Generating ten BCI commands using four simple motor imageries and classification by divergence-based DNN

Nuri Korhan1 · Tamer Olmez1 · Zümray Dokur1

Received: 12 October 2021 / Accepted: 6 September 2022 / Published online: 26 September 2022
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2022

Abstract
The brain computer interface (BCI) systems are utilized for transferring information among humans and computers by analyzing electroencephalogram (EEG) recordings. However, in this emerging research field, the number of commands in the BCI is limited in relation to the number of motor imagery (MI) tasks; in the current literature, mostly two or four commands (classes) are studied. As a solution to this problem, it is recommended to use mental tasks as well as MI tasks. Unfortunately, the use of this approach reduces the classification performance of MI EEG signals. The fMRI analyses show that the resources in the brain associated with the motor imagery can be activated independently. It is assumed that the activity of the brain produced by the MI of the combination of body parts corresponds to the superposition of the activities generated during each body part’s simple MI. In this study, in order to create more than four BCI commands, we suggest to generate combined MI EEG signals artificially by using tongue, feet, left and right hands’ motor imageneries in pairs. For the first time in the literature, combined MI EEG signal is artificially generated by using the superposition of simple MI EEG signals from two different sources. We observe in the literature that the classification performances are adversely affected as the number of classes is increased, and the classification performances for the MI with more than four classes are poor. The aim of this study is to increase the BCI commands by generating artificially combined MI EEG signals and to achieve high success rates for ten BCI commands by using a small-sized deep neural network (DNN). In this context, by analyzing the ERD signal of combined MI tasks, we investigate how to generate combined MI signals artificially using the superposition of simple MI signals. The proposed method is validated on real data which consist of simple and combined MI EEG signals, and average classification performance of 81.8% is achieved for ten BCI commands generated from the BCI Competition 3 and 4 datasets.

Keywords EEG · Motor imagery EEG classification · Artificially generating EEG · Brain computer interface · Convolutional neural network

1 Introduction

In order to establish a machine learning based means for human–computer interaction, the electroencephalogram (EEG) signals acquired by a brain computer interface (BCI) system is translated into commands that perform a specific desired action. There is a growing interest on the research carried out in the field of BCI applications in recent years. Especially for the classification of motor imagery (MI) signals, many new methods have been proposed [1–35]. The process of mentally previewing a movement without any physical output is described as motor imagery. The activation of the specific areas on the human brain is similar when the subject imagines it or performs the action in real world [13].

The EEG signals mostly resemble random noise. Unlike the electrocardiogram, they do not have any specific wave pattern. When working with motor imagery signals, researchers should know that (1) the EEG has low spatial resolution, (2) EEGs show variations between sessions and subjects, (3) classification of multiple classes with high performances is not an easy task, (4) data collection can sometimes be difficult if the subject needs a pre-training/
preparation, and (5) in order to compensate for all those issues a complex algorithm might be needed to be developed in order to increase the classification accuracy, which, in turn, makes the system very hard to implement in real-life applications [13]. The number of electrodes mounted on the subject’s head determines the spatial resolution. The resolution can be improved by increasing the number of electrodes keeping in mind the fact that it will require more computational power for the BCI system. Too many electrodes cause the experiments to be impractical for the researchers and uncomfortable for the subjects. One of the greatest challenges of BCI is that the manifestation of the same imagined movement in the EEG recordings varies between the subjects, and even between two different sessions for the same subject. Another disadvantage of working with EEG recordings is that the number of BCI commands is limited with the number of MI tasks that can be accurately discriminated from each other; in the current literature, mostly two [3, 6–8, 10, 12] or four commands [1, 2, 4, 5, 9, 11, 15, 23, 25, 32] (classes) are studied. One classifier which shows good performance for one subject might not show the similar level of performance for another one, which is also another cause of problem. Therefore, the training of the subjects before the final data collection task becomes very critical. To make the final system efficient, not only the correct classification, but also the speed is also an important factor for the classification of motor imagery EEGs. Since the researchers are mostly focused on the classification accuracy using the benchmark datasets [36, 37], more complicated DNN [1, 21, 34, 35] or traditional neural network [22–33] structures have been developed. Such complicated algorithms make it impossible to integrate them into portable embedded systems [15, 16]. Moreover, to determine the optimum features for a subject, a relatively large amount of data is needed [6], but the datasets used in the literature are of small sizes [36, 37]. In a previous study, researchers suggested to use a novel method of data augmentation on small-sized EEG sets in order to increase the success rate of MI signals [38] and also proposed to use a novel convolutional neural network (CNN) structure, called as DivFE [38, 39]. Another problem is to increase the number of BCI commands which is limited by the number of MI tasks that can be recognized individually. In this study, while dealing with the problems stated in (3–5), we will be more concerned with increasing the number of BCI commands.

2 Related works

In the current literature, at most four commands (tongue, feet, left and right hands) are studied [1, 2, 4, 5, 9, 11, 15, 23, 25, 32, 38]. In Fig. 1, motor imageries related to the left and right feet are somewhere inside the red ellipses. MI EEG signals for the left and right feet were not even used in all BCI competition datasets. The reason for not using left foot and right foot as different classes is that it is almost impossible to discriminate MI EEGs of left and right feet from each other. For this reason, both feet are used as a single class in these benchmark datasets [36, 37]. MI centers for left and right feet are inside the brain and are very close to each other, as shown in Fig. 1. For this reason, in most of the four-class problems, classes were chosen as feet, tongue, right and left hands. The MI locations of these four classes have been chosen particularly far from each other. In fact, it is necessary to use more electrodes on the scalp in order to distinguish close centers in the brain. In the literature, there is only one study that succeeds in distinguishing MIs related to the right and left feet by increasing the number of electrodes on the scalp [40]. However, due to the increased complexity and computational burden, this approach makes the system less efficient.

Another way to increase the number of BCI commands is to use the MIs in pairs [41–46]. In [41], MI signals from six healthy subjects were recorded at 250 Hz by using the OpenViBE platform. An EEG cap that incorporates 26 electrodes was mounted according to the international 10–20 system. MI EEG recordings corresponding to both hands, right hand, left hand and a ‘rest’ task in which the subjects must not think about any motor movement, were classified with an average success rate of 51.6%. In the study [43], eight healthy subjects (two males and six females, 23–25 years old) participated in four different MI tasks. For the MI EEG BCI system, twenty electrodes were mounted over the motor cortex. MI signals corresponding to both hands, feet, the right and left hands were classified with an average success rate of 54.2% by using the common spatial patterns (CSP) method. In the study [45], researchers had made experiments with seven subjects that were given eight different MI tasks (rest, feet, right hand, left hand, right hand–feet, both hands, left hand–feet and both hands–feet) which they had to perform in a series of trials. The EEG cap was comprised of 26 electrodes, and the electrodes were mounted according to the international 10–20 system in order to cover the cerebral cortex. Eight different MI EEG signals were classified with an average success rate of 55% by using the CSP. In [46], seven females and three male subjects (23–25 years old, all right-handed) were given three simple MI tasks, three combined MI tasks and a ‘rest’ task. In the experiments, the EEG recordings were collected via 64 Ag/AgCl electrodes. MI signals corresponding to the right hand, feet, left hand, both hands, right hand–left foot, left hand–right foot and rest were classified with an average success rate of 70% by using the CSP.
In the studies of [41–46], it is observed that the CSP gives low classification performance. One of the reasons for the low performance is that the CSP method was applied to combined MIs. It is known that the CSP is a transformation that reveals the distinction between the samples of different classes. In simple MIs, event-related desynchronization (ERD) and synchronization (ERS) signals are generated at a single location in the brain. However, in combined MIs, such as the left hand versus both hands, ERD and ERS signals will be generated at more than one location in the brain. Therefore, the use of the CSP may not be convenient in distinguishing combined MIs.

In another study [41], the subjects were given the tasks of performing four different MIs: both hands, rest, right hand, and left hand. The second and third figures in [41] show ERD/ERS ratios of MIs corresponding to both hands, right hand, left hand, and rest for C3 and C4 electrodes, respectively. As a result, the ERD/ERS ratios of combined MI signals (both hands) appear as a superposition of simple MI signals (left hand or right hand). The fourth figure in [41] further shows that this assumption is correct. It is a result of the subjects’ being able to make several motor movements at the same time.

