Future Generation Computer Systems 101 (2019) 542–554

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

# Deep Learning for EEG motor imagery classification based on multi-layer CNNs feature fusion

Syed Umar Amin [a,b], Mansour Alsulaiman [a,b], Ghulam Muhammad [a,b,*], Mohamed Amine Mekhtiche [a,b], M. Shamim Hossain [c]

[a] Department of Computer Engineering, College of Computer and Information Sciences (CCIS), King Saud University, Riyadh 11543, Saudi Arabia
[b] Center of Smart Robotics Research, CCIS, King Saud University, Riyadh, Saudi Arabia
[c] Department of Software Engineering, College of Computer and Information Sciences (CCIS), King Saud University, Riyadh 11543, Saudi Arabia

## HIGHLIGHTS

- Multi CNN models with different layers and filters for robust EEG feature extraction.
- Fusion model for merging multiple CNNs for EEG classification.
- Use of transfer learning and pretraining to further improve EEG decoding accuracy.
- Autoencoders cross-subject feature reconstruction to achieve better results.

## ARTICLE INFO

*Article history:*
Received 4 March 2019
Received in revised form 28 May 2019
Accepted 19 June 2019
Available online 3 July 2019

*Keywords:*
EEG motor imagery classification
Deep learning
Convolution neural network
Multi-layer CNNs feature fusion

## ABSTRACT

Electroencephalography (EEG) motor imagery (MI) signals have recently gained a lot of attention as these signals encode a person's intent of performing an action. Researchers have used MI signals to help disabled persons, control devices such as wheelchairs and even for autonomous driving. Hence decoding these signals accurately is important for a Brain–Computer interface (BCI) system. But EEG decoding is a challenging task because of its complexity, dynamic nature and low signal to noise ratio. Convolution neural network (CNN) has shown that it can extract spatial and temporal features from EEG, but in order to learn the dynamic correlations present in MI signals, we need improved CNN models. CNN can extract good features with both shallow and deep models pointing to the fact that, at different levels relevant features can be extracted. Fusion of multiple CNN models has not been experimented for EEG data. In this work, we propose a multi-layer CNNs method for fusing CNNs with different characteristics and architectures to improve EEG MI classification accuracy. Our method utilizes different convolutional features to capture spatial and temporal features from raw EEG data. We demonstrate that our novel MCNN and CCNN fusion methods outperforms all the state-of-the-art machine learning and deep learning techniques for EEG classification. We have performed various experiments to evaluate the performance of the proposed CNN fusion method on public datasets. The proposed MCNN method achieves 75.7% and 95.4% on the BCI Competition IV-2a dataset and the High Gamma Dataset respectively. The proposed CCNN method based on autoencoder cross-encoding achieves more than 10% improvement for cross-subject EEG classification.



## 1. Introduction

Brain–computer interfaces (BCI) [1–3] provide an alternate mode of communication between the human brain and external devices [4]. The non-invasive scalp Electroencephalograph (EEG) [1] is an easy and inexpensive technique for recording brain activity. The EEG signal is recorded using multiple electrodes placed on the specific scalp areas. EEG signals have a characteristic high temporal resolution, up to the milliseconds-range, which is still not possible with the latest imaging techniques such as computed tomography (CT) or magnetic resonance imaging (MRI). These properties make EEG an important tool for research and diagnosis related to brain functioning and disorders.

Among the different types EEG signals, motor imagery (MI) signals [5,6], have recently attracted a lot of research interest, as it is quite flexible EEG technique through which we can discriminate various brain activations. MI EEG signals are brain activity

* Corresponding author at: Department of Computer Engineering, College of Computer and Information Sciences (CCIS), King Saud University, Riyadh 11543, Saudi Arabia.
*E-mail address:* ghulam@ksu.edu.sa (G. Muhammad).

https://doi.org/10.1016/j.future.2019.06.027
0167-739X/

recorded when the subject imagines or intends to perform actions like hand or leg movement. The MI EEG signal is produced in the brain's sensorimotor cortex area as a response to these imagining or thinking tasks [2]. These MI signals have been utilized by researchers to discriminate between different oscillatory brain activations for different tasks. Automated MI classification [4,5] has been performed using different machine learning and deep learning techniques.

In the past, researchers have employed conventional techniques based on machine learning methods which used handcrafted features for classifying EEG data. MI EEG based BCI have used machine learning to build systems to help strokes and epilepsy patients to communicate [7], to control external devices like wheelchairs and robots [6] etc. Besides these applications, EEG data is also being used recently to impart human cognitive behavior and artificial intelligence to systems using deep learning [8]. But as MI has limited spatial resolution, low signal-to-noise ratio (SNR) and highly dynamic characteristics, the extraction of robust features is a crucial step in developing a successful BCI system. Due to these issues and the presence of large amounts of noise in the EEG data, it is a very challenging task to analyze brain dynamics and classify EEG data. Although the conventional machine learning methods have been successful up to a certain extent in classifying the EEG MI data, they have not been able to reach good decoding accuracies with handcrafted features. The recent success of deep learning methods has driven researchers to apply them for EEG classification, and deep learning has proven that automated feature extraction can reach better performance. Various types and architectures of deep learning have achieved state-of-the-art results for different areas like image [9], speech classification [10], forgery detection [9,11]. Convolution neural networks (CNN) have the ability to find robust spatial features from images [12]. Recurrent neural networks (RNN), can extract temporal features better than other models for applications like video and speech classification [13]. There are models like autoencoders, which suit unsupervised feature learning [14].

Some recent studies which have employed different deep learning techniques for automated feature extraction from EEG data [15]. Training deep learning models on small datasets is a difficult task as they may have millions of parameters which typically requires huge training data. Not many EEG public datasets are available and those available have a limited size. Due to this reason, we have had limited applications and research on deep networks in this area. However, we now have techniques like transfer learning, which have given researchers a way to use deep networks pretrained on large datasets and then fine-tune them for smaller datasets. These techniques have been able to increase performance and reduce training time for deep models [12]. Very few studies have exploited them for EEG data, such as [16], where researchers used deep belief networks (DBN) and CNN with transfer learning for EEG and fMRI datasets having comparatively limited training samples. Hence, deep learning models pretrained on similar EEG datasets could help increase EEG decoding performance. The increase in accuracy achieved by using deep learning models for fields like image or speech processing is not evident in the case of EEG, so we need more research in this area.

Many variations of CNN models have been used for image classification with good effect. One of them is fusing multiple CNN streams for feature aggregation, which has led to improved accuracy. Different CNN models can specialize in extracting various spatial and temporal features, hence the architecture and depth of CNN affect its performance and accuracy. Many studies that have exploited feature fusion and multiple CNN models [17–23] for extracting intermediate features and fusing models with different architectures and have been quite successful. During the course of training, different convolution layers can extract features at different levels of abstraction. Initial layers learn local features and end layers learn global features. CNN with different depths and different size of filters are able to extract different features. The EEG signal is time-series data having multiple channels, low SNR, and is difficult to interpret because of its non-stationarity. Although researchers have applied CNN and other deep learning models for EEG MI data to achieve good results they have not been able to achieve major improvements over machine learning techniques [24–26].

In this study, we build a fusion method by merging multiple CNN models. Each CNN has a different depth and filter size. Feature fusion and multiple CNN architectures have not been explored for EEG data. Our method reports performance improvement for EEG MI data, showing that convolution features depend on the depth of CNN and filter size. These features can be combined to develop a robust EEG classification system. We also use transfer learning and pretraining to further improve EEG decoding accuracy.

The rest of the paper is organized as follows. In Section 2, we provide the literature review related to the EEG MI classification using machine learning and deep learning techniques and CNN based fusion methods. In Section 3 we present our proposed multi-layer CNNs fusion method and in Section 4 we present the experimental results. At last, we conclude in Section 5.

## 2. Related study

This section discusses machine learning and deep learning based methods which have been used for EEG MI classification. Some studies which have used multi-layer CNNs features and fusion methods are also discussed in this section.

### 2.1. Motor imagery classification

Many conventional machine learning based methods have been developed for MI decoding and feature extraction. Among these methods filter bank common spatial patterns (FBCSP) based on common spatial patterns (CSP) features [4,5] has achieved the best performance. All these methods use handcrafted features. Support vector machine (SVM) has been used by many researchers as a classifier [27–29]. Independent component analysis (ICA) and principal component analysis (PCA) have been used for dimensionality reduction and noise removal [11,30,31].

Recently, deep learning techniques like CNN, DBN, and restricted Boltzmann machine (RBM) have been shown to reach competitive accuracies for EEG MI decoding using the automated feature extraction. Researchers in [16] have used multiple RBM to extract robust features for MI dataset. CNN has been a popular choice for analyzing spatial features and classifying EEG signals [28,29,32,33]. DBN is applied to extract temporal features as EEG are time-series signals [34–36]. Some researchers have combined CNN with RNN to extract both spatial and temporal features [28,29]. Another study used DBN and SVM [34] for MI classification. CNN features were combined with augmented CSP features to give good accuracies [35]. CNN was used with RNN in [36], to extract multidimensional features for capturing cognitive events from MI signals. CNN and autoencoders were used in a study for emotion recognition using EEG signals [37]. CNN has also been used for classification of EEG images which were formed by transforming 1D EEG signals into 2D. This study proposed a new set of features by combining spatial, spectral, and temporal information in EEG data. Another study [33], also converted the EEG signal into images by utilizing short-time Fourier transform (STFT). The mu and beta band features were

utilized by some researchers using CNN and stacked autoencoder (SAE) for MI classification [38,39].

The studies mentioned above have used various conventional machine learning and deep learning methods for MI classification and decoding. Although deep learning methods like CNN have recently achieved state-of-the-art results for this task but have not been able to provide a substantial improvement in accuracy as they had achieved for image and speech processing. With the maximum accuracies still less than 75% on public MI datasets, we need to apply new inventions related to CNN and other deep learning methods for further improvement.

### 2.2. Feature fusion using CNN

Various CNN models and techniques have been applied in different fields to improve performance. CNN can extract rich features automatically as the convolution process takes place layer by layer. Initial layers extract local and spatial features and end layers specialize in extracting global features. Lower level features can be simple shapes like edges, boundaries, and high-level features represent complex shapes and complete objects.

