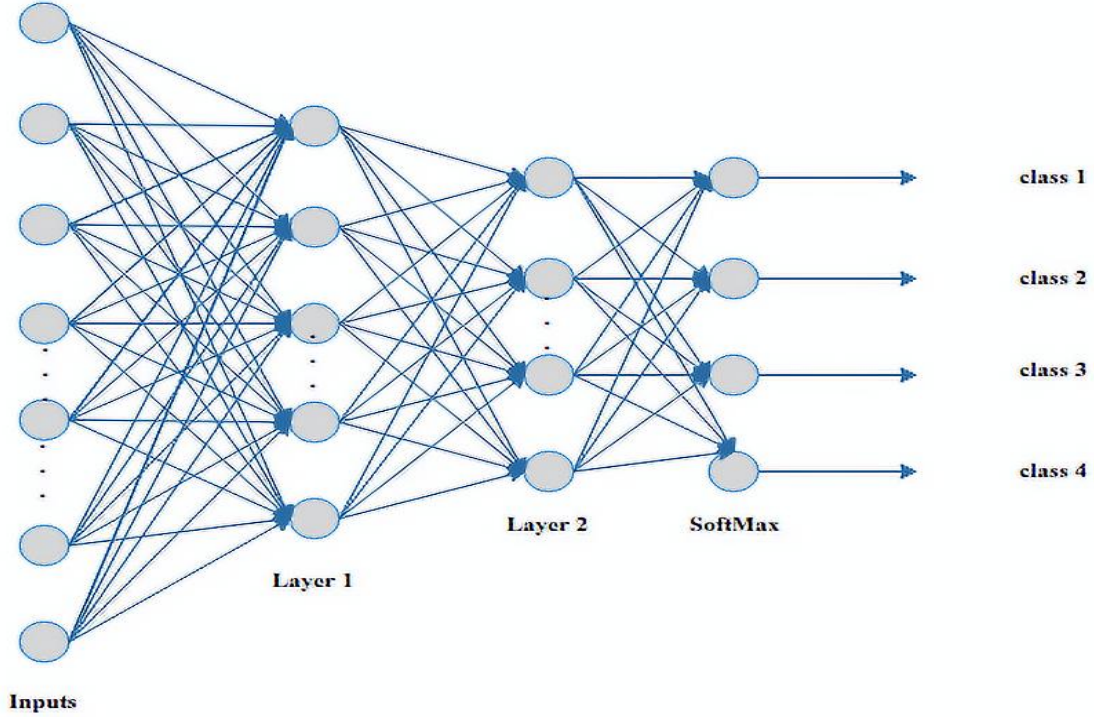


Figure 4: DNN architecture of the proposed method FBCSP-DNN

2.5 DNN Hyperparameter Search

Table 1: showing methods and hyperparameters used for DNN model selection

DNN parameters	FBCSP-DNN
layers	1-10
nodes	50-1800
Learning rate	0.001 to 10.
No of inputs	4-32

Table 2: showing the selected hyperparameters for the proposed FBCS-DNN

DNN parameters	nodes	epochs
No of inputs	16	-
Layers 1	288	100
Layers 1	100	100
Softmax	-	100

We choose the DNN parameters, which include several hidden nodes, the number of epochs, and other network-related parameters that are considered as hyperparameters and can be optimized for each layer. There are also hyperparameters governing the unsupervised pre-training for each layer and hyper-parameters for fine-tuning the whole model for classification. We used two hyper-parameters for the unsupervised learning part, which include the number of hidden nodes and maximum epochs. For the classification layer part, which is the supervised learning, we used the only number of maximum epochs, and all other parameters are fixed. Our objective was to select a set of hyper-parameters θ to obtained high accuracy within a reasonable training time.

We set hyper-parameters $\theta = (\theta_1, \theta_2, \dots, \theta_N)$ for each layer, then for each iteration or hyper-parameter, we start from small values to the higher values until we get the desired result. For better results, the procedure has been repeated for several iterations.