In our study, in order to create more than four BCI commands, we suggest to generate combined MI EEG signals artificially by using simple feet, tongue, right and left hands motor imageries of the BCI Competitions 3 and 4. A maximum of ten (4 simple MIs + C(4,2) combined MIs) different BCI commands are generated artificially by using four simple MIs in pairs. It is observed that increasing the number of classes generally leads to a decrease in the success rate, and accuracies for MIs with more than four classes are poor. The aim of this study is to increase the number of BCI commands by generating artificially combined MI EEG signals and to achieve a high success rate for ten BCI commands by using a small-sized deep neural network (DNN). For the first time in the literature, ten BCI commands are generated and high classification performances are achieved for these commands. In the Computer Simulations section, ERD signals of simple and combined MI tasks are analyzed, and we investigated how to generate combined MI signals artificially using the superposition of simple MI signals. The proposed method is validated on real data which consist of simple and combined MI EEG signals, and success rates are given for ten BCI commands generated from the BCI Competitions 3 and 4 datasets.

3 Methodology

3.1 Structure of the proposed convolutional neural networks

Most DNN structures are composed of convolutional layers which extract features from the samples, and fully connected layers which pave the path to the classification. In addition to these layers, a DNN structure contains pooling and batch normalization layers to increase the performance of the network. Another indispensable component of DNN is the layers of activation units that determine the behavior of the neurons, and this layer is usually selected as rectified linear units (ReLU).

Having a number of small-sized learning filters that are convolved with each channel of EEG recordings in order to create one-dimensional activation maps, the critical part of the CNN is the convolutional layers. The main purpose of these layers is to reveal features that are of greater importance in representing the data. When the size of the
representation needs to be decreased, either the strides of the filters are increased or the pooling layers are inserted between the convolutional layers in order to carry out the down-sampling operation while also contributing to the translation invariance (reducing the variance of the data). In comparison with the other functions such as the average-pooling and min-pooling, the max-pooling is the most favored nonlinear function to implement pooling operation in research and applications. Utilization of pooling layers in CNNs in the literature is wide despite the fact that the need is problem dependent. Likewise, whether a pooling layer should be placed after a convolutional layer is mostly determined by the trial-and-error method. Pooling layers are deployed between all convolutional layers of the model. In our model, the activation function is selected as ReLU, which performs the nonlinear operation of neutralizing all the negative values of the activation maps. Additionally, two of the most common regularization methods are used to obtain a more robust model. Having all the components of both the feature extractor (CNN) and the classifier (fully connected neural network-FCNN) at hand, we face the challenging task of tuning all hyper-parameters such as the number of neurons in the hidden layers, the number of hidden layers in the DNN, the size of filters, the number of convolutional layers and the number of filters in the convolutional layers, simultaneously. This tuning process may take weeks in the studies that contain a relatively large dataset. If the DNN is constructed by using only the CNN part of the aforementioned structure, the number of hyper-parameters that needs to be tuned decreases dramatically. In [38, 39], researchers had put forward a DNN structure that contains the CNN and replaced the FCNN with a minimum distance network (MDN). This structure is called DivFE (Divergence-based Feature Extractor), and shown in Fig. 2 in detail.

When the FCNN is used in a network that aims to solve a machine learning problem, it brings a number of issues that are not easy to tackle. Some of these issues are high memory requirement, high computational cost, difficulty of convergence and additional hyper-parameters. An MDN, on the other hand, deals with extracted features in a much easier manner. When the MDN is used as a classifier, the class labels of the MDN are set to the Walsh vectors (columns or rows of the Walsh matrix). The label of the data is founded by the distance between the feature extractor’s (FE) output and the given previously
determined Walsh vector. The MDN node that is closest to the output of the FE sets the class label for that input data. Given that, \( M \) is the dimension of the Walsh matrix (which is equal to 16 in this study), \( O_j \) is the \( j \)th output of the flatten layer (see Fig. 2), \( h_{c,j} \) is the \( j \)th element of the Walsh vector of the \( c \)th class, and the class label is determined according to the following equation:

\[
d_c = \frac{\sum_{j=1}^{M} (O_j - h_{c,j})^2}{d_i} ; d_i = \min(d_c)
\]

where index \( i \) corresponds to the class label of the input epoch that is entered into the FE.

### 3.2 Training process of the convolutional layers

In deep learning applications, as the number of parameters to be trained increases, training the DNN becomes more and more difficult. The reason for this phenomenon is that as the number of trainable parameters increases, the number of local minima also increases and the algorithm becomes more likely to be stuck in some local minima. It has been shown in previous studies \([38, 39]\) that feature extractor and classifier can be trained separately, one after another, as long as the FE is trained before the classifier. Once the features are proven to be extracted successfully, the minimum distance network can be used instead of the FCNN for the classification process.

In this study, we built up a DNN that aims to maximize the divergence for the feature space so that the proposed DNN can converge to the desired output more easily. Maximum divergence value for the feature space refers to the strength of the features in representing the intra-class variation and inter-class discrimination. The following equation gives a class separability measure:

\[
\text{divergence value} = \frac{\text{tr}(B)}{\text{tr}(S)}
\]

\[
S = S_1 + \cdots + S_c + \cdots + S_C
\]

where \( C, S_c, S \) and \( \text{tr}(\cdot) \) refer to the number of classes, the covariance matrix of the \( c \)th class, the sum of all the covariance matrices and the trace operation, respectively. Being the measure of how far apart class centers are from each other, the \( B \), is the covariance matrix of the mean vectors of all classes. In the literature, \( B \) and \( S \) are also called between-class scatter matrix and within-class scatter matrix, respectively. As seen in Eq. 2, high divergence value is obtained by minimizing the \((i)\) within-class scatters, \( \text{tr}(S) \) and maximizing the \((ii)\) between-class scatters, \( \text{tr}(B) \).

In this study, a training strategy that will increase the divergence value is selected. In order to analyze the scattering, the divergence value is calculated for the output vectors \( \{o_j\} \) of the FE. The \( S \) matrix is created with \( o \) vectors. In this respect, in order to increase the divergence
(a) **Algorithm 1:** Basic DNN: Feature Extractor (CNN) + Classifier (FCNN)

1- **One iteration of the training process for two classes**

for each input data in (training set) do
   run the network (FE + FCNN) and obtain the network output \( \mathbf{o} = [o_1, o_2] \)
   define desired output: \([0, 1]^T\) for the input data of the first class
   \([1, 0]^T\) for the input data of the second class
   update weights of network (FE + FCNN) by calculating \(\sum (\text{network output} - \text{desired output})^2\)
end for

2- **Validation/Test process for two classes**

for each input data in (validation/test set) do
   run the network (FE + FCNN) and obtain the network output \( \mathbf{o} = [o_1, o_2] \)
   obtain the decision "r" of basic DNN by calculating the below equation
   \( o_i = \max (o_1, o_2) \)
end for

**calculate accuracy by comparing the decision of basic DNN for each input data with the class label of the input**

(b) **Algorithm 1b:** DivFE in Fig. 1 and Fig 2.: Feature Extractor (FE or CNN) + Classifier (MDN)

1- **One iteration of the training process for two classes**

for each input data in (training set) do
   run the network (FE) and obtain the network output \( \mathbf{o} \)
   define desired output: \( \mathbf{h}_r = [1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0]^T \) for the input data of the first class
   \( \mathbf{h}_i = [1,1,0,0,1,0,0,1,0,0,1,1,0,0,1,0]^T \) for the input data of the second class
   update weights of network (FE) by using \(\sum (\text{network output} - \text{desired output})^2\)
end for

2- **Validation/Test process for two classes**

for each input data in (validation/test set) do
   run the network (FE) and obtain the network output \( \mathbf{o} = [o_1, \ldots, o_n] \)
   obtain the decision "c" of DivFE by calculating the below equation for \( \mathbf{h}_r \) and \( \mathbf{h}_i \)
   \[d_c = \sum_{i=1}^{H} (o_i - h_{c,i})^2 \quad d_i = \min (d_c)\]
end for

**calculate accuracy by comparing the decision of the DivFE for each input data with the class label of the input**

value, it is necessary to reduce the (i) within-class scatters, \( \text{tr} (S) \). Since it is proposed to train the two blocks of DNN (CNN + FCNN) separately, training is carried out only on the weights in the convolutional layer (feature extractor). Because the FCNN is not employed, the training algorithm is focused only on the features, so eventually, a decrease in the (i) within-class scatters, \( \text{tr} (S) \), is obtained.

