Available online at http://scik.org

Commun. Math. Biol. Neurosci. 2021, 2021:4

https://doi.org/10.28919/cmbn/5265

ISSN: 2052-2541

DEVELOPMENT OF LSTM&CNN BASED HYBRID DEEP LEARNING MODEL TO CLASSIFY MOTOR IMAGERY TASKS

CAGLAR UYULAN*

Mechatronics Engineering Department, Bülent Ecevit University, Zonguldak, Turkey

Abstract: Recent studies underline the contribution of brain-computer interface (BCI) applications to the enhancement process of the life quality of physically impaired subjects. In this context, to design an effective stroke rehabilitation or assistance system, the classification of motor imagery (MI) tasks are performed through deep learning (DL) algorithms. Although the utilization of DL in the BCI field remains relatively premature as compared to the fields related to natural language processing, object detection, etc., DL has proven its effectiveness in carrying out this task. In this paper, a hybrid method, which fuses the one-dimensional convolutional neural network (1D CNN) with the long short-term memory (LSTM), was performed for classifying four different MI tasks, i.e. left hand, right hand, tongue, and feet movements. The time representation of MI tasks is extracted through the hybrid deep learning model training after principal component analysis (PCA)-based artefact removal process. The performance criteria given in the BCI Competition IV dataset A are estimated. 10-folded Cross-validation (CV) results show that the proposed method outperforms in classifying electroencephalogram (EEG)-electrooculogram (EOG) combined motor imagery tasks compared to the state of art methods and is robust against data variations. The CNN-LSTM classification model reached 95.62 % (±1.2290742) accuracy and 0.9462 (±0.01216265) kappa value for datasets with four MI-based class

*Corresponding author

E-mail address: caglaruyulan@beun.edu.tr

Received November 23, 2020

validated using 10-fold CV. Also, the receiver operator characteristic (ROC) curve, the area under the ROC curve (AUC) score, and confusion matrix are evaluated for further interpretations.

Keywords: brain-computer interface; motor imagery; deep learning; long-short term memory; convolutional neural network; artificial intelligence; classification; principal component analysis; stroke rehabilitation.

2010 AMS Subject Classification: 92B25, 37N25, 68T05.

1. INTRODUCTION

The use of assistive technologies to help disabled people is importantly increasing in recent years and researchers propose inspirational new scientific methods for restoring functions to those with motor impairments such as paralysis, amyotrophic lateral sclerosis, cerebral palsy, loss of limb [10]. Recent neuroscience and robotic studies indicate that even the imagination of a movement generates the same mental pattern as the performance of the movement itself [13]. Thus, transforming a brain activity or a task signal into direct control of any hardware device without the involvement of the peripheral nervous system or muscle is applicable. Therefore, it becomes promising for subjects suffering from mobility reduction. The techniques that enable the researchers to translate and interpret the brain signals as physical tasks are referred to as BCI [44]. BCI application provides a means of non-muscular communication and control paradigm to transmit signals&commands from the individuals with severely impaired movement to the external world or devices by measuring brain activity.

BCI technology is broadly composed of five consecutive processes, which are sequenced as; *signal acquisition (1), extraction of the intended action or desired features from the task (2), selection of more relevant subset from the feature set (3), classification of the mental state (4), and finally, feedback signals generated by the prosthetic device (5)* [38]. These brain signals are extracted, decoded and studied with the help of various imaging techniques like EEG, EOG, magnetoencephalography (MEG), positron emission tomography (PET), functional magnetic resonance imaging (fMRI), electrocorticography (ECoG), etc.[11].

But, since these techniques involve sophisticated and expensive equipment and therefore the availability and utilization are mostly assigned to high budget corporations or hospitals, the traditional measurement of brain activity, in the context of EEG-based BCI, has relied on the acquisition of EEG data via non-invasive electrode arrays. The acquired EEG data is analyzed in either the time or frequency domain dynamics and subsequently translated into corresponding control commands. A BCI technology aims at decoding characteristic brain modalities, which are generated from various brain locations, to control a robotic device. In the non-invasive method, the neurophysiological rhythms are recorded by sensors placed over the scalp to avoid the risks of surgery with a trade-off of low signal-to-noise (SNR) [22].

MI-related BCI, which is based on the imagination of the execution of movements is widely implemented. MI stimulates similar brain pathways as if it was performed a real movement. It replaces the exercises in cases where there is no residual motor function [48]. Several studies in the literature have focused on classifying MI-EEG signals to provide feedback during MI training [46; 65; 1]. While performing MI tasks, the synchronized rhythmic variations called event-related synchronization (ERS) and event-related desynchronization (ERD) in sensorimotor regions can be captured from EEG [24].