Many researchers have extracted convolutional features from different layers and fused them to improve performance, showing that at each level convolutional features accomplish different abstractions of object features. Some of the relevant features that are lost in the layered architecture can be utilized to make the final feature rich. Though the approach consumes more resources and includes redundant features, it improves performance. Different techniques proposed by researchers for extracting and fusing multilevel convolutional features for different domains. Some of them are discussed here. The authors in [20] extracted multilayer features from different convolution layers for salient object detection. They fused these features for each resolution and created multiresolution images. These combined features were used to detect salient objects. Low-level features were extracted from initial CNN layers to form resolution maps which could detect an object's edges and boundaries. When combined, the multiresolution images could detect a complete object. Another study integrated convolutional features extracted from middle CNN layers and dense layers [21]. They used a method called hyper column for feature aggregation. The authors in [22], fused multilevel CNN features extracted from videos, to classify human action. They developed a multi-stream CNN model and extracted spatial and temporal features at multiple layers. At first, spatial and temporal feature streams are locally fused to detect actions from videos. In the second step, the combined features are used as input to a stacked long short-term memory (LSTM) model which finds features at the global context.

A music input tagging system was proposed in [17] which used multiple pretrained CNN's with feature fusion. Multilevel and multiscale feature were extracted both locally and globally. Pretrained CNN was used to capture different time scale audio features from all its layers. Then these local audio features were combined to get global audio features. The aggregated features were passed on to the classifier for music tagging.

A study used a bag-of-features method for CNN feature fusion and proposed discriminative features to improve the performance of CNN image classification [23]. In [18], the authors proposed a method to combine multilevel features from CNN for the purpose of multimodal biometric detection. They used multimodal input streams with multiple CNN. The CNN features extracted at various levels were compressed and combined together to improve classification accuracy.

Multilayer CNN features fusion was also utilized for remote sensing hyperspectral image classification [19]. The authors tried to achieve more efficient feature discrimination. Pretrained CNN model was used to extract multilayer features. They also used the Fisher kernel coding scheme to build a mid-level feature representation. These features were then combined using PCA.

All of the feature fusion methods discussed above improved performance and accuracy using multilevel features from different CNN layers. In this study, we propose a feature fusion approach for MI classification with multiple CNN's. EEG MI signals are non-stationary with have low SNR. Since the MI signals have subject dependent properties, they differ from person to person. Therefore to address the issues with feature extraction of MI signals, we propose multi-layer CNNs feature extraction and fusion model, which has been applied before for EEG classification.

## 3. Proposed multi-layer CNNs feature fusion methods

Our proposed methods for multi-layer CNNs feature fusion is composed of multilevel CNN methods, a multilayer perceptron (MLP) and autoencoders. In the first phase, the CNNs are pretrained individually on an MI dataset. Then in the second phase, these pretrained method are trained on the target MI dataset. After the training phases are complete, the CNN features are concatenated and passed as input to MLP or autoencoder for fusion. The MLP and autoencoder fusion methods are then trained separately using the CNN features. Then a softmax activation function layer is added to both MLP and autoencoder methods and they are fine-tuned to get output probabilities for the MI classes.

The CNN models are based on AlexNet [9] architecture, a well-known CNN model which has achieved good results for image classification problems. CNN models are composed of convolution and pooling layer blocks. CNN has achieved good results on signals which have a natural hierarchical structure, like images. Initial convolution layers learn edges and boundaries for objects and later convolution layers learn more complex object shapes. Hence, the later layers use convolution operation together with nonlinearities to produce high-level complex features which are a combination of low-level simple features. Pooling layers reduce the dimensions of the convolution features and thereby induce translational invariance in the CNN. In this progressive manner, CNN can learn hierarchical features automatically layer by layer. We have used this generic CNN architecture to study their effect on EEG classification. Such CNN architectures have been able to extract good features for image and speech processing tasks. The CNN model with four convolution layers (CNN-4) is shown in Fig. 1. The architectural details are given in Table 1.

As we needed to test different CNN design strategies, we created and tested multiple CNN models with a different number of layers and filters. We started with CNN models with a single convolution, pooling block and a dense classification layer at the end. We progressively increased the number of convolution and pooling blocks, until the performance of the models degraded. We found from the literature that most of the successful studies using CNN or other deep learning models for EEG classification have shallow architectures [32–36]. Some of them just have only one or two layers [24,25]. We recreated the code for deep CNN [24] to act as a baseline and to compare our results with this model since this study is the best deep learning technique available for EEG MI classification.

The CNN model with one convolution-pooling block consists of one logical convolution, which is split into two separate convolutions. As the MI recordings have multiple channels ranging from 3 to 128 channels, this strategy helps to manage multi-source inputs. In this strategy proposed in [25], the first convolution operation is performed on each channel or across some time-samples, and the second convolution is done for all the channels simultaneously, one sample at a time. The resulting effect is convolution

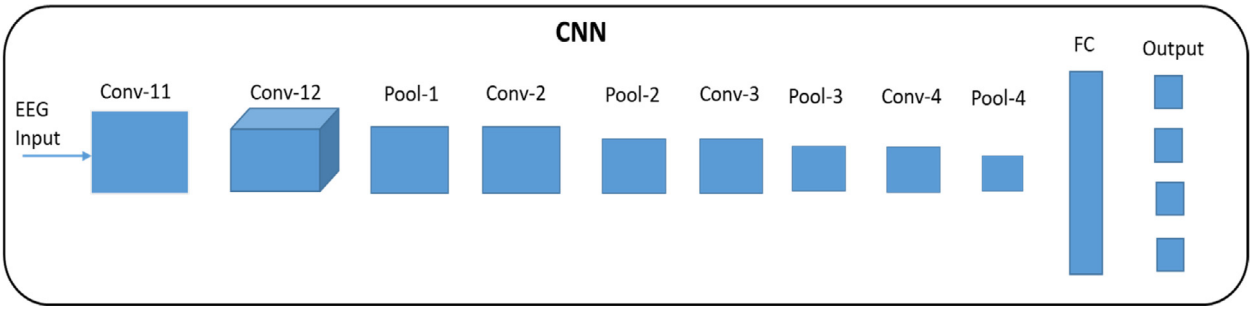



**Fig. 1.** CNN model with four convolution layers (CNN-4).

across all input channels for a number of samples. The MI data is fed to the CNN as a 2-dimensional array, having channels as rows and time-samples as columns. The split convolution favors this representation. The first convolution across time-samples can learn temporal features and the second convolution across channels is more adept to learn spatial features.

To make the CNN learn generic features, we tested the models with the different training techniques, nonlinearities, and activation functions. We also used recent normalization techniques like dropout and batch normalization.

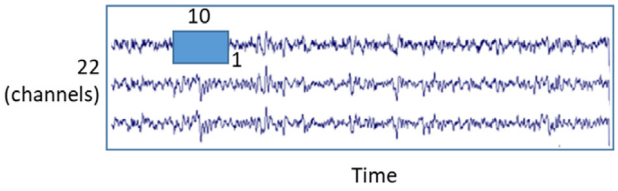
The CNN models are trained and evaluated on popular EEG MI dataset, BCI Competition IV dataset 2a (BCID) [40], recorded by Graz University researchers. Numerous machine learning and deep learning techniques have used this dataset, this would make the evaluation of our proposed model more accurate.

In some studies, the EEG signal was converted into topo-maps and images [32] to help CNN models which favor 2D inputs. However, this conversion risks losing important information and features. The research also shows that EEG data is correlated over time, therefore we use raw EEG data, and aim to learn generic features. Some researchers have used electrode voltage over the flattened scalp surface and converted the EEG signal to topographical time series images [32]. However, there is evidence that EEG signals are correlated over time scales which involves modulation in time [41]. Recently CNN has achieved good accuracies for EEG data represented as 2D input, with time-samples across channels [24,25]. For these reasons, we also store MI data in the form of a 2D array. The signals are minimally preprocessed, just to remove noise.

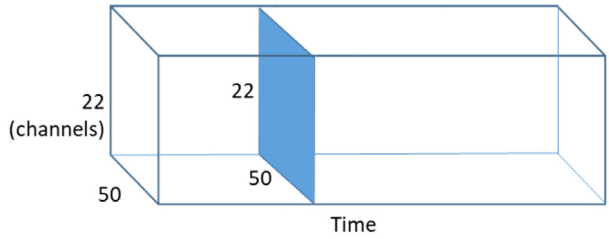
As the public BCID dataset used is small in size, we pretrain all the CNN models on the High Gamma dataset (HGD) [24]. Using pretraining also helps to avoid overfitting on training data having a small size. HGD is an MI dataset created under controlled recording conditions and therefore contains minimum noise. It is recorded using 128 electrodes from 20 healthy subjects. It consists of 880 trials in training set and 160 trials in the test set. As more training data is available as compared to the BCID dataset, it is an excellent resource for pretraining deep learning models.

### 3.1. Training the multi-layer CNNs method

There are two techniques used by researchers for training systems on EEG datasets. 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 acts as a testing set and




a. In the first part, convolution is performed across time steps




b. In the second part, convolution is performed across all channels

**Fig. 2.** (a & b) The first convolution operation split into two parts.

all the others are part of 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 techniques called cross-subject training is more challenging, and the evaluation is more robust and generalized. We used both of these techniques for training and testing our proposed deep learning method.

The BCI Competition IV dataset 2a (BCID dataset), is the MI dataset which is used to train and evaluate our proposed method. This MI dataset is a challenging dataset which has been used by numerous researchers. It has been recorded using 22 scalp electrodes from nine healthy subjects over two sessions. Each session consists of a total of 288 fours second trials. In each trial, subjects were asked to imagine moving of left hand, right hand, feet, and tongue [40].

The BCID dataset is bandpass filtered into four different frequency bands, 0–38 Hz, 4–38 Hz, 8–31 Hz, and 8–38 Hz. Exponentially moving standardization was applied for each electrode with a decay factor of 0.999 for computing mean and variances to standardize the continuous EEG signal.