The DNN has two layers where the first layer has a 288 number of nodes with 100 as the maximum number of epochs and the second layer has a 100 number of nodes with 100 as the maximum number of epochs as shown in

Table 2. We perform 10-fold cross-validation only once to select the parameters by training layer 1, layer 2, sampling from the following:

For each layer, we chose:

1. Several hidden nodes (between 3 and 288).
2. Several layers (between 3 and 10).
3. Several iterations of contrastive divergence to perform for pre-training (from 50 to 500).

We chose a learning rate for the fine-tuning of the final classifier from 0.001 to 10.

3 RESULTS

In this section, the proposed model result was presented, as shown in Figure 5 which was determined by the hyperparameters search results. Then the final classification result of the proposed model was evaluated by comparing it to popular traditional methods and other DNN state of art methods.

The experiments were conducted in a MATLAB environment on an Intel 3.2 GHz Core i7 desktop with 8 GB of RAM. BCI Competition IV dataset 2a was used to evaluate our proposed framework for a session to session and cross-subject classification.

Several experiments were conducted before selecting the proposed FBCSP-DNN as explained above in section 2.5, both the size of FBSCP features and DNN hyperparameters were determined by taking classification accuracy and DNN training time into consideration. We used several kinds of inputs to train the proposed DNN, such as time-domain EEG data, frequency domain EEG data, band-passed raw EEG data, CSP feature data, and FBCSP feature data. Each input was trained with several DNN hyperparameters, as shown in Table 1. Our analysis shows that the selected parameters are effective to the EEG data and the network is indeed learning from the input EEG data, proving that the DNN selection of parameters is meaningful to the input data as shown in

Table 2. Note that applying DNN to raw EEG data is computationally expensive because we are applying a 403-by-626 matrix for each subject. Hence the hyperparameters selection was applied only to FBCSP features, and the selected DNN parameters for the proposed FBCSP-DNN were applied to the rest of the inputs.

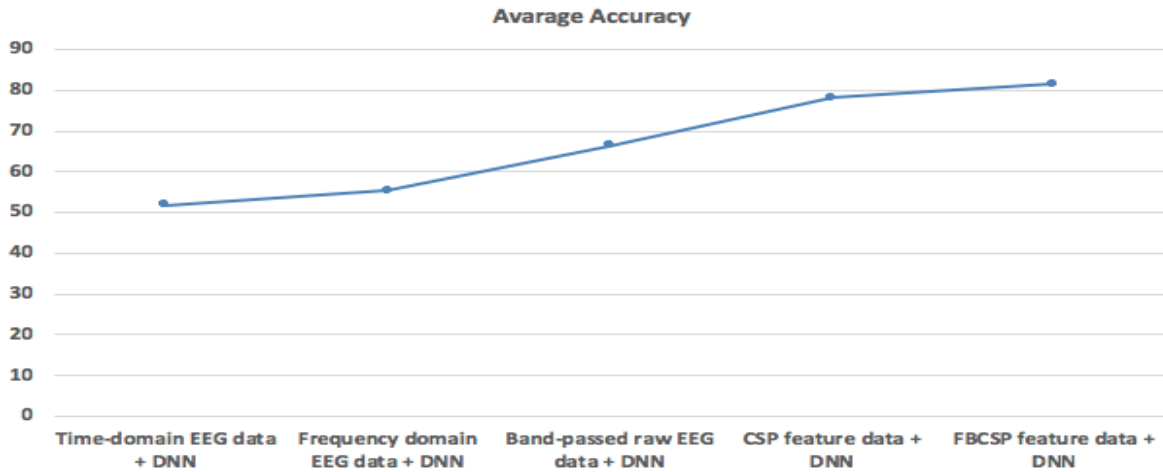


Figure 5: showing average accuracy results for all methods using 10 by 10-fold cross-validation

To evaluate the proposed model, two techniques have been used, which are often used by researchers to train BCI dataset systems. One of them divides the EEG data for each subject into a training set and test set. EEG data is usually recorded in multiple sessions; hence some sessions are put in the training set and the rest in the test set. In this way, the system is tested on sessions it has not seen before, but they belong to the same subject. This training technique called within-subject training is preferred by researchers and gives higher accuracy than the others. The second training technique involves subject to subject information transfer. One subject used as a testing set and all the remaining belong to the training set. Hence in this technique, the system is tested on the new subject altogether,

which it never saw before. This process is repeated for all users. This testing technique, called cross-subject training is more challenging than others, and the evaluation is more robust and generalized. We used both of these techniques for training and testing our proposed FBCSP-DNN method.