Ensuring the class centers as distant as possible from each other, while the distance between any two classes remains the same across all classes, was made successfully by representing each class center by a Walsh vector. By assigning each input vector to a Walsh vector, the feature extractor is trained to output the specific Walsh vector which is the class label of this input. Assigning the output vectors of the feature extractor as Walsh vectors increases the (ii) distances between the centers of the classes, \( \text{tr} (B) \), thus the divergence value which represents efficient features is increased. The \( B \) matrix is created by using the \( h_c \) vectors that are shown in Fig. 2. A symbolic representation of two-dimensional and four-dimensional modified Walsh matrices (substituting zero for the − 1 values) is seen in Eq. 3. It can easily be seen in the modified Walsh matrix that the Hamming distance (HD) between any two rows is half the rank value of the matrix. (The dimension of output vectors is equal to the rank.) Hence, the value of MDN nodes for each class label is selected among the \( h_c \) vectors. Increasing the rank of the matrix also increases the HD between the centers of the classes. The rank of the Walsh

![Springer](image_url)
matrix should be selected carefully in order to avoid underfitting and prolonged training time.

The training algorithm of the DivFE is shown in Fig. 3. Each frame of EEG signals is called an epoch which is represented by a multi-dimensional input signal matrix, EEG(k,n), where k and n are the channels and the time samples, respectively. The epochs are formed by using the EEG data that is acquired after presenting the cue which requests the subject to initiate the corresponding motor imagery task. In this study, 1D convolution filters are applied to the input epoch matrices EEG(k,n).

The algorithms in Fig. 4 illustrate the training, validation and test processes of the basic DNN and DivFE. Algorithms 1a and 1b reveal the differences between the training procedures and structures of the basic DNN and DivFE: (i) in the training process of the DivFE, Walsh vectors are used as the desired (target) output vectors, and (ii) the MDN is preferred instead of the FCNN as a classifier. Algorithms 1a and 1b are the pseudocode of the system and show the operations performed in one iteration for a general two-class problem.

### 3.3 Dataset preparation and validation process

for the DivFE

Combined MI signals (left hand&feet—EEG<sub>LH-F</sub>, right hand&left hand—EEG<sub>RH-LH</sub>, feet&tongue—EEG<sub>F-T</sub>, right hand&tongue—EEG<sub>RH-T</sub>, left hand&tongue—EEG<sub>LH-T</sub>, and right hand&feet—EEG<sub>RH-F</sub>) are artificially created by using four simple MI signals (feet, tongue, right and left hands) of the BCI.

Competitions 3 and 4 datasets. In order to generate a combined MI signal, one simple MI signal is added to other simple MI signals as follows:

\[
\begin{align*}
\text{EEG}_{LH-F}(i) & = \frac{\text{EEG}_{LH}(i) + \text{EEG}_{F}(i)}{2} \\
\text{EEG}_{LH}(i) & = \frac{\text{EEG}_{LH}(i) + \text{EEG}_{R}(i)}{2} \\
\text{EEG}_{F-T}(i) & = \frac{\text{EEG}_{R}(i) + \text{EEG}_{T}(i)}{2} \\
\text{EEG}_{LH-T}(i) & = \frac{\text{EEG}_{LH}(i) + \text{EEG}_{T}(i)}{2} \\
\text{EEG}_{R-F}(i) & = \frac{\text{EEG}_{F}(i) + \text{EEG}_{R}(i)}{2} \\
\text{EEG}_{RH-LH}(i) & = \frac{\text{EEG}_{R}(i) + \text{EEG}_{LH}(i)}{2}
\end{align*}
\]

where \(k\) and \(n\) represent the channels and samples, respectively; \(\text{EEG}_{LH-F}(i)\) represents the \(i\)th tongue-feet combined MI epoch; \(\text{EEG}_{LH}(i)\) and \(\text{EEG}_{F}(i)\) represent the simple tongue and feet MI epochs, respectively. In this way, all the combined MI signals are artificially generated for each subject. The number of combined MI EEG signals is equal to the number of samples of tongue or feet.

The validation set is made up of 10% of training data. 90% of the training set is used for the training of feature extractor, and then, mean squared error loss and accuracy values are computed for all data of the validation set. In the phase of the validation, each input epoch in the validation dataset is entered to the feature extractor (shown in Fig. 2), and the feature extractor produces an output for each input epoch. The generated output vector \(O\) serves as an input for the MDN. The final classification is done by calculating the closest node of MDN to the output vector \(O\). The closest node of the MDN determines the class of the vector.

The loss values and mean accuracies are computed over all epochs of the validation set. To avoid under- and overfitting-related issues, mean squared error (MSE) and accuracy values for both the validation and training datasets are examined. Relevant to the literature, hyperparameters of the FE network, \(i.e.,\) filter length, number of features in each layer, number of layers, etc., are determined by trial and error. The training for longer iterations is started once the coarse structure of the FE network is finalized. Validation dataset’s loss values and the number of iterations are the parameters that are considered for the termination of training. For \(N\) randomly formed test and training datasets, the feature extractor is trained \(N\) times, and then the average of \(N\) success rates is calculated for test datasets in order to demonstrate the statistical significance of the results.

### 4 Computer simulations

All simulations were done on a Linux-based workstation by using Python. The CPU had 32 cores which are clocked at 2.7 GHz. In addition, the workstation was equipped with a GTX2080 Ti graphics card. The benchmark datasets BCI Competitions 3 [36], Competitions 4 [37] and other dataset acquired by The OpenViBE platform [45] are utilized for training and testing of the DivFE in MI signal classification.

Table 1 presents a summary of the recent studies performed for the classification of MI EEG signals by the DNNs. Table 1 shows the success rates of the studies in the literature and authors’ previous study [38]. Table 2 provides a summary of the recent studies that classified motor imagery signals by traditional neural networks. As given in Tables 1 and 2, the accuracies achieved by DNNs are higher than those achieved by traditional neural networks. Moreover, in order to determine the features, there is no need to spend too much effort. Thus, the features are extracted by DNNs automatically during training.
4.1 Analysis of the ERD signals for simple and combined MI

In this section, relative average ERD power signals will be plotted and studied for the combined MI signals artificially produced using the simple MI signals. Especially both hands, the right and left hands, MI signals will be analyzed for C3 and C4 electrodes, and the results will be compared with the graphs in the study [41]. In this context, the advantages and disadvantages of generating combined MI signals using simple MI signals will be discussed. In this section, only BCI Competitions 4 dataset will be used for the analyses. Then, the classification results for simple and combined MIs obtained by using the BCI Competitions 3 and 4 databases will be given.

In the BCI Competition 4 dataset [37], 22 electrodes were applied to record EEGs; the electrode connections are shown in Fig. 5. The signals were acquired at 250 Hz, and the dataset is made up of the EEG recordings from nine subjects. The paradigm is cue-based and contains the MI tasks of both feet, tongue, the right and left hands. Two sessions that are comprised of six runs were set up on different days for each subject. One run contains 48 trials (12 for each class), yielding a total of 288 trials per session. The parameters (such as the electrode number and sampling frequency) used in the dataset collected in BCI Competition 4-2a are almost the same as the parameters used in the study [41].