Research has shown that different frequency bands show different response for each subject for EEG motor imagery classification hence the accuracy for each subject is band specific [42]. Since alpha, beta and lower gamma frequency bands show best response for motor imagery classification [5], the input signal has been divided into alpha, beta bands. Additionally many researchers have used combined frequency bands also [5], obtaining improvement over separate frequency bands. Hence this paper

**Table 1**
Structure of CNN models used for feature fusion.

| CNN-1 | CNN-2 | CNN-3 | CNN-4 |
|---|---|---|---|
| Conv (30 × 1, 50 filters) | Conv (25 × 1, 50 filters) | Conv (20 × 1, 50 filters) | Conv (10 × 1, 50 filters) |
| Conv (1 × 22, 50 filters) | Conv (1 × 22, 50 filters) | Conv (1 × 22, 50 filters) | Conv (1 × 22, 50 filters) |
| Max Pool (3 × 1, stride 3) | Max Pool (3 × 1, stride 3) | Max Pool (3 × 1, stride 3) | Max Pool (3 × 1, stride 3) |
| Dense (1024) | Conv (10 × 1, 100 filters) | Conv (10 × 1, 100 filters) | Conv (10 × 1, 100 filters) |
| Softmax (4 classes) | Max Pool (3 × 1, stride 3) | Max Pool (3 × 1, stride 3) | Max Pool (3 × 1, stride 3) |
| | Dense (1024) | Conv (10 × 1, 100 filters) | Conv (10 × 1, 100 filters) |
| | Softmax (4 classes) | Max Pool (3 × 1, stride 3) | Max Pool (3 × 1, stride 3) |
| | | Dense (1024) | Conv (10 × 1, 200 filters) |
| | | Softmax (4 classes) | Max Pool (3 × 1, stride 3) |
| | | | Dense (1024) |
| | | | Softmax (4 classes) |

investigates the effect of supplying separate as well as combined frequency bands, for each subjects and for CNN models.

The EEG data is cropped using 2 s sliding window, before being fed to the CNN. The window is slid for each time sample and as a result, a large number of crops are created. These crops increase the size of the training data [24] and also increases the decoding performance. It also saves the CNN models from overfitting on the limited training samples. Cropped training forces CNN to learn generic features using 2 s crops, which otherwise would learn features specific to the trial, or subject. The sampling frequency used for the BCID dataset is 256 Hz, which means that each of the 4 s trials consists of around 1000 samples. Cropping into the 2 s window makes it about 500 samples in each input.

We got good performance with multiple CNN models having a varying number of convolution-pooling blocks and different size of filters. The first logical convolution operation is split into two parts, the first convolution is done across time-samples and the second convolution performed across all channels like a spatial filter as shown in Fig. 2. Convolutions are followed by nonlinearity, max-pooling and dense layer with softmax. The performance increased when we used batch normalization and dropout. Exponential linear units (ELU) is used as the activation function. We also tried ReLU but ELU were faster and gave better accuracy. The number and size of filters and stride for each CNN model is given in Table 1. The softmax function produces probability scores for each class. Batch normalization technique also helped to improve the performance.

The network parameters are optimized using Adam algorithm which is a mini-batch stochastic gradient optimization method. Adam is a suitable optimizer for high-dimensional parameters such as in CNN. The CNN models are jointly optimized after training. The fusion methods, MLP and autoencoders are trained and optimized separately using Adam while the CNN models are frozen.

We increased the convolution-pooling blocks gradually to see if there is an increase in performance. With two and three-layered architectures the performance degraded a bit, however, the CNN model with four convolution-pooling blocks (CNN-4) showed better learning capabilities. The filter size and numbers were varied across models until we had the best results for each.

Using more than four convolution-pooling blocks, the performance degraded and we could not find any other CNN architecture which gave us improved performance. This observation is in line with other studies that have used CNN architectures with few layers for EEG decoding. Deeper CNN models might not be suitable for EEG decoding as no research achieved good EEG decoding accuracy with them.

CNN models with one convolution-pooling block (CNN-1), two convolution-pooling blocks (CNN-2), three convolution-pooling blocks (CNN-3) and four convolution-pooling blocks (CNN-4) showed reasonable performance, with particular filter sizes. CNN-1 achieved the best result with filter size 30, CNN-2 and CNN-3 with filter size 25 and 20 respectively, while CNN-4 learned best features for a filter size of 10. Shallow CNN using larger filter size as compared to deeper CNN shows that the shallow CNN might be good at learning specific temporal and spatial features like FBCSP, but deeper models can be good in extracting generic EEG features. In this study, we tried to investigate whether fusing the features from these different CNN models can help us to improve the MI decoding performance.

The CNN parameters are trained using a supervised learning algorithm which produces a real number as output for each class. CNN processing can be represented as a function: $\boldsymbol{f}(\boldsymbol{X^j};\boldsymbol{\theta})$: $\mathbb{R}^{\boldsymbol{E.T}} \longrightarrow \mathbb{R}^{\boldsymbol{K}}$ where $\theta$ denotes function parameters (weights and bias), $E$ denotes electrodes, $T$ are the time steps and $E$ are output labels for the trial $j$. Training is carried out per subject so the output is converted by softmax function into subject-specific conditional probability for each class label for a given output:

$$\boldsymbol{p}\left(l_k\middle|\left(\boldsymbol{f}\left(\boldsymbol{X^j};\boldsymbol{\theta}\right)\right)\right)=\frac{e^{\boldsymbol{f_k}(\boldsymbol{X^j};\boldsymbol{\theta})}}{\sum_{k=1}^{K} e^{\boldsymbol{f_k}(\boldsymbol{X^j};\boldsymbol{\theta})}} \tag{1}$$

The network is trained by minimizing the sum of per example loss and by assigning a high probability to the correct output labels. This can be represented as:

$$\boldsymbol{\theta^*}=\boldsymbol{\arg\min_{\theta}}\sum_{\boldsymbol{j=1}}^{\boldsymbol{N}}\boldsymbol{loss}\left(\boldsymbol{y^j},\boldsymbol{p}\left(l_k\middle|\boldsymbol{f_k}\left(\boldsymbol{X^j};\boldsymbol{\theta}\right)\right)\right) \tag{2}$$

where the loss is the negative log-likelihood:

$$\boldsymbol{loss}\left(\boldsymbol{y^j},\boldsymbol{p}\left(l_k\middle|\boldsymbol{f_k}\left(\boldsymbol{X^j};\boldsymbol{\theta}\right)\right)\right)$$
$$=\sum_{\boldsymbol{k=1}}^{\boldsymbol{K}}-\boldsymbol{\log}\left(\boldsymbol{p}\left(l_k\middle|\boldsymbol{f_k}\left(\boldsymbol{X^j};\boldsymbol{\theta}\right)\right)\right)\cdot\delta(\boldsymbol{y^j}=l_k) \tag{3}$$

The early stopping technique is used which divides the training set into a fold of training and validation set. When there is no improvement in the validation accuracy for a number of epochs, the first training phase is stopped. It is continued using the parameters which gave the best validation accuracy, on the same training and validation fold. When the loss on validation fold decreases to the loss of training fold the training ends. In early stopping strategy, the number of training epochs is not predefined which helps to automatically choose an optimum value.

The features that are extracted by CNN layers act as input to the classification layer. Machine learning techniques such as FBCSP has separate feature extraction and classification stages, but in CNN both these stages are jointly optimized during training and in this manner, the CNN is able to learn such robust features which can provide a discriminative classification for EEG data. As the EEG signals are complex and noisy hence CNN features can carry more information than handcrafted features like FBCSP.

EEG data is recorded in many sessions as it is both time taking and tiring activity for the subjects. As the training is conducted per subject, the sessions for each subject are divided such that

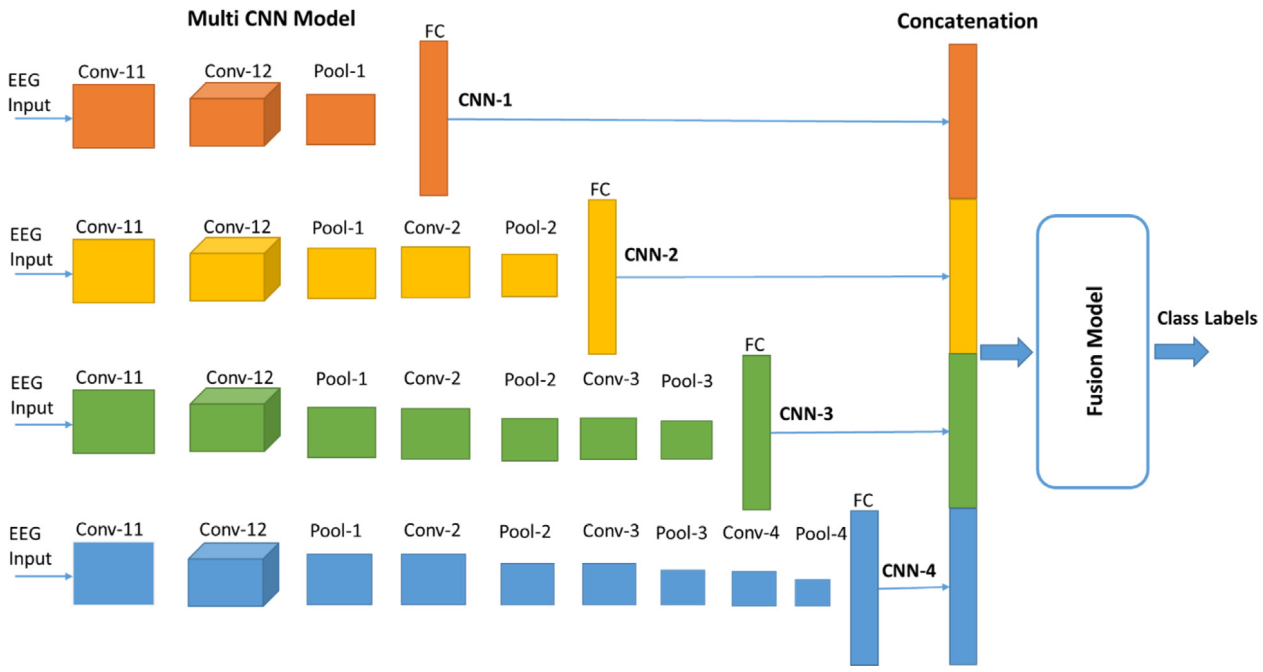



**Fig. 3.** Multi-layer CNNs.

each of the two sessions acts as a training set and the other session as a testing set. Finally, the accuracy is calculated by averaging the values obtained for all the sessions for each subject.