In the literature, several feature extraction and machine learning (ML)-based classification techniques are used for EEG-based BCI. Generally, feature extraction methods for the MI-EEG focus on deriving time-domain features, i.e. energy and amplitude of the signal, autoregressive modelling [17; 2], and on establishing frequency domain features [31; 37] or on extracting time-frequency features [59; 58]. With its adaptive structure and the ability for analyzing the non-stationary signals wavelet transform (WT) keeps and processes both time and frequency components of the signal. The combination of useful frequency and time information on the non-stationary EEG signal improves the performance of classifiers. Stating the merit of potential biomarkers is a critical threshold contributing to the classification performance. Therefore, with the proliferation of high-dimensional data, feature selection (FS) methods have been widely applied as a vital task before the learning process. The purpose of using FS methods is selecting a valuable subset of features from the original set of features without sacrificing from the accuracy in representing the initial set of features, in which plenty of spurious information and irrelevant features exist [52]. Extensive implementation of wrapper-based approaches is also underlined, particularly in bioinformatics, employing genetic algorithm (GA) [35], particle swarm optimization (PSO) [47], ant colony optimization (ACO) [16], etc. Besides, an increasing number of researches make use of the embedded capacity of several classifiers to discard less informative input features. ML-supported classification techniques such as quadratic discriminant analysis (QDA), linear vector quantization (LVQ), k-nearest neighbour (KNN), multilayer perception (MLP), support vector machine (SVM), linear discriminant analysis (LDA), decision tree, naive Bayesian classifier for EEG-MI classification have been widely studied [18; 14; 54; 62]. Since the brain signals recorded using EEG have non-linear, complex, non-stationary and non-Gaussian nature, finding a robust and accurate feature extraction and ML-based classification method is a challenge in EEG-based BCI application. To improve the effectiveness of the classifiers, the specialized pre-processing, artefact removal, feature reduction techniques can be used [23; 26; 63]. PCA is one of these techniques. With the aid of PCA, the higher dimensions of the signal, which contains relatively insignificant or insensitive features, are reduced to lower dimensions to increase the correctly classified percentage of data. It tries to find the "best" eigenvalues in the sense of variances while accounting the temporal variability. This means, that it is sought to extract the most dynamic one, but this does not necessarily mean that these features are the most prominent ones. Therefore DL algorithms should be applied for solving this problem.

DL is a prediction method, which uses a sequence of nonlinear processing stages, which jointly learns from data. It serves a new way of neural network (NN) fitting approach with hierarchical feature extraction and helps to find the representations that are invariant to inter-and intra-subject differences while reducing dimensions, in this way it is possible to construct a unified end-to-end model that can be applied to raw signals [4].

Deep NN's have specialized and proved their effectiveness in recognition tasks including applications of images, videos, speech, and text classification. CNN's are very suitable to study with images and video data because they are capable to extract representative features, which are robust to partial translation and deformation of inputs [30; 5]. CNN's are also effective in many applications, which comprise temporal dynamics such as, handwriting, speech recognition. Additionally, CNN's are utilized in the field of the combination of spatial representation and time-series structure, i.e. moving object detection or video classification [40; 33]. CNN's provide significant performance enhancement minimizing the error rates of competing techniques in ImageNet competition 2012 [28].

In this paper, a continuous classification output for each sample in the form of class labels of MI tasks (0 (Left Hand), 1 (Right Hand), 2 (Feet), 3 (Tongue)) including labelled trials was provided by implementing CNN-LSTM based deep classifier based on the BCI Competition IV dataset A. The classification algorithm is causal, meaning that the classification output at time k may only depend on the current and past samples $x_k, x_{k-1}, \dots, x_0$. A confusion matrix was then built from all artefact-free trials for each time point. The time course of the accuracy and loss was obtained. The mean and standard values of the validation accuracy and validation kappa value after 10-fold CV are computed, respectively. The confusion matrix and ROC curve were plotted, and the AUC score is evaluated.

1.1.RELATED STUDIES

A unified end-to-end CNN-based DL model was developed to classify MI-related tasks. Transfer learning was used to adapt the global classifier to single individuals improving the overall mean accuracy. However, in this study, the classification performance for four class is quite low with 68.51% [15]. A novel approach for learning deep representations from multi-channel EEG time-series was proposed and its advantages in the mental load classification conceptualization were demonstrated. A deep recurrent-convolutional network was trained by mimicking the video classification techniques. As a result, the spatial, spectral, and temporal dynamics of EEG were preserved and mental load classification performance and robustness were improved [5]. A tensor-based multiclass multimodal scheme for hybrid BCI was developed to generate nonredundant tensor components. Multimodal discriminative patterns were selected through a weighted fisher criterion and support vector machine (SVM) was used for multiclass classification. The main advantage of this method is to capture the interactive effects of simultaneous tasks, but tensor generation and decomposition processes are very time-consuming [25]. Deep ConvNets with a range of various architectures including batch normalizations, exponential linear units, cropped training strategy were designed for decoding imagined or executed movements acquired from raw EEG. This method was compared with a validated baseline method named as filter bank common spatial patterns (FBCSP) decoding algorithm. One of the important findings of this study is that the deep ConvNets are learned features different from FBCSP, which could explain their higher accuracies in the lower frequencies where band power may be less important. Deep ConvNets can learn band power features with specific spatial distributions from the raw input in an end-to-end manner [51]. A novel MI classification framework was introduced by building a new 3D representation of EEG and training a multi-branch 3D CNN. Experimental evaluations reveal that the framework reaches state-of-the-art kappa value and outperforms other algorithms by a 50% decrease in the standard deviation of various subjects in terms of robustness criteria [66]. A CNN was employed to classify and characterize the error-related brain response as measured in 24 intracranial EEG recordings. It was found that the decoding accuracies of CNNs were higher than those of regularized linear discriminant analysis. The 4-layered CNNs were able to learn in all-channel decoding of errors from intracranial EEG electrodes in epilepsy patients [60]. It was proved that the CNN and LSTM capacity to learn high-level EEG features consisted of low-level ones, after feature extraction by discrete wavelet transform (DWT). The CNN and LSTM schemes are suitable and relatively roust to the BCIs and MI-EEG decoding [64].

2. THEORETICAL BACKGROUND

2.1.CONVOLUTIONAL NEURAL NETWORK

CNN’s are a kind of NN having a grid-like topology, which is specialized for processing data. The CNN’s are capable of extracting spatial features to a granular level and these features perform a high discrimination power for classification issues. A typical CNN architecture consists of three layers: convolution, pooling, and fully connected. Each layer comprises filter windows that slide over the input layer from the preceding layer.