The DNN was trained and tested on each subject. The classification accuracy of each subject was validated using 10-fold cross-validation. The two (2) sessions, each consisting of 288 trials were combined and randomly divided into 7/10 and 3/10 for training and testing, respectively, and the results are shown in Table 3. We also analyzed transfer learning across sessions and subjects, then compare the FBCSP-DNN results to those obtained by traditional classifiers and other

competing methods. We could show in Table 4 and Table 5 that the proposed framework (FBCSP-DNN) performed better

than traditional methods and other studies respectively.

3.1 Comparison of Traditional methods and proposed FBCSP-DNN

Table 3: BCI competition IV dataset 2a Subjects Classification accuracy % (mean + standard deviation) for SVM, KNN, LDA and FBCSP-DNN

Subjects	SVM	KNN	LDA	FBCSP-DNN (Proposed method)
Subject 1	83.12 \pm 1.71	69.50 \pm 2.51	82.53 \pm 2.37	88.10 \pm 0.55
Subject 2	72.71 \pm 1.82	58.41 \pm 3.73	72.12 \pm 1.74	73.94 \pm 0.88
Subject 3	82.64 \pm 2.17	76.00 \pm 2.49	82.29 \pm 1.78	83.42 \pm 1.21
Subject 4	58.29 \pm 3.48	50.29 \pm 3.53	58.17 \pm 4.00	68.50 \pm 1.80
Subject 5	48.88 \pm 3.44	36.82 \pm 1.88	49.23 \pm 2.32	67.90 \pm 2.70
Subject 6	63.17 \pm 3.44	44.11 \pm 2.58	63.64 \pm 3.34	86.56 \pm 2.25
Subject 7	88.82 \pm 1.73	79.00 \pm 2.56	87.23 \pm 2.35	91.98 \pm 0.94
Subject 8	82.35 \pm 3.09	72.29 \pm 3.91	81.88 \pm 2.28	85.26 \pm 0.26
Subject 9	71.35 \pm 3.01	63.12 \pm 4.56	71.58 \pm 2.79	89.22 \pm 0.11
Average	72.37 \pm 2.65	61.06 \pm 3.08	72.07 \pm 2.55	81.43 \pm 1.18

We compared the FBCSP-DNN to the traditional methods (SVM, KNN, and LDA) to validate the FBCSP-DNN and examine its role for motor imagery classification. In

Table 3, the average classification accuracy of FBCSP-DNN performs higher than that of traditional methods classifiers for all subjects using the same features input. This shows the representation power of FBCP-DNN over traditional methods classifiers which extract most hierarchical features from FBCSP features before classification. FBCSP-DNN outperforms traditional method classifiers for all subjects with an average classification accuracy of 81.43% while SVM, KNN, and LDA having 72.37%, 61.06%, and 72.07%

respectively. The standard deviation of other methods is also higher than the proposed FBCSP-DNN. This demonstrates the robustness of the proposed method to the changes that occur between sessions of recording IM signals.

3.2 Comparison of Traditional methods, DNN methods and proposed FBCSP-DNN for within-subjects

Table 4: BCI competition IV dataset 2a within-subjects classification accuracy for traditional methods, DNN Methods, and FBCSP-DNN

Subjects	SVM	KNN	LDA	Schirrmeister [37]	Sakhavi [22]	Amin [7]	FBCSP-DNN (Proposed method)
Subject1	79.00	70.48	78.65	87.5	87.50	90.21	87.70
Subject 2	58.30	50.18	39.93	65.28	65.28	63.4	68.2
Subject 3	82.05	61.9	76.19	90.28	90.28	89.35	85.6
Subject 4	60.09	52.63	36.4	65.61	66.67	71.16	66.7
Subject 5	53.99	39.13	51.81	55.19	62.82	62.82	67.00
Subject 6	56.14	44.19	53.02	48.47	45.49	47.66	67.40
Subject 7	92.50	79.42	82.31	86.07	89.58	90.86	91.00
Subject 8	78.23	62.73	51.66	78.41	83.33	89.72	83.80
Subject 9	82.95	68.88	74.24	82.95	79.51	82.82	87.60
Average	71.47	58.84	60.47	73.31	74.46	75.78	78.33