### 3.2. Multi-layer CNNs feature fusion

In this study, we propose a fusion method to combine CNN models for EEG classification. Researchers have used one layer CNN models [24] and have achieved similar state-of-the-art results as reported by FBCSP. Some other researchers have proposed CNN models with more number of layers [24,25] to achieve good decoding accuracy. In this study, we build a fusion method using MLP and autoencoders to fuse CNN models. Different CNN architectures might be good in extracting different types of EEG features, hence their fusion can help us to build generic features for EEG decoding. Feature fusion for deep learning models has not been developed and evaluated for EEG data.

As EEG data is time-series recording which has multiple channel sources, has low SNR, and is a non-stationary nature, it is a challenging task to extract relevant features. Conventional machine learning has achieved good results and deep learning methods have tried to improve those, but have failed to report a substantial increase in performance. Thus, by applying the feature fusion method we can utilize domain-specific knowledge extracted by various CNN models to build class-discriminative generic features set that improves EEG decoding accuracy.

We fuse CNN-1, CNN-2, CNN-3, and CNN-4 models, as these models are able to extract good features and achieve better accuracy than the other CNN architectures. Feature fusion is done using two different architectures, MPL, and an autoencoder. Both of these networks have been utilized by researchers for fusion and features extraction [25,26,43,44]. After the pretrained CNN models on the HGD, they are fused together by removing the final softmax classification layer from each of them and concatenating the features using a linear layer. We call this architecture as multi-layer CNNs method as shown in Fig. 3. The multi-layer CNNs method is now trained on the BCID dataset using both the within-subject and the cross-subject training approach. The resulting multi-layer CNNs features from the concatenation layer are fed to the MLP. The MLP consists of two hidden layers, each having 50 nodes. The complete fusion method composed of CNNs and MLP is named as MCNN. We used a 50% dropout rate, to achieve good generalization. The MLP method is then trained on the combined feature vector, and the output is sent to the softmax layer to get the probability score for the MI classes. The overall method architecture is provided in Fig. 3.

### 3.3. Cross encoding with autoencoders

An autoencoder is usually a three-layer neural network with an input layer, one hidden layer and an output layer [20], with the output layer having the same number of neurons as the input layer in order to reconstruct its own inputs. Therefore, autoencoders are unsupervised learning methods. An autoencoder is trained so that the input $x$ is mapped to the hidden layer, this stage is called the encoding stage, then the output of hidden layer $z$ is mapped to the output layer, to reconstruct the input, this stage is called the decoding stage. These steps are shown in the following equations.

$$z = \sigma(Wx + b) \tag{4}$$

$$x' = \sigma'\left(W'x + b'\right) \tag{5}$$

where $W$ and $W'$ are weight matrix, $b$ and $b'$ are bias vectors. The functions $\sigma$ and $\sigma'$ are element wise activation functions. The weights are said to be tied if $W = \left[W'\right]^T$. Then the autoencoders is trained to minimize reconstruction error $E\left(x, x'\right)$.

$$\arg\min_{W,b}\left[E\left(x, x'\right)\right] \tag{6}$$

Some researchers have trained autoencoders for learning CNN features [26,43], and have been able to extract improved features and provide better performance than CNN only networks. This is because the autoencoders are trained in a way to automatically learn hidden and robust features by reconstructing the input. In this study we have used autoencoders for fusing the CNN features and extracting relevant information from the combined feature vector.

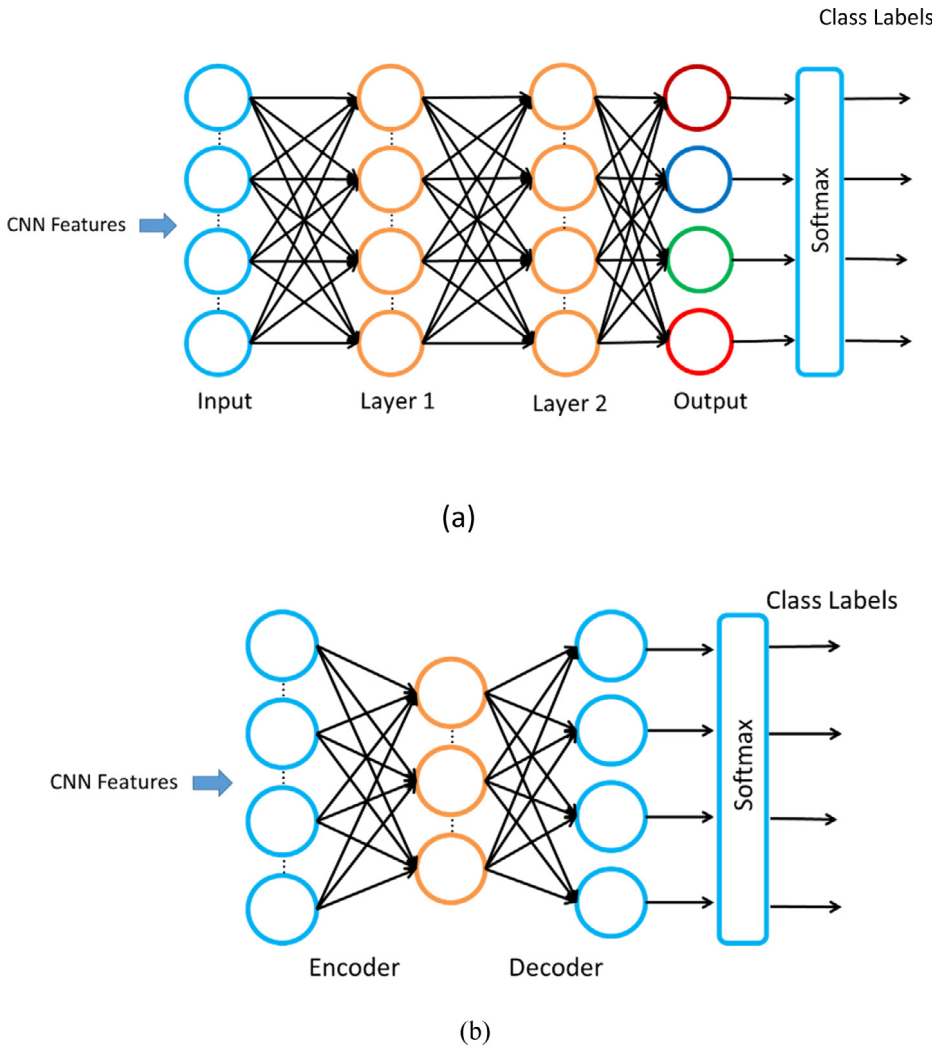



**Fig. 4.** Feature fusion models. (a) MLP. (b) Autoencoder. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

As the EEG signal is complex, and it differs from subject to subject as well as for different trials for the same subject. Therefore discriminative set of features are required to classify the data accurately. Even in controlled recording environment there are differences between trials of the same subject. Hence extracting generic features for all subjects is a tough task. In this study the autoencoder is trained in a novel manner to obtain robust and generic feature representation across subjects. The proposed technique shows a major improvement in cross subject accuracy. The technique is able to extract EEG patterns that are stable across subjects and trials.

The CNN features are concatenated and supplied as input to the autoencoders. The autoencoder used in this study has 100 nodes in the hidden layer. Cross-encoding method is used motivated from autoencoder pretraining methods used in [45]. The combined feature set given as input is not reconstructed by the autoencoder, but it is forced to reconstruct a feature set belonging to the same class but different subject. The autoencoder acts as a bottleneck and the relevant and common features belonging to the class are learned and the noise is also removed automatically.

The trials for the same class for different subjects show a lot of differences so the autoencoders is given the combined CNN feature vector for a particular trial for a class C and is forced to reconstruct a different trial belonging to the same classC. If a class has $n$ trials then $n^2$ input and target trial pairs can be constructed for each class. The autoencoder reconstructs different trial for given input and the best model is saved. When the autoencoder is trained for the same subject, it reconstructs a trial from another session from the same subject and same class. For cross subject autoencoder training, it is provided a trial from a particular class and it reconstructs a trial belonging to the same class but another subject.

This cross-encoding scheme also helps to increases the number training samples manifolds. The softmax layer acts as a classifier for the reconstructed features. Cross encoding helps to extract common adaptations reflecting individual differences between subjects. This method composed of fused CNNs and cross-encoding autoencoders is named as CCNN. The model for the best accuracy for input and output pair for each class is stored and used for testing. Fig. 5 shows the cross encoding method.

## 4. Experiments and results

The experiments were performed on a machine having 17 cores Intel Xeon processors and 64 GB RAM. GTX 1080 GPU having 8 GB memory was utilized for training and testing deep learning models. PyTorch framework was used for developing CNN and feature fusion models. Preprocessing of EEG MI data was done using MNE-Python.

**Table 2**
Subject-specific classification results obtained for the BCID and HGD datasets.

| Methods | Description | Accuracy (BCID) | Accuracy (HGD) |
|---|---|---|---|
| Ang et al. [5] | Filter bank CSP | 68.0% | 91.2% |
| Tabar et al. [26] | 1D CNN with SAE | 70.0% | – |
| Lawhern et al. [36] | CNN with depth and separable convolutions | 69.0% | – |
| Schirrmeister et al. [24] | CNN with cropped training | 72.0% | 92.5% |
| Sakhavi et al. [25] | Temporal features with FBCSP and CNN | 74.4% | – |
| Multi-level CNN [44] | CNN layers fusion | 74.5% | – |
| Proposed method | **MCNN** | **75.7%** | **95.4%** |
| | CCNN | 73.8% | 93.2% |

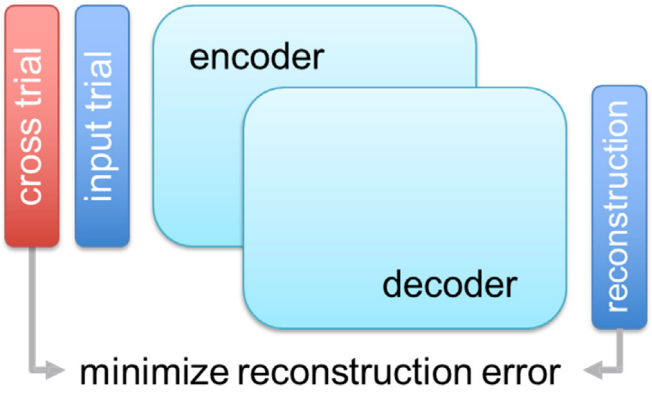



**Fig. 5.** Cross-encoding with autoencoders.

Deep CNN model [24] was chosen as a baseline to evaluate the proposed multi-layer CNNs fusion methods as this study had recently achieved the best results for EEG classification [46]. This model [24] is implemented in PyTorch and tested on the BCID dataset. The comparison of the overall accuracy of the proposed methods and other methods is shown in Table 2. The proposed MCNN method gave good accuracy improvement over the baseline [24] which also used CNN.