The ERD modulation is known to be specific at 8–12 Hz range. First, a band filter with 8–12 Hz range is applied to specific channels (such as C4, C3, CZ and C13) of the input epoch. Subsequently, moving average powers of the simple and combined MIs are computed separately by squaring each sample and smoothing the instantaneous powers by a 0.4 s (0.4 × 250 = 100 samples) sliding window with a 4 ms (1/250) shifting step (a stride of one sample). The moving average powers are normalized in order to see the relative changes between the tasks (for example LH, RH

Table 1 Success rates of the motor imagery signals with two and four classes by using DNNs (Color table online)

| methods                      | BCI Competition dataset | classes | % mean accuracy | transformation or preprocessing | number of subjects |
|------------------------------|-------------------------|---------|-----------------|---------------------------------|-------------------|
| Yang et al. [1]              | 4-2a                    | 4       | 69.27           | Filter + CSP                    | 9                 |
| Sakhavi et al. [2]           | 4-2a                    | 4       | 70.6            | Filter + CSP                    | 9                 |
| Lu et al. [3]                | 4-2b                    | 2       | 84              | FFT                            | 9                 |
| Sakhavi et al. [4]           | 4-2a                    | 4       | 74.46           | Filter + CSP                    | 9                 |
| Abbas et al. [5]             | 4-2a                    | 4       | 70.7            | Filter + CSP                    | 9                 |
| Wu et al. [6]                | 4-2b                    | 2       | 80.6            | Filter bank                     | 9                 |
| Dai et al. [7]               | 4-2b                    | 2       | 78.2            | STFT                            | 9                 |
| Tabar et al. [8]             | 4-2b                    | 2       | 77.6            | STFT                            | 9                 |
| Tang et al. [9]              | 3-3a                    | 4       | 91.9            | CSP                             | 3                 |
| Chaudhary et al. [10]        | 3-4a                    | 2       | **99.3**        | CWT                             | 5                 |
| Zhao et al. [11]             | 4-2a                    | 4       | 75              | 3D-EEG                          | 9                 |
| Zhang et al. [12]            | 4-2b                    | 2       | 82              | CSP+bispectrum                  | 9                 |
| Deng et al. [15]             | 3-3a                    | 4       | 85.3            | FBCSP                           | 3                 |
|                              | 4-2a                    | 4       | 78.9            | FBCSP                           | 9                 |
| Olivas-Padilla et al. [16]   | 4-2a                    | 4       | 78.4            | DFBCSP                          | 9                 |
| Liu et al. [17]              | 4-2a                    | 4       | 76.86           | CSP                             | 9                 |
| Echtioui et al. [18]         | 4-2a                    | 4       | 61.68           | WT+CSP                          | 9                 |
| Mahamune et al. [19]         | 4-2a                    | 4       | 71.25           | CSP                             | 9                 |
| Zaho et al. [20]             | 4-2a                    | 4       | 72.13           | WT                              | 9                 |
| Senwei et al. [21]           | 4-2a                    | 4       | 76.87           | Filter+CSP                      | 9                 |
| Olmez et al. [38]            | 4-2a                    | 4       | **79.3**        | no CSP                          | 9                 |
| Olmez et al. [38]            | 4-2b                    | 2       | **88.6**        | no CSP                          | 9                 |
| Olmez et al. [38]            | 3-4a                    | 2       | 96.2            | no CSP                          | 5                 |
| Olmez et al. [38]            | 3-3a                    | 4       | **96.5**        | no CSP                          | 3                 |

CSP: Common spatial patterns, FBCSP: Filter bank common spatial patterns, CWT: Continuous wavelet transform, FFT: Fast Fourier transform, DFBCSP: Discriminative filter bank common spatial patterns, STFT: Short-time Fourier transform
and LH-RH) in the same channel. In the BCI Competition 4 dataset, after two seconds \( (t = 2 \text{ s}) \), a cue 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 stayed on the screen for 1.25 s. In specific channels, the relative average powers are computed by dividing the moving average powers by the average power found at the 2.5th second \([41]\). Window selection of 0.4 s and moving the window by shifting one sample (4 ms) are determined by trial and error. Figure 6(a) shows the relative average powers for both hands, the left and right hands in channels C3 and C4.

As shown in Fig. 6, similar results are obtained compared with the graphs shown in the third and fourth figures of the study \([41]\) for the MIs of both hands, the left and right hands. This similarity shows that the artificially produced combined MI signal has the same characteristics with the combined MI signal acquired directly from the subjects in the study \([41]\). Therefore, the artificially produced MI signal can be used as if it were a real signal acquired from a subject. This artificially produced signal actually shows the ideal situation. It is observed that some subjects have difficulty generating even a single MI signal. Therefore, it will be even more difficult to generate two MIs at the same time for some subjects. The approach of artificially creating MI signals described in this paper can provide feedback for these subjects during experiments.

| Methods          | BCI Competition dataset | classes | % mean accuracy | Transformation + features | Classifier |
|------------------|-------------------------|---------|-----------------|---------------------------|------------|
| Ghanbar et al.   | 3-3a                    | 4       | 90.72           | CSP                       | SVM        |
| Wang et al.      | 3-3a                    | 4       | 77.2            | CSP + variance            | MLP        |
| Aljalal et al.   | 3-4a                    | 2       | 80.2            | Wavelet + statistical, entropy, energy features | MLP        |
| Miraziri et al.  | 4-2a                    | 4       | 61.7            | CSP + variance            | MLP        |
| Silva et al.     | 4-2b                    | 2       | 67.8            | Linear Predictive Coding  | MLP        |
| Alansari et al.  | 4-2b                    | 2       | 83.8            | Wavelet                   | SVM        |
| Behri et al.     | 3-4a                    | 2       | 89.4            | Wavelet                   | SVM        |
| Zhang et al.     | 3-4a                    | 2       | 94.5            |                           | K-NN       |
| Li et al.        | 4-2a                    | 3       | 84              | CSP + variance            | SVM        |
| Wang et al.      | 4-2b                    | 2       | 81.2            | FBCSP + variance          | SVM        |
| Mishuhina et al. | 3-3a                    | 4       | 89.8            | RCSP – FWR                | LDA        |
| Molla et al.     | 3-4a                    | 2       | 92.2            | CSP + subband features    | SVM        |
|                  |                         |         |                 |                           | LDA        |
With this feedback, the subject can learn to generate two motor imageries (combined MI signal) simultaneously.

Figure 7 shows relative average power at the electrodes CZ, C3 and C4 according to the simple (feet, left and right hands) and combined (right hand–left hand, right hand–feet, and left hand–feet) MI tasks. In the study [41–46], researchers have studied on the three simple MI tasks and the combined MI tasks of them. In this study, we have dealt with four simple MI tasks and their combinations. Figure 8 shows relative average power at the electrodes C4, C3, CZ and C13 according to the simple (tongue T, feet F, right hand RH and left hand LH) and combined (left hand–right hand LH-RH, left hand–feet LH-F, left hand–tongue LH-T, right hand–feet RH-F, right hand–tongue RH-T and feet–tongue F-T) MI tasks. When a subject performs a task, a decrease in the power of the signal at the electrodes close to the areas where ERD signal occurs in the brain is observed. This phenomenon provides the distinction

![Figure 7](image-url) Relative average power at the electrodes CZ, C3, and C4 (in Fig. 5) according to the simple (feet, left and right hands) and combined (right hand–left hand, right hand–feet and left hand–feet) MI tasks for the subject S3 in BCI 4 dataset 2a

![Figure 6](image-url) a ERD signals of simple and combined MI tasks according to the C3 and C4 electrodes, and b similar results obtained for simple and combined MI tasks in the study [41]
between the tasks. As shown in Figs. 6, 7 and 8, the tasks can be discriminated even by using C3 or C4 electrodes. In fact, more electrodes are used in the classification phase. Figures 6, 7 and 8 show that simple and combined MI EEG signals can easily be classified by using the DNNs. It is observed that for more MI tasks, it will be harder to distinguish the classes from each other. Moreover, more than four electrodes were used to classify MI EEG signals.