While time-series data represents a 1-D grid, image data express 2-D grid nature comprising pixels. They take their name from a mathematical operation defined as *"convolution operator"*. CNN's use the convolution operator instead of the general matrix multiplication in at least one of its layers [21]. The convolution of two discrete signals x_n and ω_n is given as $x_n \otimes \omega_n = \sum_{m=-\infty}^{\infty} x_m \omega_{n-m}$, where $\otimes$ corresponds to the convolution operator. The generalized architecture of CNN is demonstrated in Fig.1.

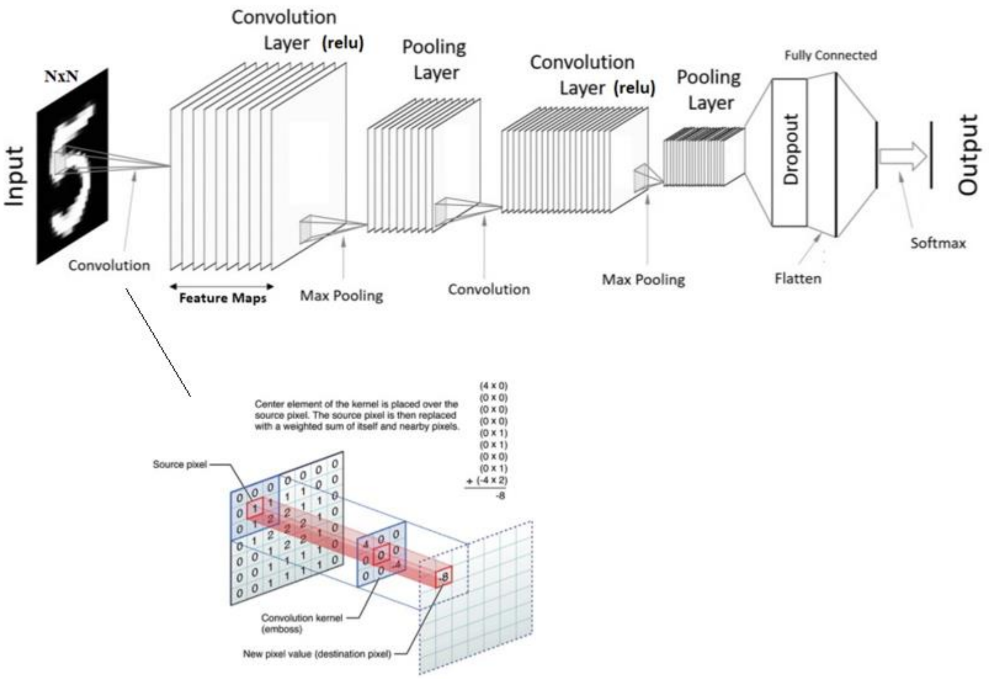

Figure 1: The generalized architecture of CNN's.

Each neuron in the first layer of the CNN interacts only with a small region of the input neurons, which is defined as a convolution window (Fig.1). While the convolution window is passing through an entire input sequence, each neuron in the hidden layer learns by changing its connection weight and overall bias. The size of the convolutional window is named as the *"kernel size, k"*. The mathematical expression of this process is given as in Eq.1

$$a_{ij} = \sigma\left(b_i + \sum_{k=1}^{l} \omega_{ik} x_{j+k-1}\right) \tag{1}$$

where a_{ij} is the output of the j^{th} neuron of the i^{th} filter in the hidden layer, b_i denotes to the overall bias of filter i, ω corresponds to the shared weights and $\sigma(.)$ is the nonlinear activation function.

It can be deduced that if it is given a finite kernel size (k), the input to a specific neuron only relies on the subspace from the previous layer. This implies the sparse connectivity [43]. The sparse connections and weight sharing attribute extremely reduce the number of weights to be learned and shorten the training process by decreasing the gradient computation process. In this way, complex and high-level features are possible to be learned while preventing overfitting. Multiple filtered forms of the input data take place in the stacked hidden layers of the CNN as feature maps. Filter size, strides (sliding of windows), and padding (window offset over input) settings are parameterized.

Another operation is named as *"pooling"*. Through the pooling operation, high-dimensional input space is gradually confined to a low dimensional space between layers by maximizing or averaging, etc its neighbouring values in the feature map. The location independence of the model is increased because the feature in various positions can be mapped to the same feature through the help of the aggregation of adjacent neurons [53]. *"Fully connected layer"* structure behaves same as ANN. Each neuron is connected to every other neuron of the preceding layer. The theoretical basics of the CNNs and learning theories can be investigated more detailed from [32; 8; 39].

2.2.LONG-SHORT TERM MEMORY

LSTM networks are the modified version of the recurrent neural networks (RNN). RNNs are specialized to process sequential data ($x^{(1)}, x^{(2)}, \dots, x^{(\tau)}$), which plays an over time such as speech, language, audio, video etc. RNNs can scale long sequences and also process sequences of variable length and context. RNNs remember essentials related to the input signal through an internal memory while enabling a prediction of next states.

The main difference of the LSTM is that the gradient can flow for long durations. A crucial modification has been to make the weight on the self-loop conditioned on the context, rather than fixed. The time scale of integration is adjusted by doing the weight of self-loop gated based on the input sequence because the time constants are model output [19]. The architecture of the LSTM is demonstrated in Fig.2.