The comparison of the overall accuracy of our proposed models and other models is shown in Table 2. One-sided Wilcoxon signed-rank test is used to evaluate the significance of the accuracy. The Wilcoxon signed-rank test has also been used by other researchers for showing statistical significance for their EEG classification approaches as p-value is not an accurate measure of significance where the number of paired statistical samples are relatively small in number and are non-Gaussian [24,25]. The proposed MCNN method outperforms other methods on both the BCID as well as the HGD. While CCNN method registered 73.8% accuracy on the BCID dataset which is similar to the results reported by previous studies, the MCNN method achieved even better accuracy of 75.7% for subject-specific training and testing on the BCID dataset. (Using Wilcoxon signed-rank test, there is significant increase with $p < 0.05$ in the average accuracy for MCNN method but there is no significant difference in accuracy for CCNN method with respect to the baseline method [24] accuracy).

The shallow and deep CNN model [24], was developed for MI classification using cropped training and got 73.0% and 71% decoding accuracy respectively on the BCID dataset. Another CNN based model [26] also utilized SAE over 1D CNN features and achieved an accuracy of 70% for the BCID dataset. A CNN based compact architecture called EEGNet [36] was developed and tested for various EEG applications, and they achieved an accuracy of 69% for BCID dataset. Filter bank common spatial patterns (FBCSP) [5] was the most successful conventional machine learning technique which produced the best results for EEG decoding before the deep learning techniques were applied to EEG data. This technique used handcrafted spatial features to give 68% accuracy on the BCID dataset. FBCSP and CNN features were combined in [25] in a novel way to produce good results for MI decoding. The technique used FBCSP and CNN to create static and dynamic energy envelopes respectively for EEG data, and extract temporal and spatial features from the BCID dataset. This technique was able to achieve 74.4% test accuracy on the BCID dataset. In [44] researchers proposed fusion of CNN layers and obtained good accuracy of 74.5%. The multi-layer CNNs feature fusion models proposed in this study were able to get better overall accuracy on the BCID and HGD datasets. While CCNN method registered 73.8% accuracy on the BCID dataset which is similar to the results reported by previous studies, the MCNN method achieved even better accuracy of 75.7% for subject-specific training and testing on the BCID dataset. (Using Wilcoxon signed-rank test, there is significant increase with $p < 0.05$ in the average accuracy for MCNN method but there is no significant difference in accuracy for CCNN method with respect to the baseline model [24] accuracy). We also tested individual pretrained CNN models, which showed that the fusion methods offered improved performance as shown in Table 3.

Individual pretrained CNNs are tested, which showed that the fusion methods offered improved performance as shown in Table 3. Each CNN gave a good result for a particular frequency band, so a different band-pass filtered signal is used for each CNN. For CNN-1, we used alpha band (7–13 Hz) as input, CNN-2 was fed with the beta band (13–31 Hz), CNN-3 with combined alpha and beta band (7–31 Hz) and CNN-4 with (0–40 Hz). Five blocks CNN-5 model was also tested but the accuracy dropped and it did not help in accuracy improvement in the fusion network as well. These results show that CNN models with different depths can obtain improvement in accuracy based on the frequency bands used, hence frequency specific input is important for CNN models. It also shows that the deeper models gave better results with combined frequency bands. CNN-4 and CNN-5 give best results with 0–40 Hz frequency band. It can also be noticed that the shallow models give the best result with larger filter size as compared to the deeper models. Hence filter size has an important role in CNN models.

In Table 4, some of the results achieved with the fusion of different CNNs are presented. The fusion of CNN-1 with CNN-4, and CNN-1 with CNN-2 and CNN-4 gave an improvement in accuracy but the accuracy obtained by fusing all CNNs was the best one overall. Table 5 shows the comparison of the subject-specific accuracy obtained by different models. Different models fared better for a different subject.

Tables 6–8 give the confusion matrix for different output classes for the BCID dataset, for MCNN, CCNN fusion methods and the baseline method [24] respectively. Both proposed methods

**Table 3**
Classification results for CNNs with different number of Convolution-Pooling Blocks layers with a corresponding convolution filter size.

| CNN with different no. of Conv-Pooling blocks | Frequency bands | Conv-Filter size | Accuracy (BCID) | Accuracy (HGD) |
|---|---|---|---|---|
| CNN-1 | 7–13 Hz | 30 × 1 | 73.7% | 89.1% |
| CNN-2 | 13–31 Hz | 22 × 1 | 71.3% | 88.6% |
| CNN-3 | 7–31 Hz | 15 × 1 | 70.4% | 90.2% |
| CNN-4 | 0–40 Hz | 10 × 1 | 72.8% | 92.8% |
| CNN-5 | 0–40 Hz | 10 × 1 | 67.0% | 86.4% |

**Table 4**
Classification results for combinations of CNN fusion method.

| Fusion stages | Accuracy |
|---|---|
| CNN-1 + CNN-2 | 71.9% |
| CNN-1 + CNN-3 | 70.3% |
| CNN-1 + CNN-4 | 74.5% |
| CNN-2 + CNN-4 | 69.7% |
| CNN-1 + CNN-2 + CNN-4 | 73.9% |
| CNN-1 + CNN-3 + CNN-4 | 72.2% |
| MCNN | 75.7% |
| CCNN | 73.8% |

**Table 5**
Subject-specific classification results for each subject for the BCID dataset.

| Subjects | Schirrmeister et al. [24] | Sakhavi et al. [25] | (Proposed method) MCNN | (Proposed method) CCNN |
|---|---|---|---|---|
| Subject-1 | 86.56 | 87.5 | 90.21 | 87.14 |
| Subject-2 | 62.29 | 65.28 | 63.4 | 63.1 |
| Subject-3 | 89.86 | 90.28 | 89.35 | 86.76 |
| Subject-4 | 65.61 | 66.67 | 71.16 | 68.29 |
| Subject-5 | 55.19 | 62.5 | 62.82 | 63.61 |
| Subject-6 | 48.47 | 45.49 | 47.66 | 48.32 |
| Subject-7 | 86.07 | 89.58 | 90.86 | 87.73 |
| Subject-8 | 78.41 | 83.33 | 83.72 | 80.17 |
| Subject-9 | 76.05 | 79.51 | 82.32 | 78.83 |
| Average | 72.05 | 74.46 | 75.72 | 73.77 |

**Table 6**
Confusion matrix for the proposed MCNN fusion method.

| Predictions | | | | | |
|---|---|---|---|---|---|
| | | Left hand | Right hand | Feet | Tongue |
| Targets | Left hand | **81.71** | 11.07 | 4.25 | 3.03 |
| | Right hand | 10.33 | **82.07** | 4.02 | 3.58 |
| | Feet | 9.39 | 11.25 | 67.23 | 12.13 |
| | Tongue | 8.56 | 12.24 | 7.13 | **72.09** |

**Table 7**
Confusion matrix for the proposed CCNN fusion method.

| Predictions | | | | | |
|---|---|---|---|---|---|
| | | Left hand | Right hand | Feet | Tongue |
| Targets | Left hand | 77.52 | 14.07 | 4.87 | 3.54 |
| | Right hand | 11.05 | 78.75 | 6.17 | 4.03 |
| | Feet | 11.05 | 9.18 | 66.69 | 13.11 |
| | Tongue | 9.13 | 11.22 | 8.01 | 71.81 |

show better decoding accuracy for left and right-hand movement imagination tasks. It can be also noticed that the proposed MCNN method provides accuracy improvement for the "left", "right" hand classes as well as the "feet" class.

One of the major contributions of this work is cross-subject classification improvement. CCNN method with cross encoding technique better results for cross-subject EEG classification than any other result reported in the literature so far. This study is the first one to investigate the effects of cross-trial autoencoder training and this method provided us with state-of-the-art performance as shown in Table 9. Cross-trial autoencoding not

**Table 8**
Confusion matrix for the baseline method [24].

| Predictions | | | | | |
|---|---|---|---|---|---|
| | | Left hand | Right hand | Feet | Tongue |
| Targets | Left hand | 78.01 | 10.81 | 5.33 | 4.03 |
| | Right hand | 9.27 | 79.23 | 6.45 | 4.91 |
| | Feet | 12.06 | 8.31 | 68.52 | 11.02 |
| | Tongue | 9.16 | 11.48 | 13.92 | 65.12 |

**Table 9**
Cross-subject classification results for the BCID and HGD datasets.

| Methods | Description | Accuracy (BCID) | Accuracy (HGD) |
|---|---|---|---|
| Ang et al. [5] | Filter bank CSP | 38.0% | 65.2% |
| Lawhern et al. [36] | CNN with depth and separable convolutions | 40.0% | – |
| Schirrmeister et al. [24] | CNN with cropped training | 41.0% | 69.5% |
| Sakhavi et al. [25] | Temporal features with FBCSP and CNN | 44.4% | – |
| Proposed method | MCNN | 42.1% | 71.4% |
| | **CCNN** | **55.3%** | **79.2%** |

**Table 10**
Cross-subject classification results for each subject for the BCID dataset.

| Subjects | Schirrmeister et al. [24] | (Proposed method) MCNN | (Proposed method) CCNN |
|---|---|---|---|
| Subject-1 | 47.06 | 51.91 | **62.07** |
| Subject-2 | 31.22 | 38.06 | **42.44** |
| Subject-3 | 41.02 | 43.34 | **63.12** |
| Subject-4 | 33.19 | 35.81 | **52.09** |
| Subject-5 | 41.57 | 41.50 | **49.96** |
| Subject-6 | 34.71 | 31.11 | **37.16** |
| Subject-7 | 43.09 | 48.09 | **62.54** |
| Subject-8 | 46.01 | 45.01 | **59.32** |
| Subject-9 | 51.78 | 51.29 | **69.43** |
| Average | 41.07 | 42.09 | **55.34** |

only helped us to increase the training set manifolds but it also helped autoencoders learn generic EEG features that are not subject-specific. Subject-wise results for cross-subject classification shown in Table 10. As it can be noticed, the proposed method got more than 10 percent accuracy improvement for cross-subject classification when compared with state-of-the-art deep learning models.