To compare our work with some related studies, we applied session to session (within-session) with the FBCSP-DNN and compare the results. As can be seen in **Table 4**, the accuracy of the FBCSP-DNN is higher than that of traditional methods and related studies for 7 subjects out of 9 subjects except subject 3 and subject 4. FBCSP-DNN outperforms other methods with an average classification accuracy of 78.22% while SVM, KNN, LDA, Schirrmeister[37], Sakhavi[22] and amin[7] having 71.47%, 58.83%, 60.47, 73.31%, 73.56 and

75.78% respectively. The lower accuracy for other methods is due to the variation between session to session of subject 4, subject 5, and subject. Hence, our proposed FBCSP-DNN framework can capture the most discriminative features from one session and use it to classify another session.

3.3 Comparison of Traditional methods and DNN methods for cross-subjects

Table 5. Cross Subjects accuracy (%) for each subject for the BCI Competition dataset 2a.

Subjects	Schirrmeister [37]	Amin [7]	FBCSP-DNN (Proposed method)
Subject 1	47.60	62.07	78.60
Subject 2	31.22	42.44	62.50
Subject 3	41.02	63.12	72.50
Subject 4	33.19	52.09	53.90
Subject 5	41.57	49.96	44.20
Subject 6	34.71	37.16	49.30
Subject 7	43.09	62.54	78.40
Subject 8	46.01	59.32	67.90
Subject 9	51.78	69.43	78.40
Average	41.13	55.35	65.07

We also perform cross-subject classification and compare it with two of the studies ([7] and [37]) where we trained the FBCSP-DNN on 8 subjects and tested it with the remaining 1 subject, this procedure is repeated for all the 9 subjects. FBCSP-DNN outperforms these competing methods for all subjects except subject 5. The FBCSP-DNN has an average classification accuracy of 65.07% while on the other hand[7] and [37] have 55.35% and 41.13% respectively. This improvement in cross subjects' classification accuracy is important for FBCSP-DNN as it shows that it increases the performance by 9.72% in average classification accuracy compared to other studies, the result is shown in Table 5

4 DISCUSSION

We used the network as shown in **Figure 1**: It was described in section 2.4, it has only two layers and an output layer, the first layer has hidden nodes of 288 and second layer has hidden nodes of 100. This configuration was adopted after several experiments have been carried out that give good results in terms of minimum time and accuracy. Thus it had been used. This configuration found to be consistent in giving the highest classification accuracy through the subjects. Though in one of our experiments that has 10 nodes in the first layer and 5 nodes in the second layer can give the same result in some subjects, it was not consistent across all subjects and can lead to poor performance hence it has not adopted.

The FBCSP features are selected based on the first and last m column of CSP projected data. $m=2$ has been used for seven subjects except subject 3, subject 6, and subject 8, which used $m=3$, $m=4$, and $m=4$, respectively. These features dimension yielded the best result for the subjects when evaluating DNN parameter, so has been adopted. Therefore, the dimension of the proposed FBCSP-DNN is $m * N1 * N2 * 4$, where m is the FBCSP feature, $N1$ is the number of nodes in the first layer,

$N2$ is the number of nodes in the second layer, and 4 is the number of motor imagery classes. For each subject, FBCSP took about 9.62 seconds; the training session took about 12.13 seconds; testing took about 0.3 seconds. In total it took 21.75 seconds for training the FBCSP-DNN. When performing transfer learning, we just combined FBCSP features from other subjects for training and used the FBCSP features of the target subject for testing.

The effect of epoch size on the DNN performance is shown in Figure 6 together with the average training time for one training set with 288 trials. As you can see in the figure, the performance is higher for maximum epochs as 200 than greater than that, and the training time increases as epoch size increases. Thus, the Number of epochs were selected as 200 in each layer in this study. The size of hidden nodes is also crucial to the performance of classification accuracy. A comparison between performances of different hidden node sizes was investigated, and the DNN with hidden nodes the same as number testing set had the best classification accuracy. This means that the features are presented better by using these hidden nodes.