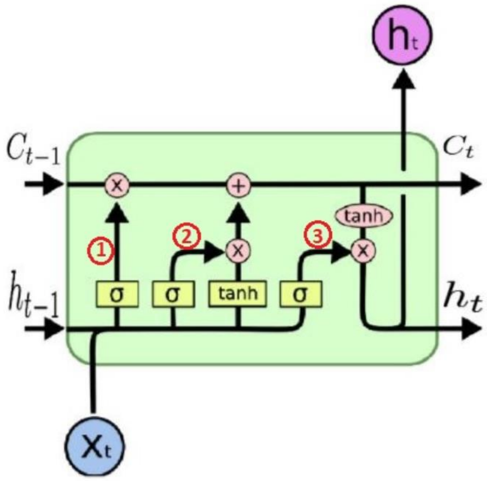

Figure 2: LSTM architecture. 1) Forget gate, 2) Input gate, 3) Output gate

According to Fig.2, $\times$ is the scaling of information, $+$ is adding information, σ is the sigmoid layer, which is used as a memory for remembering or forgetting, $tanh$ is the activation function, which is used to solve the gradient vanishing problem, $h(t)$ corresponds to the output of LSTM unit, $c(t-1)$ denotes the memory from previous LSTM unit, $X(t)$ is input, $c(t)$ represents new updated memory. The path from $c(t-1)$ to $c(t)$ is defined as a memory pipeline. Forget gate takes $X(t)$ and $h(t-1)$ as input and decides whether to forget or not the incoming information. Input gate decides what information is stored in memory and output gate choose what information becomes an output.

3. Materials and Methods

3.1.EXPERIMENTAL PARADIGM

BCI Competition IV 2008-Graz data set A provided by the Institute for Knowledge Discovery (Laboratory of Brain-Computer Interfaces, Graz University of Technology), is used in the deep learning-based modelling process. The data set comprises of EEG data from 9 subjects. The BCI paradigm has been established based on the four different motor imagery tasks (imagination of the movement of the left hand (class 0), right hand (class 1), both feet (class 2), tongue (class 3)). The session is comprised of 12 runs having 48 trials (12 for each of the four existing classes) and are dichotomised by short breaks. 576 trials are conducted in total.

At first, a recording of approximately 5 minutes was run for estimating EOG interference. The recording is constructed by three-phase [1) two minutes with eyes open, 2) one minute with eyes closed, and 3) one minute with eye movements]. The details of the data acquisition process are represented in [55].

3.2.DATA ACQUISITION AND PREPROCESSING

EEG signals were acquired by utilizing twenty-two (22) Ag/AgCl electrodes having inter-electrode distances of 3.5 cm. The montage which maps the EEG and EOG channels are depicted in Fig3. a and Fig3.b, respectively.

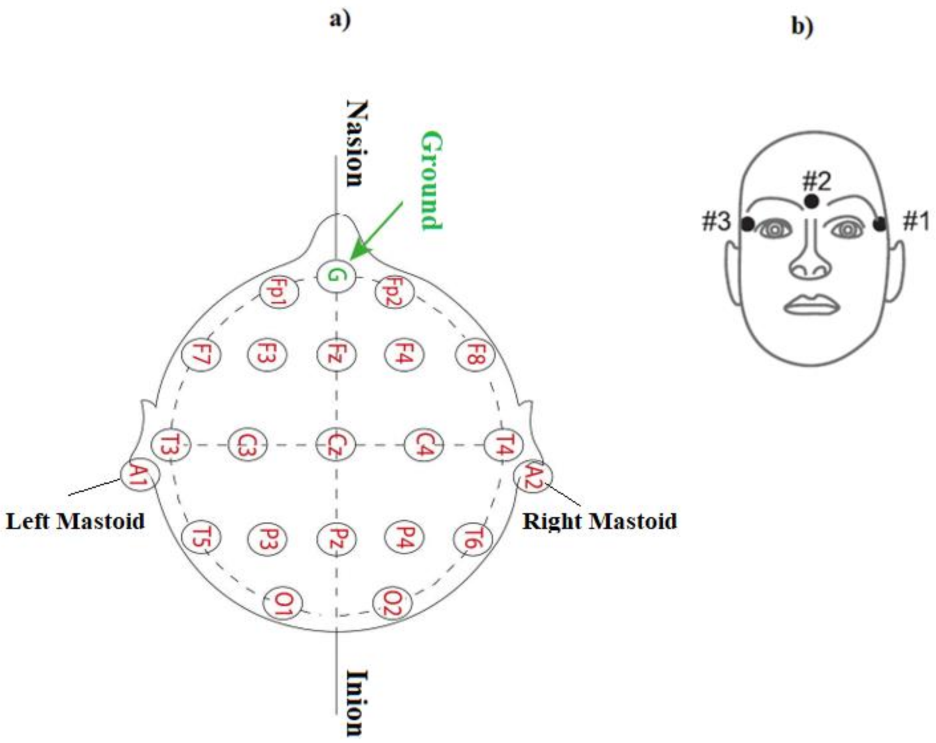

Figure 3: a) Electrode montage of the twenty-two (22) channel EEG device. **b)** Electrode montage of the three (3) monopolar EOG channels.

All signals (EEG and EOG) were collected monopolar with the left mastoid as a reference, and the right mastoid as ground. The signal sampling rate is 250 Hz. The signals were filtered with a 0.5-100 Hz. band-pass filter, and with an additional 50 Hz. notch filter, which suppresses the line noise. The sensitivity of the EEG amplifier was set to 100 μV. The sensitivity of the EOG amplifier was set to 1mV. The details about the data file description may be accessed from [7]. The data collected from 9 subjects for each mental task was cropped to a total of 196500 sample to equalize the length of the data after NaN values are cleaned. The total size of the processed data is reduced to [196500*4 (sample), 25 (channel-EEG+EOG)] matrix form. After obtaining the final matrix form, the data is standardized by removing the mean and scaling to unit variance. Through the centring and scaling process, the relevant statistics are computed on the samples in the training set. Mean and standard deviation are then stored to be utilized with the transformation. The standardization process should be applied before classification for enhancing the performance of the machine learning-based estimator. They might give poor resuşts when the individual features do not more or less look like standard normally distributed data (e.g. Gaussian with zero mean and unit variance) [12].