4.2 Validation on the datasets of simple and combined MI signals acquired in real time

To create the dataset [45], subjects were seated in a comfortable chair, and task which consisted of one of the eight MIs were generated with all the combinations of both feet, right and left hands together, i.e., both hands–feet, rest, feet, left hand, left hand–feet, both hands, right hand, and right hand–feet. In the dataset, there are eight classes with 40 trials performed for each class. Each task was randomly performed, and lasted for 12 s. MI EEG data were acquired at 256 Hz. The OpenViBE platform was applied for both the stimulation process and the signal acquisition. The EEG cap has 26 electrodes which were placed according to the international 10–20 system so that the primary sensorimotor cortex is covered. MI signals were processed by a band-pass filter (frequency range: 8–30 Hz) [45]. It is known that MI signals are highly dependent on the subjects. In this study, we did not deal with subject independent MI EEG signals. Therefore, each subject’s own data was used for validation. In other words, the DivFE trained with data collected from one subject was not used to classify motor imagery signals of other subjects. In the dataset [45], there are eight classes: Left hand (LH), right hand (RH), feet (F), left hand–feet (LH-F), right hand–feet (RH-F), both hands (BH), both hands–feet (HsF) and rest (R). Right hand–feet, left hand–feet and left hand–right hand combined MI EEGs are generated artificially by using the simple feet, left hand and right hand MI EEG data of [45] according to the calculation given in Eq. 4. All of the artificially generated MI EEG signals (LHF, RHF and BH) are included into the training set, and all of the real combined MI EEG signals (LHF, RHF and BH) are included into the test set. 80% of the LH, RH, F, R and HsF motor imagery signals in [45] is used in the training set, and the remaining 20% is reserved for the test set. In this paper, all the tables present the averages of 30 experiments with randomly partitioned test and training sets.

In order to compare the results obtained in our study with the results in [45], one versus all (OVA) method is preferred in this study. Therefore, eight different DivFE networks are used to determine the eight classes: right hand versus others (RH-O), left hand versus others (LH-O), feet versus others (F-O), BH versus others (BH-O), LHF versus others (LHF-O), RHF versus others (RHF-O), HsF versus others (HsF-O).

Table 3 Training and test sets for the LH-O (left hand versus others) class (Color table online)

| Classes | LH | RH | F | BH | LHF | RHF | HsF | R |
|---------|----|----|---|----|-----|-----|-----|---|
| All epochs | 40r | 32r | 40r | 40a+40r | 40a+40r | 40r | 40r | 40r |
| training set | 248u | 32r | 32r | 40a | 40a | 32r | 32r | 40r |
| test set | 8r | 8r | 8r | 8r | 8r | 8r | 8r | 8r |

40r: 40 real epochs
40a: 40 artificial epochs
248u: augmented epochs obtained from 32 real epochs of the training set

Table 4 Training and test sets for the BH-O (both hands versus others) class (Color table online)

| Classes | LH | RH | F | BH | LHF | RHF | HsF | R |
|---------|----|----|---|----|-----|-----|-----|---|
| all epochs | 40r | 40r | 40r | 240a+40r | 40a+40r | 40r | 40r | 40r |
| training set | 32r | 32r | 32r | 240a | 40a | 40a | 32r | 32r |
| test set | 8r | 8r | 8r | 8r | 8r | 8r | 8r | 8r |

40r: 40 real epochs
40a: 40 artificial epochs
240a: 240 artificial epochs
Table 5 Accuracies and Kappa for the subjects of the datasets in [45] (Color table online)

| Classes   | S1   | S2   | S3   | S4   | S5   | S6   | S7   | % Mean accuracy |
|-----------|------|------|------|------|------|------|------|-----------------|
| LF-O      | 69.2 | 86.3 | 80.0 | 78.1 | 78.1 | 78.8 | 86.3 |                |
| RH-O      | 76.7 | 86.3 | 81.9 | 73.8 | 79.4 | 83.8 | 76.9 |                |
| F-O       | 75.8 | 81.9 | 65.0 | 76.9 | 75.6 | 89.4 | 77.5 |                |
| BH-O      | 71.7 | 80.0 | 68.1 | 59.4 | 74.3 | 68.1 | 70.0 |                |
| LHF-O     | 61.7 | 74.4 | 78.1 | 68.1 | 71.9 | 60.0 | 75.6 |                |
| RHF-O     | 72.5 | 76.3 | 72.5 | 68.1 | 63.8 | 70.6 | 63.1 |                |
| HsF-O     | 81.7 | 81.3 | 62.5 | 68.8 | 76.3 | 73.8 | 80.0 |                |
| R-O       | 83.3 | 97.5 | 92.5 | 81.3 | 96.3 | 90.0 | 93.0 |                |
| Accuracy  | 74.1 | 83.0 | 75.1 | 71.8 | 76.9 | 76.8 | 77.8 | 76.5            |
| (Kappa)   | (0.482) | (0.66) | (0.502) | (0.346) | (0.538) | (0.536) | (0.556) |
| in our study |           |       |       |       |       |       |       |                 |
| Accuracy  | 33.7 | 67.5 | 46.5 | 31.8 | 35.9 | 49.3 | 58.7 | 46.2            |
| (Kappa)   | (0.35) |       |       |       |       |       |       |                 |
| by OVA in [45] |       |       |       |       |       |       |       |                 |

Table 6 Confusion matrices for the subject S2 (Color table online)

| 86.3% accuracy | LH   | O     | 86.3% accuracy | RH   | O     |
|----------------|------|-------|----------------|------|-------|
| LH             | 6    | 2     | RH             | 7    | 1     |
| O              | 20   | 132   | O              | 21   | 131   |
| 81.9% accuracy | F    | O     | 80% accuracy   | BH   | O     |
| F              | 7    | 1     | BH             | 23   | 17    |
| O              | 28   | 124   | O              | 15   | 105   |
| 74.4% accuracy | LHF  | O     | 76.3% accuracy | RHF  | O     |
| LHF            | 27   | 13    | RHF            | 28   | 12    |
| O              | 28   | 92    | O              | 26   | 94    |
| 81.3% accuracy | HsF  | O     | 97.5% accuracy | R    | O     |
| HsF            | 5    | 3     | R              | 8    | 0     |
| O              | 27   | 125   | O              | 4    | 148   |

others (HsF-O), and R versus others (R-O). Table 3 and Table 4 shows the number of epochs corresponding to the classes in the test and training sets for LH-O and BH-O classes, respectively; there are 40 trials for each class in the dataset introduced in [45]. In the LH-O experiment, the O class (blue colored) consists of RH, F, BH, LHF, RHF, HsF and R classes. Due to the use of OVA method, the number of epochs for each class in the training set is not balanced. The augmentation process is used to eliminate the imbalance within the training set. For example, in Table 3, the LH class with 248 epochs is formed using randomly selected replicates from 32 epochs belonging to LH class. In the other OVA experiments conducted in this study, the same method is used for the augmentation.

Table 5 shows the classification performances obtained for all subjects of the dataset in [45]. In Table 5, success rates are defined as the averages of accuracies obtained for all classes; it can be observed that high success rates have been achieved for each subject. Table 6 presents the confusion matrices obtained for the subject S2. While obtaining the results in Table 5, it is aimed to keep the sensitivities high. In this respect, Table 6 reveals our concern about achieving high sensitivity values.

A DivFE network of similar structure was used for each subject. The input signal size is selected as $1536 \times 26$ (input epoch with 26 channels for 6 s). In the DivFE architecture, the feature extractor has eleven convolutional layers and a single dense layer. The size of the filters is 45.
in each layer, and there are 70 feature maps in each convolution layer. In these experiments, max-pooling process is not used. The single dense layer has sixteen output nodes due to using sixteen-dimensional Walsh vectors for the training; so, it is composed of $70 \times 1536 \times 16$ weights. The number of weights of the DivFE is calculated as follows:

$$26 \times 45 \times 70 + (70 \times 45 \times 70) \times 10 + 70 \times 1536 \times 16 = 4,007,220.$$  

The paired t test is applied to the results to show that the achievements in Table 5 are not random. For this purpose, two sets are formed with the sensitivities of the results achieved from the seven subjects for randomly generated epochs (RGE) and artificially generated epochs (AGE) (by using Eq. 4), respectively. In these experiments, only BH-O, LHF-O and RHF-O sensitivity values obtained for each subject are used. Therefore, the first set is created with a total of 21 data (three data from each subject) by using the sensitivities of the results in Table 5. In order to form the second set, a different experiment is created. In this experiment, epochs with random values are replaced with the artificial epochs (240a and 40a) in the training set, while the test set remains unchanged as shown in Table 4, and DivFEs are trained for each subject individually. The sensitivities corresponding to the BH, LHF and RHF classes for seven subjects are used to form the second set. The violin plot shown in Fig. 9 is used to investigate visually the effect of the artificially generated epochs on classification performances. The paired t test revealed that the success rates achieved via artificially generated epochs are statistically significantly higher ($p$-val: $1.82e-18$) than the success rates achieved via randomly generated epochs.