Traditional classifiers and other studies were also compared to the proposed FBCSP-DNN to investigate the role of the proposed FBCSP-DNN. It was found out that the proposed method was able to capture the similar features across sessions and subjects better than other methods. Notice that in Table 3 and Table there is an improvement in classification accuracy for some subjects that has low performance in other competing methods. Our analysis shows that low accuracies can also be due to the ability of some subjects to perform some of the motor imagery tasks correctly in some trials.

Overall, the proposed DNN and FBCSP feature combined performed better than other inputs both in terms of classification accuracy and minimum training time. In this

way, the FBCSP-DNN method limits the number of EEG samples and DNN parameters that result in long DNN training

time.

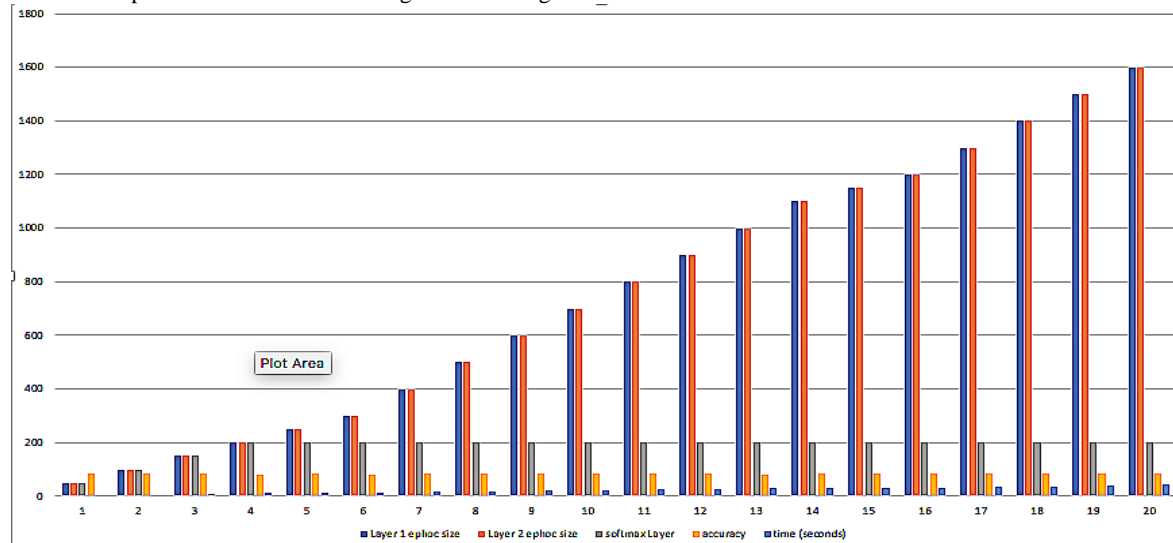


Figure 6: a graph showing the effect of epoch size on accuracy and time.

5 CONCLUSION

The proposed FBCSP-DNN was selected on the basis of several experiments conducted on numerous different frameworks with different parameter settings. Not only was the FBCSP able to reduce the high EEG dimension, but it also extracted significant features that contributed to the ideal DNN model. The proposed combination of FBCSP and DNN (FBCSP-DNN) was found to be more reliable as it achieves higher classification accuracy and computational efficiency than other state-of-the-art DNN methods. The FBCSP-DNN method can, therefore, be used reliably to develop a high-BCI-motor imagery application in a minimum of time and with good accuracy.

5.1 Data availability

The motor imagery dataset used to evaluate this study is a public benchmark dataset provided by the Institute of knowledge discovery. It can be accessed via a link <http://www.bbc.de/competition/iv/> upon application through the mentioned link.