3.3. ARTIFACT REMOVAL AND DIMENSIONALITY REDUCTION

The signal and noise (both random and deterministic) should be separated to obtain a high-quality measurement. Random noises can be eliminated by repeating a single signal measurement process. Some artefacts originate from the measurement process itself, therefore a statistical analysis of sensors is necessitated [29]. The characteristics of the artefacts differ from the signal of interest. When the artefact is limited to a specific frequency range, it can be removed by a frequency filtering approach. If it is based on discrete frequencies or their harmonics, it can be eliminated by notch filtering. When it is limited to a certain time range, e.g. in the case of eye blinks, the time intervals are discarded by observing the signal. However, if the artefacts are originated from various sources or a limited to a subspace of the signal space, the topography of the artefacts exhibit superposition state. For this reason, the artefacts can be eliminated by utilizing signal-space projection [57], so that the remaining signals do not include artefact subspace [50]. Methods, i.e. ICA [61] or PCA, based on the assumption that artefacts and signal sources are sufficiently independent of each other are also proposed [27; 41]. It is also possible to define the artefact by a particular temporal pattern, i.e. exponential decay. The artefacts are also modelled through fitting its parameters to the data, and then it is removed from the physiological signal, i.e. frequency range, amplitude, etc. The other methods include regression techniques, FIR filters, wavelet denoising, denoising with multilevel wavelet DWT functions, etc. Several artefact removal methods can also be united and run as stand-alone or EEGLAB plug-in. [45]. The suitability of artefact rejection method depends on not distorting the main component or feature sought in the signal. This is especially the case for some automated methods such as (independent or principle) component analysis and some filtering methods. The spatiotemporal nature and the generation mechanisms of the artefacts should be investigated. It is possible to reduce the effects of the muscular and eye-movement artefacts due to their structural difference from simultaneous reference signals such as EOG, ECG, EMG etc. When the large transient artefacts, e.g. from electric stimulation are expected, the recording epoch should be wide enough for confining artefacts in time [56; 42]. The multi-class classification performance of the BCI has also been improved with the current source density (CSD) method, which depends only on the position of the sensors on the scalp [49].

In light of the above basic information, in this paper, the PCA method was preferred to remove artefacts. PCA serves the speed-boosting of the fitting of the classifier by dimensionality reduction. PCA converts data linearly into new features that are not correlated with each other by doing the orthogonal transformation [34].

3.4. CNN-LSTM FRAMEWORK

The classifier utilized in this paper is the combination of CNN and LSTM. First, the time-domain features of the EEG data are extracted through 1D-CNN, and after that, these features are feeding into the LSTM to obtain high-level representative features. Finally, the classifier dichotomizes four MI tasks. The framework of the methodology is depicted in Fig.4.

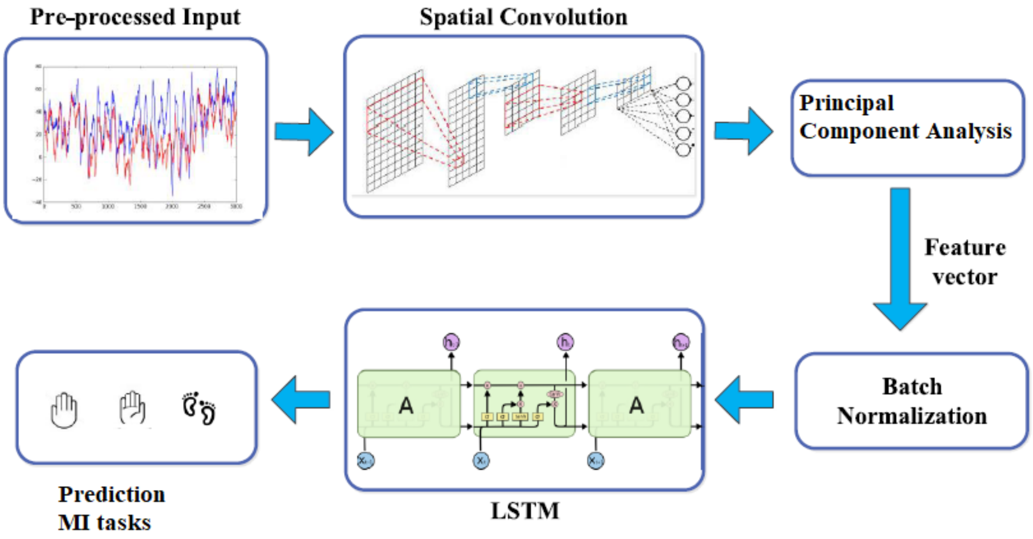

Figure 4: The framework of the classification methodology.

ERD and ERS can be seen in $\mu - band\ (8 - 13\ Hz.)$ and $\beta - band\ (13 - 30\ Hz.)$. For this reason, the μ and β bands are extracted through the fifth-order butter-worth filter. In this way, it is possible to benefit from band power optimally. In the experiment, the subjects performed MI tasks in 3 seconds, the time segment of 2-6 seconds are extracted to reduce the temporal redundancy.

Multichannel EEG data is two-dimensional, but time and channel have different units, which drives a non-trivial selection of the filter kernel dimensions. The model is based on a hybrid CNN-LSTM structure and the summary of the model is given in Table 1.