### 4.3 Classification of the simple and combined MI EEG signals of BCI 4-2a and BCI 3-3a

Table 7 presents the classification results (sensitivities and mean accuracy) of the DivFE for the ten-class MI signals generated from BCI 4-2a [37]. The BCI Competition 4 (2008) dataset 2a contains epochs belonging to four classes from nine subjects. Each epoch contains 22-channel MI EEG signals. There are two sessions for each subject and each session contains 288 epochs. Table 8 shows the DivFE structures specifically found to provide the success rates in Table 7. In order to obtain these achievements, max-pooling, batch normalization and ReLU are used at the output of each convolution layer. However, the success achieved for 10 classes is slightly lower than that of for four classes as presented in Table 1. It is known that as the number of classes increases, the classification performances decrease. Tables 7 and 8 show that high classification performances are achieved by using DivFEs with a small number of weights for dataset BCI 4-2a. In the DivFE architecture, the feature extractor has seven convolutional layers and a single dense layer.
layer. Moreover, in Table 10, the structures of the networks for the three subjects are similar with each other.

Table 11 presents the success rates of the simple and combined MI EEGs with more than three classes. A different dataset was used in each study shown in Table 11. In the literature, it is observed that increasing the number of classes leads to a decrease in the success rates. Likewise, the successes of simple and combined MI EEG tasks are observed to be low. In our study, the DivFE is applied in the classification of simple and combined motor imagery signals. The use of the DNN is one of the reasons why the performance is higher than the other studies.

5 Discussion and conclusions

In most studies in the literature, the success rate of the CNN for MI signals with small datasets is increased by using transformation techniques such as the CSP [1, 2, 4, 5, 9, 12, 19, 21, 34], fast Fourier transform [3], short-time Fourier transform [7, 8], wavelet transform [10, 27, 28, 35] and the other methods [6, 11]. The success rates for two-class EEG problems are observed to be higher than those of four-class EEG problems. Hence the transformation process becomes a must for the classification of four-class problems in order to enhance the performance. The CSP is the most utilized raw data transformation technique when the classification of MI signals is of concern. The CSP and similar techniques are preprocessing steps which are used as a guide for the feature extraction phase of DNNs. However, the CSP’s $m$ parameter affects the CSP’s performance and also the overall success rates [13, 14]. In a previous study [38], researchers have preferred the augmentation process instead of using the CSP in the classification of motor imageries. In [38], it has been shown that by resulting in a high classification performance for the motor imagery EEG, the augmentation was able to compete with the CSP, and there was no need to apply the augmentation process to the training set. In this study, the datasets automatically grow while generating combined EEG signals. For instance, in order to generate the combined MI signal corresponding to the right hand–left hand, the right hand MI signal is simply added to the left hand MI signal. In this way, the number of combined samples is equal to the number $(M)$ of the left (or right) hand samples. Moreover, the number of samples in the combined MI EEG dataset can be increased up to $M^2$.

The CSP was used as a preprocessing step in the studies [41–46] that classify combined MI EEG signals. In the studies [41, 43, 45], it is observed that the CSP is used with the one-versus-rest method to increase the CSP’s performance. In fact, the use of the CSP with one-versus-rest strategy in the classification of simple MI signals increases the classification performance. However, this strategy did not improve the classification performance when using combined MI EEG signals [41, 43, 45]. The sources of the

| Classes | % Sensitivities achieved for each subject |
|---------|-----------------------------------------|
|         | S1   | S2   | S3   | S4   | S5   | S6   | S7   | S8   | S9   |
| LH      | 100  | 92.6 | 97.7 | 91.7 | 100  | 90.5 | 89.8 | 92.3 | 91.2 |
| RH      | 88.9 | 44.4 | 82.7 | 75   | 69.2 | 81   | 85.2 | 57.7 | 70.8 |
| F       | 81.5 | 70.4 | 60.2 | 66.7 | 69.2 | 47.6 | 70.4 | 80.8 | 66.7 |
| T       | 63   | 81.5 | 67.7 | 70.8 | 76.9 | 76.1 | 66.7 | 42.3 | 62.5 |
| LH-RH   | 88.9 | 88.9 | 94   | 95.8 | 96.2 | 100  | 88.9 | 96.2 | 100  |
| LH-F    | 59.3 | 70.4 | 52.6 | 79.2 | 69.2 | 66.7 | 85.2 | 73.1 | 62.5 |
| LH-T    | 44.4 | 63   | 71.4 | 66.7 | 76.9 | 81   | 74.1 | 69.2 | 54.2 |
| RH-F    | 85.2 | 100  | 86.5 | 95.8 | 88.5 | 95.2 | 92.6 | 96.2 | 100  |
| RH-T    | 66.7 | 66.7 | 67.7 | 37.5 | 84.6 | 66.7 | 74.1 | 50   | 75   |
| F-T     | 92.6 | 92.6 | 97.7 | 91.7 | 100  | 85.2 | 100  | 100  | 100  |
| % Accuracy (Kappa) | 77 | 77.1 | 77.8 | 77.1 | 83   | 80.5 | 81.1 | 75.8 | 78.3 | 78.6 (0.715) |

LH: left hand, LH-RH: left hand–right hand, RH-F: right hand–feet, T: tongue
RH: right hand, LH-F: left hand–feet, RH-T: right hand–tongue
F: feet, LH-T: left hand–tongue, F-T: feet–tongue
simple MI EEG signals are located in different areas of the brain, and each simple MI EEG signal is generated from only one source. In this case, the CSP easily reveals each simple MI signal. In the case of using combined MI EEG signals, the combined MI EEG signals are generated from two sources. The low classification performances in the studies [41, 43, 45] are suspected to be stemmed from the use of the CSP.

Due to the difficulty of acquiring EEG signals in real time, studies are generally performed on small datasets. In this context, it can also be observed that some studies have been conducted on artificially generated EEG datasets in recent years [47-49]. In image processing, the generative adversarial networks (GANs) with successful applications have gained significant attention [47]. GANs may be very useful for BCI systems where acquiring large number of samples is time-consuming and expensive. In [48], the BCI Competition dataset 4-2b was used in the training of the GAN. It is observed that GANs may reveal important characteristics of MI signals, such as variations of power in the beta-band. In [49], researchers stated that the success of classifying the MI signals is related to the size of datasets used. For this reason, the training set was enriched by generating artificial MI signals to save time. This study showed that it is possible to replace up to 50% of training sets with artificially created EEG signals; and 64% success rate is obtained using a dataset formed with artificial epochs.

In [41], it is assumed that the brain activity created for the combined MI signal of both hands corresponds to the superposition of brain activities created for the simple MI signals of each hand. In [43], researchers confirmed that this simplification is convenient, and representing a multi label task as the superposition of sources used for simple tasks takes us to a reasonable model. Researchers made an assumption that combined MI could be represented as the superposition of brain activities created by each one of the used body parts. Also, in the study [46], authors imply that there are differences between the simple limb MI and combined limb MI, which may be used in the classification of MIs. The approach of the artificially generating combined MI EEG signals is validated on a dataset [45], and high classification results are achieved by using these signals in the training set. In our study, the main reason why the performance is very high compared to the results in [45] is the use of DNNs and especially the DivFE. The DivFE has previously been shown to have high success rates in classifying MI EEG signals [38, 39]. When the DNN continues learning the training set by more than 90%, the average accuracy for the validation set also continues to increase; however, it was observed that there was a decrease in the sensitivity. Therefore, the training process should be terminated to prevent over-fitting when the training accuracy rises above 90%.

It is observed in [41] that the artificially generated combined MI signal has the same characteristics as the combined MI signal acquired from the subjects. Therefore, the artificially generated MI signal can be used as if it were a real signal acquired from a subject. In fact, artificially generated signals demonstrate the ideal situation. However, it is possible that ERD signals for the combined MI do not occur at the same time as those for the artificially generated signal. This situation can be a problem with conventional classifiers. The fact that two ERD signals that occur at different times in an epoch may not present a disadvantage for DNNs.