6 REFERENCES

- [1] W. J. R. et al., "{B}rain-computer interface technology: a review of the first international meeting," IEEE Trans. Rehabil. Eng., vol. 8, no. 2, pp. 164–173, 2000.
- [2] F. Lotte, "A Tutorial on EEG Signal Processing Techniques for Mental State Recognition in Brain-Computer Interfaces. Eduardo Reck Miranda; Julien Castet. Guide to Brain-Computer Music Interfacing," 2014.
- [3] H. Lu, H. Eng, C. Guan, and S. Member, "2010 Regularized common spatial pattern with aggregation for EEG classification in small-sample setting.pdf," vol. 57, no. 12, pp. 1–10, 2010.
- [4] G. Schalk, J. Mellinger, G. Schalk, and J. Mellinger, "Brain Sensors and Signals," A Pract. Guide. toBrain-computerInterfacing with BCI2000, pp. 9–35, 2010.
- [5] F. Lotte, E. Signal, and C. Techniques, "Study of Electroencephalographic Signal Processing and Classification Techniques towards the use of Brain-Computer Interfaces in Virtual Reality Applications To cite this version?: HAL Id?: tel-00356346 Thèse 1 ' Institut National des Sciences Appliquées," 2009.
- [6] S. Zhang, S. Wang, D. Zheng, K. Zhu, and M. Dai, "A novel pattern with high-level commands for encoding motor imagery-based brain-computer interface," Pattern Recognit. Lett., vol. 125, pp. 28–34, 2019.
- [7] S. U. Amin, M. Alsulaiman, G. Muhammad, M. A. Mekhtiche, and M. Shamim Hossain, "Deep Learning for EEG motor imagery classification based on multi-layer CNN's feature fusion," Futur. Gener. Comput. Syst., vol. 101, pp. 542–554, 2019.
- [8] H. Kang and S. Choi, "Bayesian common spatial patterns for multi-subject EEG classification," Neural Networks, vol. 57, pp. 39–50, 2014.
- [9] N. Brodu, F. Lotte, and A. Lécuyer, "Comparative study of band-power extraction techniques for Motor Imagery classification," IEEE SSCI 2011 - Symp. Ser. Comput. Intell. - CCMB 2011 2011 IEEE Symp. Comput. Intell. Cogn. Algorithms, Mind, Brain, pp. 95–100, 2011.
- [10] H. Cho, M. Ahn, K. Kim, and S. Chan Jun, "Increasing session-to-session transfer in a brain-computer interface with on-site background noise acquisition," J. Neural Eng., vol. 12, no. 6, p. 66009, 2015.
- [11] S. M. Plis et al., "Deep learning for neuroimaging: A validation study," Front. Neurosci., vol. 8, no. 8 JUL, pp. 1–11, 2014.
- [12] M. Alamgir, M. Grosse-Wentrup, and Y. Altun, "Multitask learning for brain-computer interfaces," J. Mach. Learn. Res., vol. 9, pp. 17–24, 2010.
- [13] C. T. Vi and S. Subramanian, "Detecting Error-Related Negativity for interaction design," Conf. Hum. Factors Comput. Syst. - Proc., pp. 493–502, 2012.
- [14] I. Iturrate, L. Montesano, and J. Minguez, "Shared-control brain-computer interface for a two-dimensional reaching task using EEG error-related potentials," Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBS, pp. 5258–5262, 2013.
- [15] R. Chavarriaga, I. Iturrate, Q. Wannebroucq, and J. D. R.