Table 1: Model summary of the CNN-LSTM.
Model: "CNN-LSTM"

Layer (type)	Output Shape	Parameters #
conv1d (Conv1D)	(N, 122, 128)	2176
max_pooling1d (MaxPooling1D)	(N, 30, 128)	0
conv1d_1 (Conv1D)	(N, 29, 64)	16448
lstm (LSTM)	(N, 29, 25)	9000
lstm_1 (LSTM)	(N, 29, 25)	5100
lstm_2 (LSTM)	(N, 25)	5100
dense (Dense)	(N, 4)	104

Total params: 37,928

Trainable params: 37,928

Non-trainable params: 0; N refers to the length of the input.

The timestep is selected as 125, which means the input shape of the first convolutional layer is (125, 4). The kernel size and the filter number of the first 1-D CNN are 4 and 128, respectively.

The padding is activated with 1 stride, and the activation function is selected as “ReLu”. MaxPool size is 4. The second 1-D CNN has 64 filters with a kernel size of 2. LSTM layers have “tanh” activation functions and recurrent activation functions with a dropout and recurrent dropout of 0.2. The total unit of each LSTM layer is 25. The dense layer has “softmax” activation function.

The hyperparameters of the model are represented in Table 2.

Table 2: The hyperparameters of the model.

parameter	**type or value**
optimizer	Adam
activation	ReLu, tanh, softmax
regularization	dropout
loss function	categorical cross-entropy
batch size	64
epoch	400

Dropout, which a determined portion of the neuron is randomly turned off at each iteration, is necessary for model generalization. Dataset was shuffled at each epoch to avoid overfitting.

4. RESULTS

All steps are executed by the publicly available *“Google Colaboratory”*, which is a free cloud service giving an opportunity to AI developers to apply their deep learning-based algorithms. Datasets are trained on “Google Colaboratory” including several significant modifications, which allows evaluations on multiple GPUs. The GPU model is Tesla k80 supporting Python environment and Keras deep learning libraries. It is easy to upload data from “Google Drive Application” to train the model. Multi-GPU training exploits data parallelism and is carried out by splitting each batch of training images into several GPU batches, processed in parallel on each GPU. The gradient of the full batch is obtained by averaging the computed GPU batch gradients. Gradient computation is synchronous across the GPUs, so the result is precisely the same as when training on a single GPU. First, the filtered datasets are uploaded to “Google Colaboratory”. The size of the dataset matrix is [195000*4 (time-series representing each mental task), 25 (EOG+EEG channels]. Then, the dataset is normalized via StandardScaler command in the sklearn. preprocessing library, and is subjected to the PCA. After the PCA process, the electrode sources are reduced into four main principal components. The obtained size of the dataset matrix becomes [195000*4, 4 (principal component)]. After that, the dataset matrix is divided into timesteps each part includes 125 samples and reshaped as [6288, 125, 4]. Finally, Datasets are split randomly into the train part and test part, for the fitting via sklearn.model_selection.train_test_split command with parameters of test size 0.1, and random state 0.2. After splitting process, the size of the train and test data matrices are [5659, 125, 4], [629, 125, 4], respectively. The CNN-LSTM model is trained with the 10-fold CV method by using train data. This procedure was done by splitting the training dataset into 10 subsets and takes turns training models on all subjects except one which is held out, and computing model performance on the held-out validation dataset. In this paper, 10 models are build and evaluated for CV [37]. For each trial, a sliding window of size 125 along the time axis. The models obtained from the training phase are tested section-by-section, and finally, the average validated the accuracy and kappa value obtained for each section having epochs=400 and batch_size=64.

The results of the 10-fold CV process are represented in Table 3.

Table 3: Performance table of the 10-k CV process.

Trials	Accuracy (Train)	Loss(Train)	Accuracy (Validation)	Loss (Validation)	Kappa Value (Validation)
1	0.9806	0.0549	0.9346	0.2926	0.9249
2	0.9751	0.0745	0.9611	0.1645	0.9511
3	0.9786	0.0590	0.9788	0.0882	0.9686
4	0.9827	0.0497	0.9541	0.2191	0.9442
5	0.9857	0.0442	0.9611	0.1419	0.9511
6	0.9865	0.0366	0.9470	0.2218	0.9371
7	0.9863	0.0381	0.9611	0.1977	0.9511
8	0.9868	0.0403	0.9664	0.1489	0.9563
9	0.9857	0.0451	0.9505	0.2625	0.9406
10	0.9813	0.0527	0.9470	0.1745	0.9371

Average Train Accuracy	**Average Train Loss**	**Average Validation Accuracy**	**Average Validation Loss**	**Average Validation Kappa Value**
0.9829(±0.003980801)	0.0495(±0.011465644)	0.9562(±0.012290742)	0.1912(±0.060500892)	0.9462(±0.01216265)

The mean and standard values of the validation accuracy and validation kappa value after 10-fold CV are evaluated as 95.62 % (± 1.2290742) and 0.9462 (± 0.01216265), respectively. After that, the CNN-LSTM model was tested by using test data. The accuracy and kappa value were obtained as 96.98 % and 0.9597, respectively. The disadvantage of the k-fold CV is that the size of the train test splits is predetermined. The metric of the kappa score evaluation is *"cohen_kappa_score",* which is a statistic that measures inter-annotator agreement. The details and theoretical basis are explained in [9; 3]. The metrics are implemented to measure classification performance by sklearn.metrics module.

The model accuracy and model loss values of test and train data for each epoch was represented in Fig.5a and Fig.5b, respectively.