Fig. 6 shows the average training time per subject. Both the fusion methods took extra training time which accounts for the increase in the number of parameters in the fusion methods.

### 4.1. Visualizing the learned representations

The drawback with CNN is that it is difficult to interpret and understand what the network is learning, how it is achieving such outstanding results and what type of features these models use for classification. Therefore, this study also analyzes CNN features which the CNN used for class discrimination. We know that features that the CNN layers extract depend upon what they perceive through their receptive field. A feature map is calculated for receptive fields in intermediate layers, which can show whether this feature affects the output of those particular layers or not. Therefore, in order find which features CNN layers are using it can be investigated whether domain-specific knowledge and class-discriminative features are being used by receptive fields of the CNN. Combined feature value can be computed for all the receptive fields and then we can calculate how this feature value affects the CNN output. Hence by calculating the correlation between feature values and CNN outputs, features are responsible for the CNN output can be inferred.

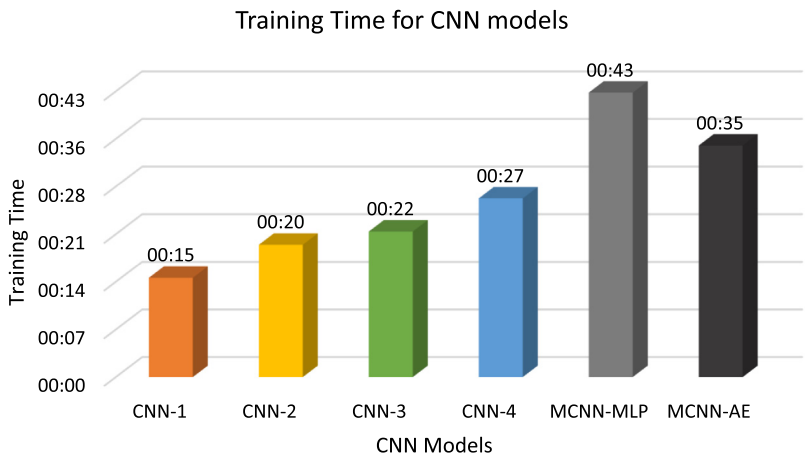



**Fig. 6.** Average training time per subject in (h: mm) for different CNN and fusion methods.

This approach for computing feature and output correlation is proposed by [24] to analyze and see how CNN learns from spectral amplitude features. As the amplitudes of the alpha, beta and gamma bands show discriminative information for classification of motor signals [5], therefore mean values for these frequency bands were calculated, to be used as feature values. Then these mean feature values are correlated inside a receptive field of the CNN as its total spectral amplitude with the CNN output. The correlation maps formed are a measure of the total spectral amplitude of a unit. These correlation maps are compared to the layer output, to see which of these amplitude features were used by the CNN. Positive or negative correlations showed that CNN can learn new information about these features. To crosscheck whether the correlation maps used the amplitude features, the amplitude is changed, to see whether the output of CNN changes. Hence, by varying the data artificially, the amplitude and features also alter, and it can be seen whether there is a change in the output of the CNN. In this manner, it can be confirmed whether the amplitude features are used by CNN. If the positive or negative correlations are different from those of untrained CNN, it suggests that the trained CNN used these features more than the untrained one. By this correlation method, the feature amplitudes that cause CNN to change its outputs can be known.

To make changes in the amplitude all the input training trials are converted into the frequency domain by using Fourier transform. Then Gaussian noise is randomly added to make changes in the amplitudes. After the changes, the frequency domain signal is converted back into the time domain. Then the output of CNN is computed for input trials before and after making changes to the amplitude. The output was computed just before the softmax layer. These values are then correlated with the change in the CNN outputs.

An example of the correlation scalp map for the alpha, beta and gamma band is shown in Figs. 7 & 8, where the color encodes the correlation of amplitude changes at the position of the corresponding electrodes and the corresponding prediction changes of the MCNN method.

Negative correlations on the scalp areas related to left hand motor imagery shows that a decrease in amplitude in this areas leads to an increase in prediction for the right-hand class and the positive correlations on the scalp area corresponding to right-hand motor imagery shows that a decrease in amplitude leads to a decrease in prediction for the right-hand class. These findings complement the information that shows band power in these areas is strongly correlated with motor imagery tasks. The frequency bands the CNN is using to produce the output, and the brain regions involved can be obtained. Hence, the visualization method shows the spatial distribution of the features learned by CNN while learning the motor imagery classes, in different frequency bands. The method also shows that the MCNN learned to extract and use band power features with specific spatial distributions. As shown in Fig. 7 for alpha bands (7–13 Hz) is more responsive in CNN-1, the electrodes corresponding to the dark red and blue areas appear towards the center of the head on both left and right sides directly over the motor cortex areas. There is a slight difference of activations between CNN-1 and CNN-2, CNN-3, CNN-4. The CNN-2 network showed the highest correlation values for the beta band (13–31). CNN-3 showed good activation for a combined frequency band of (7–31), although the rest of the networks also showed the almost similar response for this band, as shown in Fig. 8. CNN-4 network gave the best result for a combined band of (0–40 Hz). These results and visualization indicates that CNN appear to be more responsive for the combined frequency band including alpha, beta and lower gamma frequency range.

## 5. Conclusion

Conventional machine learning had provided limited accuracy improvement for EEG MI classification, but deep learning techniques have been shown to achieve better results for the same task. We proposed novel multi-layer CNNs based fusion models for MI decoding. The results achieved by fusion of different CNN models proves that with different filter sizes and depths, CNN models can extract different types of features representing the EEG data at various abstract levels. The multi-layer CNNs approach is first of its kind that is used for EEG classification. Extracted CNN features were concatenated and fused using MPL and AE deep networks. The resulting fused features show improvement EEG decoding accuracy from the previous state of the art techniques and recent deep learning models. The study proves that CNN features are better than handcrafted features extracted by techniques such as FBCSP, which use spatial features. The study also showed that using pretrained CNN models can help improve accuracy, training time of deep learning models, and also prevent them from overfitting on the limited training data available in the public EEG datasets. The fusion architecture outperforms all the machine learning and deep learning based methods for classification of the EEG MI data. MCNN and CCNN show general improvement in decoding EEG for all output classes and all subjects on the BCID dataset which proves that fused CNN models can learn to find generic EEG representations that are applicable across subjects.