To increase the number of BCI commands, mental tasks can be used in addition to the motor imageries. However, mental tasks reduce the classification performance. In the proposed method, subjects are allowed to perform more
than two simple motor imageries simultaneously. Although it seems theoretically possible, it is quite challenging in practice for the subjects to simultaneously produce more than two simple motor imageries. Instead, it would be more efficient to increase the number of simple motor imagery tasks. For example, using six simple motor imagery tasks, \(C(6,2) = 15\) different combined motor imagery tasks can be created. Thus, a total of 21 (simple + combined MI tasks) BCI commands are generated. In Fig. 1, the areas of the motor imagery tasks associated with the limbs are shown. After rigorous training, the left arm and right arm motor imageries might be used as the fifth and the sixth simple motor imagery tasks. As shown in Fig. 1, brain regions of the two new tasks are far enough from those of the other four simple motor imagery tasks. In that case, it would be necessary to improve the spatial resolution by increasing the number of electrodes.

It is well-known that some subjects have difficulties in generating even simple distinguishable MI signals. Therefore, it will be even more difficult to generate combined MIs for those subjects. Artificially generated combined MI signals can provide feedback for these subjects during the experiments. The experimental setup can be designed as follows: In the beginning, the subjects are trained with four simple MI tasks. Then, combined MI signals are artificially generated. In the third phase, DNNs are trained with a dataset created from simple and combined MI signals. Finally, the trained DNNs are used for feedback purposes.

Table 9  Success rates of the DivFE for ten-class MI signals produced from BCI 3-3a (Color table online)

| Classes   | k3b | k6b | l1b |
|-----------|-----|-----|-----|
| LH        | 87.5| 100 | 100 |
| RH        | 67.3| 75  | 87.5|
| F         | 87.5| 50  | 100 |
| T         | 74  | 75  | 62.5|
| LH-RH     | 87.5| 100 | 100 |
| LH-F      | 74  | 75  | 87.5|
| LH-T      | 67.3| 87.5| 50  |
| RH-F      | 100 | 100 | 100 |
| RH-T      | 87.5| 87.5| 100 |
| F-T       | 80.8| 100 | 100 |
| % Accuracy| 81.3| 85  | 88.8|
|           |     |     | 85  |
|           |     |     | (0.8)|

Table 10  Structures of the DivFE which provide the accuracies in Table 8

| Layers of FE | Number of inputs, size of filters, number of outputs in DivFE |
|--------------|-------------------------------------------------------------|
| k3b          |                                                             |
| Layer 1      | 43,15,70                                                     |
| Layer 2      | 70,13,70                                                     |
| Layer 3      | 70,11,70                                                     |
| Layer 4      | 70,9,70                                                      |
| Layer 5      | 70,7,70                                                      |
| Layer 6      | 70,5,70                                                      |
| Layer 7      | 70,3,70                                                      |
| FL           | 70 \times 5,16                                               |
| #weights     | 285,950                                                      |
| k6b          |                                                             |
| Layer 1      | 43,15,70                                                     |
| Layer 2      | 70,13,70                                                     |
| Layer 3      | 70,11,70                                                     |
| Layer 4      | 70,9,70                                                      |
| Layer 5      | 70,7,70                                                      |
| Layer 6      | 70,5,70                                                      |
| Layer 7      | 70,3,70                                                      |
| FL           | 70 \times 5,16                                               |
| #weights     | 285,950                                                      |
| l1b          |                                                             |
| Layer 1      | 43,15,70                                                     |
| Layer 2      | 70,13,70                                                     |
| Layer 3      | 70,11,70                                                     |
| Layer 4      | 70,9,70                                                      |
| Layer 5      | 70,7,70                                                      |
| Layer 6      | 70,5,70                                                      |
| Layer 7      | 70,3,70                                                      |
| FL           | 70 \times 5,16                                               |
| #weights     | 156,250                                                      |

FL: flatten layer which does not contain filters
so that the subjects can generate their own combined MI signals. After the temporary training phase is complete, subjects can be ready to create the datasets.

Acknowledgements The work in the article is supported by the Istanbul Technical University Scientific Research Project Unit [ITU-BAP MYL-2019-41895 and ITU-BAP MYL-2018-41621].

Declarations

Conflict of interest The authors declare that there is no conflict of interests regarding the publication of this paper.

References

1. Yang H, Sakhavi S, Ang KK, Guan C (2015) On the use of convolutional neural networks and augmented CSP features for multi-class motor imagery of EEG signals classification. 37th Annu Int Confere IEEE Eng Med and Biol Soc. https://doi.org/10.1109/EMBC.2015.7318929
2. Sakhavi S, Guan C, Yan S (2015) Parallel convolutional-linear neural network for motor imagery classification. 23rd Europ Sig Process Conf. https://doi.org/10.1109/EUSIPCO.2015.7362882
3. Lu N, Li T, Ren X, Miao H (2017) A deep learning scheme for motor imagery classification based on restricted Boltzmann machines. IEEE Trans Neural Syst Rehabil Eng 25(6):567–576. https://doi.org/10.1109/TNSRE.2016.2601240
4. Sakhavi S, Guan C, Yan S (2018) Learning temporal information for brain-computer interface using convolutional neural networks. IEEE Transact on Neural Netw Learn Syst 29(11):5619–5629. https://doi.org/10.1109/TNNLS.2018.2789927
5. Abbas W, Khan NA (2018) DeepMI: Deep learning for multiclass motor imagery classification. 40th Annu Inter Conferen IEEE Eng Med Biol Soc. https://doi.org/10.1109/EMBC.2018.8512271
6. Wu YT, Huang TH, Lin YC, et al. (2018) Classification of EEG motor imagery using support vector machine and convolutional neural network. International Automatic Control Conference – CACS. https://doi.org/10.1109/CACS.2018.8606765
7. Dai M, Zheng D, Na R et al (2019) EEG Classification of motor imagery using a novel deep learning framework. Sensors 19(3):551. https://doi.org/10.3390/s19030551
8. Tabar YR, Halici U (2017) A novel deep learning approach for classification of EEG motor imagery signals. J Neural Eng 14(1):016003. https://doi.org/10.1088/1741–2560/14/1/016003
9. Tang X, Zhao J, Fu W (2019) Research on extraction and classification of EEG features for multi-class motor imagery. IEEE 4th Adv Inform Technol, Electron and Automation Control Conf. https://doi.org/10.1109/IAEAC47372.2019.8998049
10. Chaudhary S, Taran S, Bajaj V, Sengur A (2019) Convolutional neural network based approach towards motor imagery tasks EEG signals classification. IEEE Sens J 19(12):4494–4500. https://doi.org/10.1109/JSEN.2019.2899645
11. ZhaoX,ZhangH,ZhuG,YouF,KuangS,SunL(2019)Amulti-branch 3D convolutional neural network for EEG-based motor imagery classification. IEEE Trans Neural Syst Rehabil Eng 27(10):2164–2177. https://doi.org/10.1109/TNSRE.2019.2938295
12. Zhang R, Zong Q, Zhao X (2019) A new convolutional neural network for motor imagery classification. Proceedings of the 38th Chinese Control Conference. https://doi.org/10.23919/ChiCC.2019.8865152
13. Yüksel A (2017) Classification methods for motor imagery based brain computer interfaces. PhD Dissertations. Istanbul Technical University. Institute of Science and Technology.
14. Yüksel A, Olmez T (2015) A neural network based optimal spatial filter design method for motor imagery classification. PLOS-ONE 10(5):e0125039. https://doi.org/10.1371/journal.pone.0125039
15. Deng X, Zhang B, Yu n, Liu K and Sun K, (2021) Advanced TSGL-EEGNet for Motor Imagery EEG-Based Brain-Computer Interfaces. IEEE Access 9:25118–25130. https://doi.org/10.1109/ACCESS.2021.3056088
16. Olivas-Padilla BE, Chacon-Murguia MI (2019) Classification of multiple motor imagery using deep convolutional neural networks and spatial filters. Appl Soft Comput 75:461–472. https://doi.org/10.1016/j.asoc.2018.11.031
17. Liu M, Zhou M, Zhang T, Xiong N (2020) Semi-supervised learning quantization algorithm with deep features for motor imagery EEG recognition in smart healthcare application. App Soft Comp 89:106071