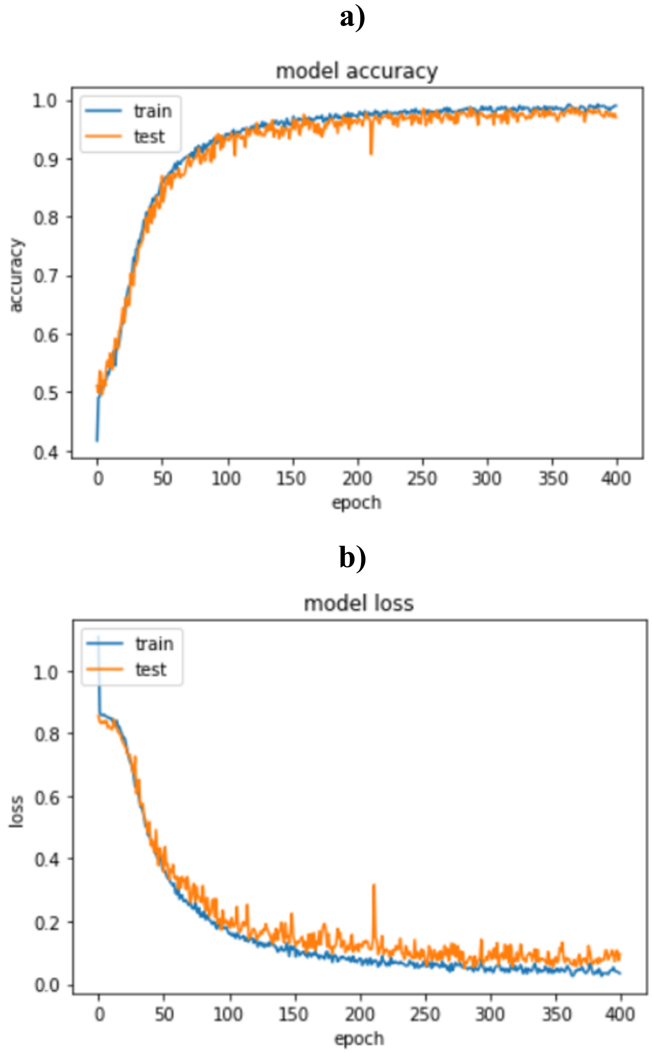

Figure 5: The performance metrics of the model obtained from test and train data a) Model accuracy b) Model loss

The classification accuracy is computed from the confusion matrix with each row corresponding to the true class. In Fig.6, the test confusion matrix was plotted.

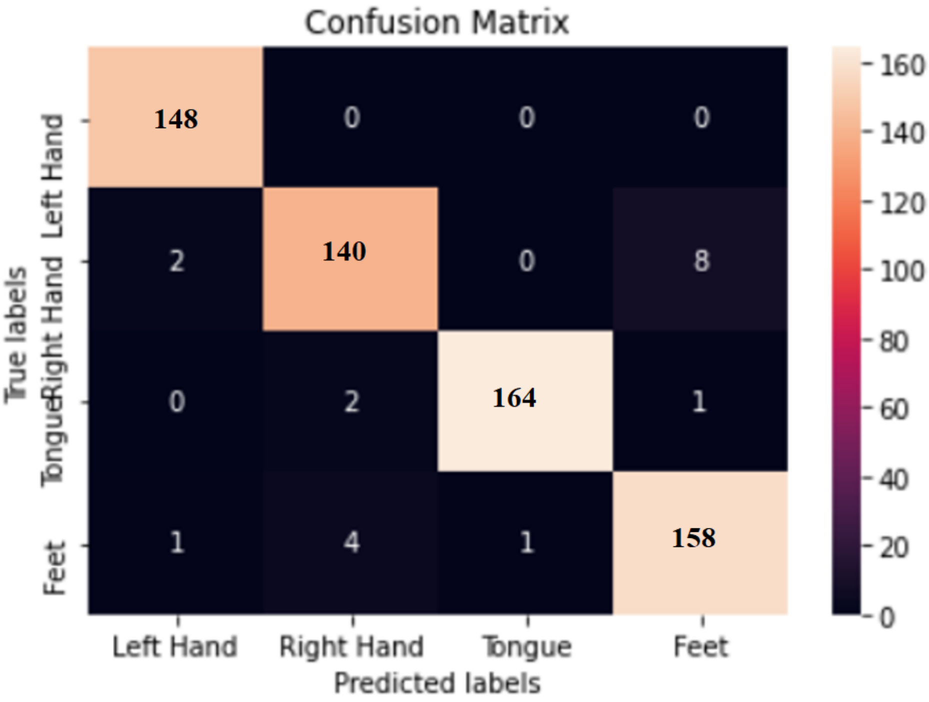

Figure 6: The confusion matrix representing each MI class obtained from test data.

According to Fig.6, the diagonal elements demonstrate the number of points for which the predicted label is equal to the true label, while off-diagonal elements are those that are mislabeled by the classifier. The higher diagonal values of the confusion matrix the better, showing many correct predictions.

When target classes are not balanced, the accuracy metric may not be the right measure. Therefore, the additional metrics like Precision, Recall, F Score etc., should be considered. In Table 4, the results corresponding to these metrics are tabulated.

Table 4: Additional metrics showing the classifier performance.

	precision	***recall***	***f1-score***	***support***
Left Hand	*0.98*	*1.00*	*0.99*	*148*
Right Hand	*0.96*	*0.93*	*0.95*	*150*
Tongue	*0.99*	*0.98*	*0.99*	*167*
Feet	*0.95*	*0.96*	*0.95*	*164*
macro avg	*0.97*	*0.97*	*0.97*	*629*
weighted avg	*0.97*	*0.97*	*0.97*	*629*

A macro-average compute the metric independently for each class and then take the average, whereas a micro-average aggregate the contributions of all classes to evaluate the average metric. The ROC curve was also plotted in Fig.7. The ROC curve indicates the "True Positive Rate" (Sensitivity/Recall) against the "False Positive Rate" (1-Specificity) at various classification thresholds.