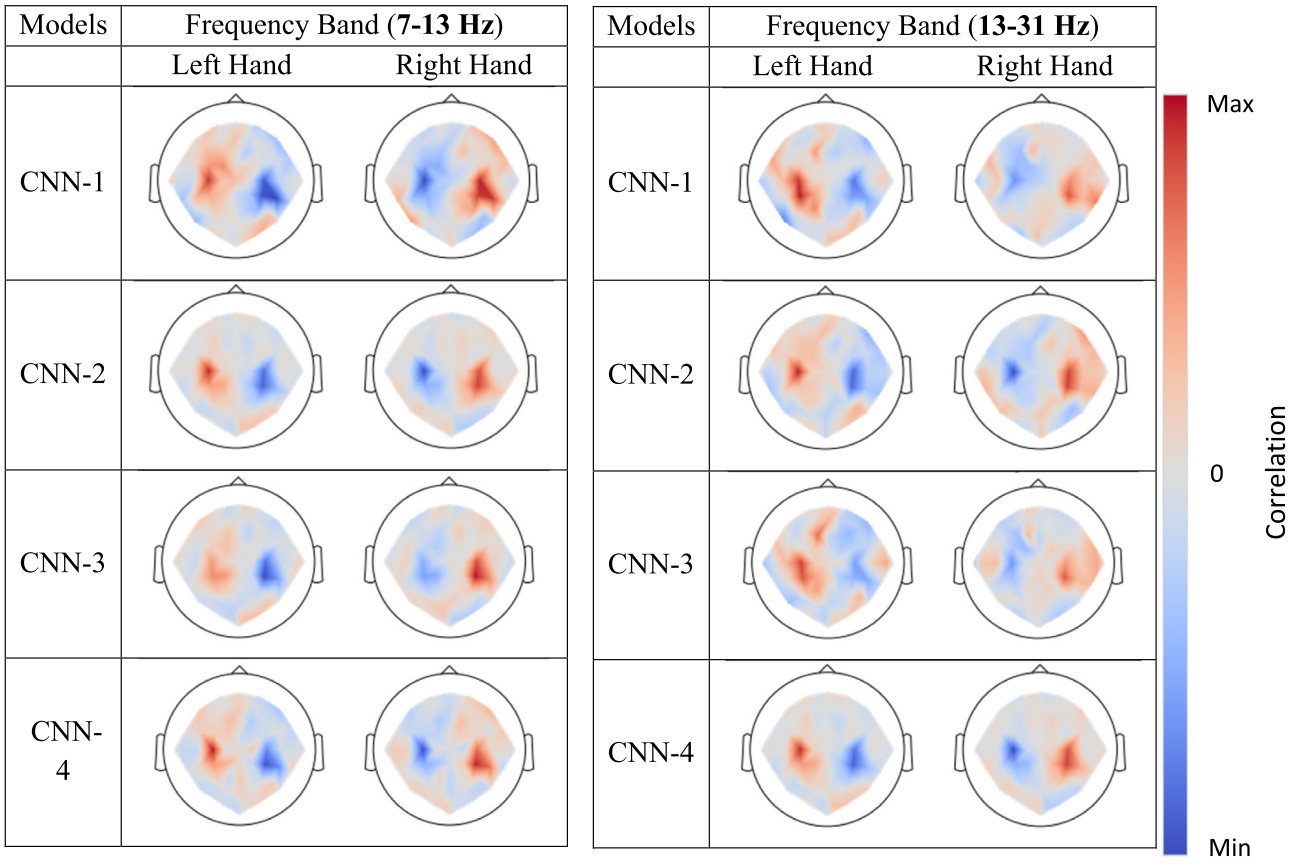



**Fig. 7.** Correlation maps for respective CNNs for different frequency bands (Alpha and Beta).

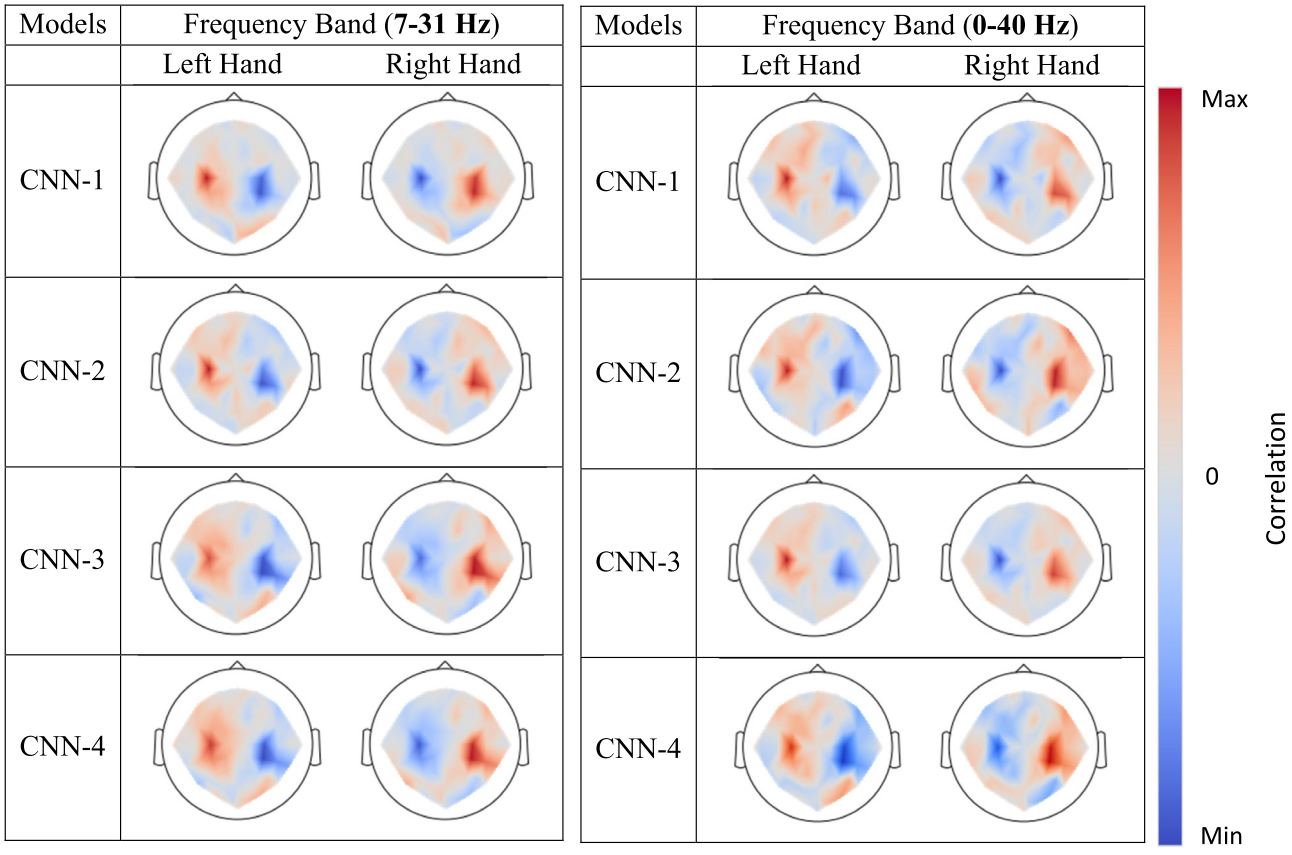



**Fig. 8.** Correlation maps for respective CNNs for combined frequency bands.

Cross-encoding technique was proposed to learn differences between individual subjects and trials, and to adapt the network to stable and common patterns across subjects. The results obtained by the proposed methods proves that CNNs with different architectures, depths, and filter sizes have an overwhelming effect on the accuracy and they can extract different feature representations, which can be fused to improve classification accuracy. Using pretrained CNNs can also help to improve feature learning and training on small-sized datasets. Cross-encoding approach used for autoencoders can help to improve the cross-subject classification. The proposed methods can learn a general representation of EEG signals which aids cross-subject classification.

Experimental results conducted on different challenging datasets confirm the superiority of the proposed fusion methods compared to state-of-the-art machine learning and deep learning methods for EEG classification. The proposed method has been evaluated for both subject-specific and cross-subject classification on challenging public dataset. Correlation maps have been used to analyze and visualize the features learned by different CNNs which help us to ascertain the type of information CNN is using.

The proposed fusion models show the ability to grasp spatially invariant characteristics of EEG MI signals; hence it would be quite interesting to apply the multi-layer CNNs fusion models on other EEG datasets. We also aim to study other methods for feature fusion to enhance the performance of the proposed method. We would like to make more variations in CNN architectures and use deep learning models such as LSTM which are preferred for extracting temporal features.

We also hope that the work proposed in this paper will encourage other researchers to improve the classification of brain signals using deep learning techniques. For future work, we want to further refine CNN models and the fusion methods to further improve both within subject and cross-subject classification accuracy. We would like to find out such robust features which would allow the proposed methods to be used as a part of advanced BCI systems by domain experts.

## Acknowledgment

This work was supported by the Deanship of Scientific Research at King Saud University, Riyadh, Saudi Arabia under project RG-1436-023.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## References

[1] L.J. Greenfield, J.D. Geyer, P.R. Carney, Reading EEGs: A Practical Approach, Lippincott Williams & Wilkins, 2012.
[2] G. Pfurtscheller, F.H. Lopes da Silva, Event-related EEG/ MEG synchronization and desynchronization: basic principles, Clin. Neurophysiol. 110 (1999) 1842–1857.
[3] J. Müller-Gerking, G. Pfurtscheller, H. H. Flyvbjerg, Designing optimal spatial filters for single-trial EEG classification in a movement task, Clin. Neurophysiol. 110 (1999) 787–798.
[4] M. Grosse-Wentrup, M. Buss, Multiclass common spatial patterns and information theoretic feature extraction, IEEE Trans. Biomed. Eng. 55 (1998) 1991–2000.
[5] K.K. Ang, Z.Y. Chin, C. Wang, C. Guan, H. Zhang, Filter bank common spatial pattern algorithm on BCI competition IV datasets 2a and 2b, Front. Neurosci. 6 (2012) 39.
[6] L. Tonin, T. Carlson, R. Leeb, J. Millán, Brain-controlled telepresence robot by motor-disabled people, in: 2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2011, pp. 4227–4230.
[7] M.S. Hossain, et al., Applying deep learning for epilepsy seizure detection and brain mapping visualization, ACM Trans. Multimedia Comput. Commun. Appl. (ACM TOMM) 14 (5) (2018) x, 16 pages.
[8] S.U. Amin, et al., Cognitive smart healthcare for pathology detection and monitoring, IEEE Access 7 (2019) 10745–10753, http://dx.doi.org/10.1109/ACCESS.2019.2891390.
[9] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, Vol. 25, Curran Associates, Inc., 2012, pp. 1097–1105, URL http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf.
[10] M.S. Hossain, G. Muhammad, Emotion recognition using deep learning approach from audio–visual emotional big data, Inf. Fusion 49 (2019) 69–78.
[11] A. Ghoneim, et al., Medical image forgery detection for smart healthcare, IEEE Commun. Mag. 56 (4) (2018) 33–37, http://dx.doi.org/10.1109/MCOM.2018.1700817.
[12] Y. Hao, J. Yang, M. Chen, M.S. Hossain, M.F. Alhamid, Emotion-aware video QoE assessment via transfer learning, IEEE MultiMedia 26 (1) (2019) 31–40, http://dx.doi.org/10.1109/MMUL.2018.2879590.
[13] M.S. Hossain, G. Muhammad, Environment classification for urban big data using deep learning, IEEE Commun. Mag. 56 (11) (2018) 44–50, http://dx.doi.org/10.1109/MCOM.2018.1700577.
[14] M. Alhussein, et. al, Cognitive IoT-cloud integration for smart healthcare: Case study for epileptic seizure detection and monitoring, Mob. Netw. Appl. (2018) 1–12.
[15] G. Muhammad, et al., Automatic Seizure detection in a mobile multimedia framework, IEEE Access 6 (2018) 45372–45383.
[16] M. Plis, Sergey, R. Hjelm, Salakhutdinov Devon, Bockholt Allen, Henry J. Johnson, J. Hans Paulsen, Deep learning for neuroimaging: a validation study, Front. Neurosci. (ISSN: 1662-453X) 8 (August) (2014) 1–11, http://dx.doi.org/10.3389/fnins.2014.00229.
[17] J. Lee, J. Nam, Multi-level and multi-scale feature aggregation using pretrained convolutional neural networks for music auto-tagging, IEEE Signal Process. Lett. 24 (8) (2017) 1208–1212, http://dx.doi.org/10.1109/LSP.2017.2713830.
[18] S. Soleymani, et al., Multi-level feature abstraction from convolutional neural networks for multimodal biometric identification. arXiv preprint arXiv:1807.01332 (2018).
[19] E. Li, J. Xia, P. Du, C. Lin, A. Samat, Integrating multilayer features of convolutional neural networks for remote sensing scene classification, IEEE Trans. Geosci. Remote Sens. 55 (10) (2017) 5653–5665.
[20] P. Zhang, et al., Amulet: Aggregating multi-level convolutional features for salient object detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2017.
[21] B. Hariharan, P. Arbelaez, R. Girshick, J. Malik, Hyper-columns for object segmentation and fine-grained localization, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 447–456.
[22] P. Bhattacharjee, S. Das, Two-Stream Convolutional Network with Multi-Level Feature Fusion for Categorization of Human Action from Videos Pattern Recognition and Machine Intelligence. PReMI 2017, in: Lecture Notes in Computer Science, Vol. 10597, Springer, Cham.
[23] K. Ueki, T. Kobayashi, Multi-layer feature extractions for image classification — Knowledge from deep CNNs, in: 2015 International Conference on Systems, Signals and Image Processing (IWSSIP), London, 2015, pp. 9–12.
[24] R.T. Schirrmeister, J.T. Springenberg, L.D.J. Fiederer, M. Glasstetter, K. Eggensperger, M. Tangermann, T. Ball, Deep learning with convolutional neural networks for EEG decoding and visualization, Hum. Brain Mapp. 38 (2017) 5391–5420, http://dx.doi.org/10.1002/hbm.23730.
[25] S. Sakhavi, C. Guan, Y. Shuicheng, Learning temporal information for brain-computer interface using convolutional neural networks, IEEE Trans. Neural Netw. Learn. Syst. 29 (11) (2018) 5619–5629.
[26] Y.R. Tabar, U. Halici, A novel deep learning approach for classification of EEG motor imagery signals, 14 (1) (2017) 016003.
[27] Rawashdeh Majdi, et al., Reliable service delivery in tele-health care systems, J. Netw. Comput. Appl. 115 (2018) 86–93.
[28] H. Cecotti, A. Graser, Convolutional neural networks for P300 detection with application to brain-computer interfaces, IEEE Trans. Pattern Anal. Mach. Intell. (ISSN: 01628828) 33 (3) (2011) 433–445, http://dx.doi.org/10.1109/TPAMI.2010.125.
[29] N. Guler, E. Ubeyli, I. Guler, Recurrent neural networks employing Lyapunov exponents for EEG signals classification, Expert Syst. Appl. (ISSN: 09574174) 29 (3) (2005) 506–514, http://dx.doi.org/10.1016/j.eswa.2005.04.011.
[30] M.T.F. Talukdar, S.K. Sakib, N.S. Pathan, S.A. Fattah, Motor imagery EEG signal classification scheme based on autoregressive reflection coefficients, in: 2014 International Conference on Informatics, Electronics & Vision (ICIEV), Dhaka, 2014, pp. 1–4.
[31] Hossain, et al., Improving consumer satisfaction in smart cities using edge computing and caching: A case study of date fruits classification, Future Gener. Comput. Syst. 88 (2018) 333–341.