- Millan, "Decoding fast-paced error-related potentials in monitoring protocols," *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBS*, vol. 2015-Novem, pp. 1111–1114, 2015.
- [16] J. Oxides, I. Iturrate, J. Minguez, and L. Montesano, "Analysis and asynchronous detection of gradually unfolding errors during monitoring tasks," *J. Neural Eng.*, vol. 12, no. 5, p. 56001, 2015.
- [17] S. Fazli, F. Popescu, M. Danóczy, B. Blankertz, K. R. Müller, and C. Grozea, "Subject-independent mental state classification in single trials," *Neural Networks*, vol. 22, no. 9, pp. 1305–1312, 2009.
- [18] J. Thomas, T. Maszczyk, N. Sinha, T. Kluge, and J. Dauwels, "Deep learning-based classification for brain-computer interfaces," *2017 IEEE Int. Conf. Syst. Man, Cybern. SMC 2017*, vol. 2017-Janua, pp. 234–239, 2017.
- [19] Y. R. Tabar and U. Halici, "A novel deep learning approach for classification of EEG motor imagery signals," *J. Neural Eng.*, vol. 14, no. 1, p. 16003, 2017.
- [20] H. Dose, J. S. Møller, H. K. Iversen, and S. Puthusserypady, "An end-to-end deep learning approach to MI-EEG signal classification for BCIs," *Expert Syst. Appl.*, vol. 114, pp. 532–542, 2018.
- [21] Z. Tang, C. Li, and S. Sun, "Single-trial EEG classification of motor imagery using deep convolutional neural networks," *Optik (Stuttg.)*, vol. 130, pp. 11–18, 2017.
- [22] S. Sakhavi, C. Guan, and S. Yan, "Learning Temporal Information for Brain-Computer Interface Using Convolutional Neural Networks," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 29, no. 11, pp. 5619–5629, 2018.
- [23] R. Alazrai, M. Abuhijleh, H. Alwanni, and M. I. Daoud, "A Deep Learning Framework for Decoding Motor Imagery Tasks of the Same Hand Using EEG Signals," *IEEE Access*, vol. 7, pp. 109612–109627, 2019.
- [24] A. Yuksel and T. Olmez, "A neural network-based optimal spatial filter design method for motor imagery classification," *PLoS ONE*, vol. 10, no. 5, 2015.
- [25] F. Lotte et al., "A review of classification algorithms for EEG-based brain-computer interfaces: A 10-year update," *J. Neural Eng.*, vol. 15, no. 3, 2018.
- [26] I. B. C. Interface et al., "Weighted Transfer Learning for Improving Motor," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 7, pp. 1–34, 2019.
- [27] X. An, D. Kuang, X. Guo, Y. Zhao, and L. He, "A deep learning method for classification of EAn, X., Kuang, D., Guo, X., Zhao, Y., & He, L. (2014). A deep learning method for classification of EEG data based on motor imagery. *International Conference on. Retrieved from http://link.springer.com/chapter,*" *Int. Conf.*, pp. 203–210, 2014.
- [28] N. Lu, T. Li, X. Ren, and H. Miao, "A Deep Learning Scheme for Motor Imagery Classification based on Restricted Boltzmann Machines," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 6, pp. 566–576, 2017.
- [29] S. Bach, A. Binder, G. Montavon, F. Klauschen, K. R. Müller, and W. Samek, "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation," *PLoS One*, vol. 10, no. 7, pp. 1–46, 2015.
- [30] K. K. Ang, Z. Y. Chin, H. Zhang, and C. Guan, "Filter Bank Common Spatial Pattern (FBCSP) in brain-computer interface," *Proc. Int. Jt. Conf. Neural Networks*, pp. 2390–2397, 2008.
- [31] S. Kumar, A. Sharma, and T. Tsunoda, "An improved discriminative filter bank selection approach for motor imagery EEG signal classification using mutual information," *BMC Bioinformatics*, vol. 18, no. Suppl 16, 2017.
- [32] S. Kumar and A. Sharma, "A new parameter tuning approach for enhanced motor imagery EEG signal classification," *Med. Biol. Eng. Comput.*, vol. 56, no. 10, pp. 1861–1874, 2018.
- [33] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe, and K.-R. Muller, "Filters for Robust EEG," *IEEE Signal Process. Mag.*, vol. January, no. 2008, pp. 41–56, 2008.
- [34] C. Brunner, M. Naeem, R. Leeb, B. Graimann, and G. Pfurtscheller, "Spatial filtering and selection of optimized components in four class motor imagery EEG data using independent components analysis," *Pattern Recognit. Lett.*, vol. 28, no. 8, pp. 957–964, 2007.
- [35] M. Naeem, C. Brunner, R. Leeb, B. Graimann, and G. Pfurtscheller, "Separability of four-class motor imagery data using independent components analysis," *J. Neural Eng.*, vol. 3, no. 3, pp. 208–216, 2006.
- [36] K. K. Ang, Z. Y. Chin, H. Zhang, and C. Guan, "Robust Filter Bank Common Spatial Pattern (RFBCSP) in motor-imagery-based brain-computer interface," *Proc. 31st Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. Eng. Futur. Biomed. EMBC 2009*, no. January 2014, pp. 578–581, 2009.
- [37] R. T. Schirrmester et al., "Deep learning with convolutional neural networks for EEG decoding and visualization," *Hum. Brain Mapp.*, vol. 38, no. 11, pp. 5391–5420, 2017.