Table 11 Success rates for simple and combined MI signals (Color table online)

| Methods | type of MI | Mean accuracy % | Feature extraction + classifier | Number of subjects |
|---------|------------|-----------------|-------------------------------|-------------------|
| Leon et al. [41] | 3 simple + 1 combined EEG signals (classes) | 51.6 | CSP + LDA | 6 |
| Yijie et al. [43] | 3 simple + 1 combined | 54.2 | CSP + SVM | 8 |
| For MC2CMI in Leon’s study [45] | 4 simple + 4 combined | 55 | CSP + LDA | 7 |
| Yi et al. [46] | 4 simple + 3 combined | 70 | CSP + SVM | 10 |
| For dataset 3-3a in our study | 4 simple + 6 combined | 85 | only DivFE | 3 |
| For dataset 4-2a in our study | 4 simple + 6 combined | 78.6 | only DivFE | 9 |
| For datasets of [45] in our study | 4 simple + 6 combined | 77.8 | only DivFE | 7 |
18. Echtioui A, Zouch W, Ghorbel M, Mhiri C, Hamam H (2021) Fusion Convolutional Neural Network for Multi-Class Motor Imagery of EEG Signals Classification. International Wireless Communication Mobile Comput (IWCMC). https://doi.org/10.1109/IWCMC51323.2021.9498885

19. Mahamune R, Laskar SH (2021) Classification of the four-class motor imagery signals using continuous wavelet transform filter bank-based two dimensional images. Int J Imaging Syst Technol 31:2237–2248. https://doi.org/10.1002/ima.22593

20. Echtioui A, Zouch W (2021) Multi-class Motor Imagery EEG Classification using Convolution Neural Network. In Proceedings of the International Conference on Agents and Artificial Intelligence (ICAART). https://doi.org/10.5220/0010425905910595

21. ZhaoX, LiuD, KongW, PengY, HuH, CaoJ (2022) Anovelclassification of the four-class motor imagery EEG classification. Biomed Signal Process Control 72:103338. https://doi.org/10.1016/j.bspc.2021.103338

22. XuS, ZhuL, KongW, PengY, HuH, CaoJ (2022) Anovelclassification method for EEG-based motor imagery with narrow bands spatial filters and deep convolutional neural network. CognNeurodyn16:379–389. https://doi.org/10.1007/s11571-021-09721-x

23. Wang L, Wu XP (2008) Classification of four-class motor imagery EEG data using spatial filtering. 2nd International Conference on Bioinformatics and Biomedical Engineering. https://doi.org/10.1109/ICBBE.2008.868

24. Aljalal M, Djemal R (2017) A comparative study of wavelet and CSP features classified using LDA, SVM and ANN in EEG based motor imagery. 9thIEEE-GCC Conference and Exhibition. https://doi.org/10.1109/IEEEGCC.2017.848212

25. Mirnaziri M, Rahimi M, Alavikakhaki S, Ebrahimpour R (2013) Using combination of μ, β and γ bands in classification of EEG signals. Basic Clin Neurosci 4(1):76–87

26. Silva VF, Barbosa RM, Vieira PM, Lima CS (2017) Ensemble learning based classification for BCI applications. IEEE 5th Portuguese Meeting on Bioengineering. https://doi.org/10.1109/ENBENG.2017.7889483

27. Alansari M, Kamel M, Hakim B, Kadah Y (2018) Study of wavelet-based performance enhancement for motor imagery brain-computer interface. 6th Inter Conference Brain-Compute Interface (BCI). https://doi.org/10.1109/IWW-BCI.2018.8311520

28. Behri M, Subasi A, Qaisar SM (2018) Comparison of machine learning methods for two class motor imagery tasks using EEG in brain-computer interface. Adv Sci and Eng Technol Inter Conf (ASET). https://doi.org/10.1109/ICASET.2018.8376885

29. Zhang Y, Liu J (2018) EEG recognition of motor imagery based on SVM ensemble. 5th Int Conf on Systems and Informatics. https://doi.org/10.1109/ICSAI.2018.8599464

30. Li B, Yang B, Guan C, Hu C (2019) Three-class motor imagery classification based on FBSCSP combined with voting mechanism. IEEE Inter Conference on Comput Intelligence Virtual Environ Measurement Syst Appl. 8:103733–103740. https://doi.org/10.1109/IEEEGCC.2017.848212

31. Wang J, Feng Z, Lu N (2017) Feature extraction by common spatial pattern in frequency domain for motor imagery tasks classification. 29th Chinese Control and Decision Conference (CCDC). https://doi.org/10.1109/CCDC.2017.7978220

32. Mishuhina V, Jiang X (2018) Feature weighting and regularization of common spatial patterns in EEG-based motor imagery BCI. IEEE Signal Process Lett 25(6):783–787. https://doi.org/10.1109/LSP.2018.2823683

33. Molla KL, Shiam AA, Islam R, Tanaka T (2020) Discriminativefeature selection-based motor imagery classification using EEG signal. IEEE Access. https://doi.org/10.1109/ACCESS.2020.2996685

34. Jin Z, Zhou G, Gao D, Zhang Y (2020) EEG classification using sparse Bayesian extreme learning machine for brain-computer interface. Neural Comput Appl 32:6601–6609. https://doi.org/10.1007/s00521-018-3735-3

35. Ahangi A, Karamnejad M, Mohammadi N, Ebrahimpour R, Bagheri N (2013) Multiple classifier system for EEG signal classification with application to brain–computer interfaces. Neural Comput Appl 23:1319–1327. https://doi.org/10.1007/s00521-012-1074-3

36. BCI Competitions 3 (2005), http://www.bbci.de/competition/iii/

37. BCI Competitions 4 (2008), http://www.bbci.de/competition/iv/

38. Olmez T, Dokur Z (2021) Strengthening the training of convolutional neural networks by using Walsh matrix. arXiv:2104.00035.

39. Dokur Z, Olmez T (2020) Heartbeat classification by using a convolutional neural network trained with Walsh functions. Neural Comput Appl 32(16):12515–12534. https://doi.org/10.1007/s00521-020-04709-w

40. Phang CR, Ko LW (2020) Global cortical network distinguishes motor imagination of the left and right foot. IEEE Access 8:103734–103745. https://doi.org/10.1109/ACCESS.2020.2999133

41. Leon LC, Bourgain L (2015) A multi-label classification method for detection of combined motor imageries. IEEE International Conference on Systems, Man, and Cybernetics. https://doi.org/10.1109/SMC.2015.543

42. Yi W, Qiu S et al (2016) EEG oscillatory patterns and classification of sequential compound limb motor imagery. Jour of Neuro Engineering and Rehabilitation 13:11. https://doi.org/10.1186/s12984-016-0199-8

43. Yijie Z, Bin G et al (2018) A multiuser collaborative strategy for MI-BCI system. 23rd IEEE Internat Conf Digital Signal Proces. https://doi.org/10.1109/ICDSP.2018.8631864

44. Chen Z, Wang Z, Wang K, Yi W, Qi H (2019) Recognizing motor imagery between hand and forearm in the same limb in a hybrid brain computer interface paradigm: An online study. IEEE Access 7:59631–59639. https://doi.org/10.1109/ACCESS.2019.2915614

45. León CL, Rimbert S, Bougrain L (2020) Multiclass classification based on combined motor imageries. Front Neurosci 14:1–14. https://doi.org/10.3389/fnins.2020.559858

46. Yi W, Qiu S, Qi H, Zhang L, Wan B, Ming D (2013) EEG feature comparison and classification of simple and compound limb motor imagery. Journal of Neuro Engineering and Rehabilitation 10:106. https://doi.org/10.1186/1743-0003-10-106

47. Fahimi F, Zhang Z, Goh WB, Ang KK, Guan C (2019) Towards EEG Generation Using GANs for BCI Applications. IEEE EMBS Inter Conf on Biomed Health Inform (BHI). https://doi.org/10.1109/BHI.2019.8834503

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.