[32] P. Thodoroff, J. Pineau, A. Lim, Learning robust features using deep learning for automatic seizure detection, in: Machine Learning for Healthcare Conference, 2016.
[33] H. Yang, S. Sakhavi, K.K. Ang, C. Guan, On the use of convolutional neural networks and augmented CSP features for multi-class motor imagery of EEG signals classification, in: 2015 37th Annual Int. Conf. of the IEEE Engineering in Medicine and Biology Society (EMBC), IEEE, 2015, pp. 2620–2623.
[34] X. An, D. Kuang, X. Guo, Y. Zhao, L. He, A deep learning method for classification of EEG data based on motor imagery, in: Intelligent Computing in Bioinformatics, Springer, Berlin, 2014, pp. 203–210.
[35] H. Yang, S. Sakhavi, K.K. Ang, C. Guan, On the use of convolutional neural networks and augmented CSP features for multi-class motor imagery of EEG signals classification, in: 2015 37th Annual Int. Conf. of the IEEE Engineering in Medicine and Biology Society (EMBC), IEEE, 2015, pp. 2620–2623.
[36] V. Lawhern, et al., EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces, J. Neural Eng. 15 (5) (2018).
[37] G. Muhammad, et al., A facial-expression monitoring system for improved healthcare in smart cities, IEEE Access 5 (2017) 10871–10881, http://dx.doi.org/10.1109/ACCESS.2017.2712788.
[38] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient based learning applied to document recognition, Proc. IEEE 86 (1986) 2278–2324.
[39] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle, Greedy layer-wise training of deep networks, Adv. Neural Inform. Process. Syst. 19 (153) (2006).
[40] C. Brunner, R. Leeb, G. Muller-Putz, A. Schlogl, G. Pfurtscheller, BCI Competition 2008–Graz Data Set A and B, Institute for Knowledge Discovery (Laboratory of Brain-Computer Interfaces), Graz University of Technology, 2008, pp. 136–142.
[41] R.T. Canolty, E. Edwards, S.S. Dalal, M. Soltani, S.S. Nagarajan, H.E. Kirsch, M.S. Berger, N.M. Barbaro, R.T. Knight, High gamma power is phase-locked to theta oscillations in human neocortex, Science 313 (2006) 1626–1628.
[42] H. CR, P. MP, An analysis of the effect of eeg frequency bands on the classification of motor imagery signals, Int. J. Biomed. Soft Comput. Hum. Sci.: Off. J. Biomed. Fuzzy Syst. Assoc. 16 (1) (2011) 121–126.
[43] S. Stober, Learning discriminative features from electroencephalography recordings by encoding similarity constraints, in: Bernstein Conference 2016, 2016.
[44] S.U. Amin, M. Alsulaiman, G. Muhammad, M.A. Bencherif, M.S. Hossain, Multilevel weighted feature fusion using convolutional neural networks for EEG motor imagery classification, IEEE Access 7 (2019) 18940–18950, http://dx.doi.org/10.1109/ACCESS.2019.2895688.
[45] W. Wang, Y. Huang, Y. Wang, L. Wang, Generalized autoencoder: A neural network framework for dimensionality reduction, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2014, pp. 490–497.
[46] M. Chen, et al., Edge-CoCaCo: Toward joint optimization of computation, Caching, and communication on edge cloud, IEEE Wirel. Commun. 25 (3) (2018) 21–27, http://dx.doi.org/10.1109/MWC.2018.1700308.

**Syed Umar Amin** is a Researcher in the Department of Computer Engineering, College of Computer and Information Sciences at King Saud University, Saudi Arabia. He received his Ph.D. degree in Computer Engineering from King Saud University in 2019, Master's degree in Computer Engineering from Integral University, India in 2013. His research interests include Deep Learning, Brain-Computer Interface and Cloud and Multimedia for Healthcare.

**Mansour Alsulaiman** received the Ph.D. degree from Iowa State University, USA, in 1987. Since 1988, he has been with the Computer Engineering Department, King Saud University, Riyadh, Saudi Arabia, where he is currently a Professor in the Department of Computer Engineering. His research areas include automatic speech/speaker recognition, automatic voice pathology assessment systems, computer-aided pronunciation training system, and robotics. He was the Editor-in-Chief of the King Saud University Journal Computer and Information Systems. He is the director of Center of Smart Robotics Research at King Saud University.

**Ghulam Muhammad** is a Professor at the department of Computer Engineering, College of Computer and Information Sciences, King Saud University (KSU), Riyadh, Saudi Arabia. Prof. Ghulam received his Ph.D. degree in Electrical and Computer Engineering from Toyohashi University and Technology, Japan in 2006, M.S. degree from the same university in 2003. He received his B.S. degree in Computer Science and Engineering from Bangladesh University of Engineering and Technology in 1997. He was a recipient of the Japan Society for Promotion and Science (JSPS) fellowship from the Ministry of Education, Culture, Sports, Science and Technology, Japan. His research interests include image and speech processing, cloud and multimedia for healthcare, serious games, resource provisioning for big data processing on media clouds and biologically inspired approach for multimedia and software system. Prof. Ghulam has authored and co-authored more than 200 publications including IEEE/ACM/Springer/Elsevier journals, and flagship conference papers. He has a U.S. patent on audio processing. He received the best faculty award of Computer Engineering department at KSU during 2014–2015. He supervised more than 10 Ph.D. and Master Theses. Prof. Ghulam is involved in many research projects as a principal investigator and a co-principal investigator.

**Mohamed Amine Mekhtiche** was born in Medea Village, Algeria in 1987. He received the B.S. degrees in Electronic Engineering from the University of Blida, in 2010 and the M.S. degrees in Electronic Engineering from the University of Blida, in 2012. From 2014 till now, he is a researcher in Center of Smart Robotic Research in King Saud University in KSA. His current research interests include image processing stereo vision.

**M. Shamim Hossain** received the Ph.D. degree in electrical and computer engineering from the University of Ottawa, Canada. He is currently a Professor with the Department of Software Engineering, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia. He is also an Adjunct Professor with the School of Electrical Engineering and Computer Science, University of Ottawa. He has authored and co-authored approximately 220 publications in refereed journals and conferences and, books, and book chapters. His research interests include cloud networking, smart environment (smart city, smart health), social media, the IoT, edge computing and multimedia for health care, deep learning approach to multimedia processing, and multimedia big data. He is a Senior Member of the IEEE and ACM. He has served as a member of the organizing and technical committees of several international conferences and workshops. He was a recipient of a number of awards, including the Best Conference Paper Award, the 2016 *ACM Transactions on Multimedia Computing, Communications and Applications* Nicolas D. Georganas Best Paper Award, and the Research in Excellence Award King Saud University. He has served as a co-chair, general chair, workshop chair, publication chair, and TPC for over 12 IEEE and ACM conferences and workshops. He is currently the Co-Chair of the second IEEE ICME workshop on Multimedia Services and Tools for smart-health (MUST-SH 2019). He is on the Editorial Boards of the IEEE Transactions on Multimedia, the IEEE NETWORK, the IEEE MULTIMEDIA, the IEEE WIRELESS COMMUNICATIONS, the IEEE ACCESS, the *Journal of Network and Computer Applications* (Elsevier), *Computers and Electrical Engineering* (Elsevier), *Human-centric Computing and Information Sciences* (Springer), *Games for Health Journal*, and the *International Journal of Multimedia Tools and Applications* (Springer). He also serves as a Lead Guest Editor for the IEEE NETWORK, *Future Generation Computer Systems* (Elsevier), and the IEEE ACCESS. Previously, he served as a Guest Editor of the *IEEE Communications Magazine*, the IEEE TRANSACTIONS ON INFORMATION TECHNOLOGY IN BIOMEDICINE (currently JBHI), the IEEE TRANSACTIONS ON CLOUD COMPUTING, the *International Journal of Multimedia Tools and Applications* (Springer), *Cluster Computing* (Springer), *Future Generation Computer Systems* (Elsevier), *Computers and Electrical Engineering* (Elsevier), *Sensors* (MDPI), and the *International Journal of Distributed Sensor Networks*. He is a senior member of both the IEEE, and ACM.