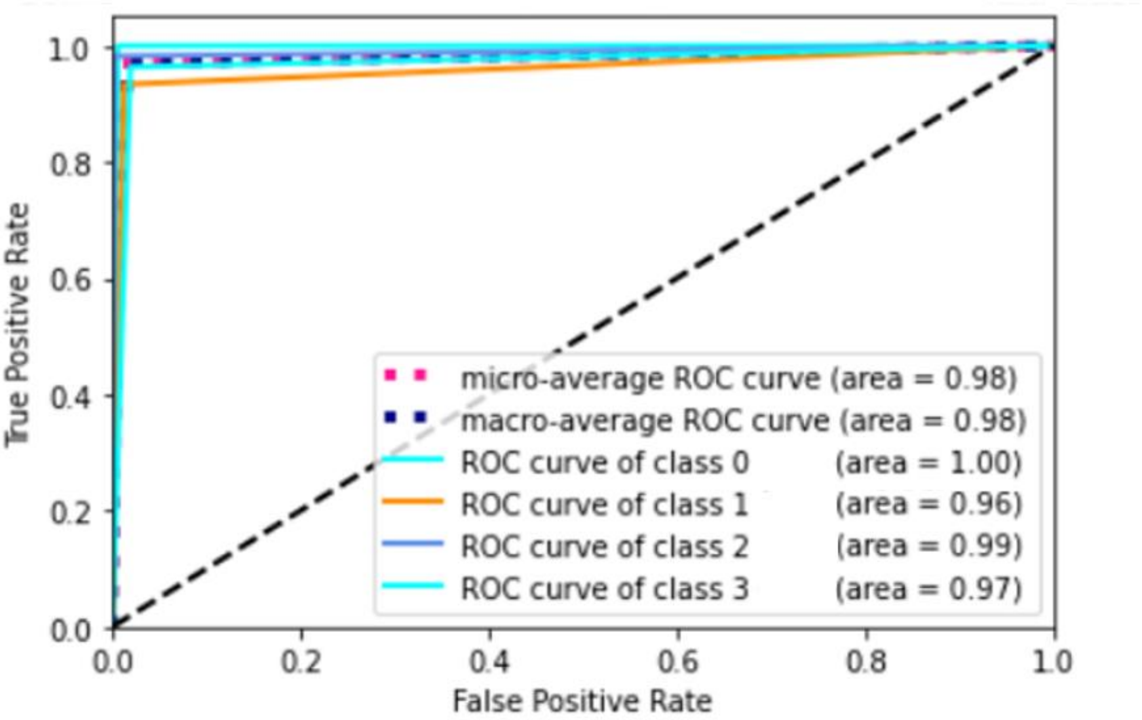

Figure 7: ROC curve. Class 0: Left Hand, Class 1: Right Hand, Class 2: Tongue, Class 3: Feet

ROC curve is adapted to multi-label classification case by binarizing the output. One ROC curve can be drawn per label, but one can also draw a ROC curve by considering each element of the label indicator matrix as a binary prediction (micro-averaging).

AUC measures the entire two-dimensional area underneath the curve. How well a parameter can distinguish among two diagnostic groups can be measured by evaluating the AUC score. The AUC score is estimated as 0.9798126815598498. AUC provides an aggregate measure of performance across all possible classification thresholds. Interpreting AUC is as the probability that the model ranks a random positive example more highly than a random negative example. AUC is scale-invariant and classification-threshold-invariant [20].

5. Conclusive Summary and Discussion

The performance of the CNN-LSTM model was evaluated robustly by 10-Fold CV. According to the results, the hybrid CNN-LSTM model achieved a quite satisfactory and reliable accuracy and kappa value. The predictive model has also extracted the most relevant information from the beginning and the end of the imagined movements. The choice of such a recurrent model depends on the requirement of increasing prediction accuracy, assuming that there is never an abruptly change of movement type in the given experiment. The robustness against overfitting was regularized by adding dropout and pooling layers, doing k-fold CV, and making a batch learning process, which averages over 64 samples from various subjects in each training step. This framework can achieve superior performance in MI classification tasks, and the robustness on different subjects can be improved with appropriate filtering and initial weights. The results are comparable with the reported accuracy values in related studies and the designed CNN-LSTM architecture outperforms the results in the literature [55; 6] on the same underlying data given that the model can learn features from data without necessitating a specialized feature extraction methods.

As a result of this study, it is highly recommended to utilize the combination of the CNN-LSTM based DL architecture for building a BCI system. It is worth investigating the possibilities of LSTM for any real-time sequence classification having online feedback.

Conflict of Interests

The authors declare that there is no conflict of interests.

References

[1] S. Aggarwal, N. Chugh, Signal processing techniques for motor imagery brain computer interface: A review, Array. 1–2 (2019), 100003..

[2] M. Ahmad, M. Aqil, QR decomposition based recursive least square adaptation of autoregressive EEG features, in: 2016 International Conference on Intelligent Systems Engineering (ICISE), IEEE, Islamabad, Pakistan, 2016: pp. 141–145.

[3] R. Artstein, M. Poesio, Inter-Coder Agreement for Computational Linguistics, Comput. Linguist. 34 (2008), 555–596.

[4] A. Bashar, Survey On Evolvıng Deep Learnıng Neural Network Archıtectures, J. Artif. Intell. Capsule Networks 1(2) (2019), 73–82.

[5] P. Bashivan, I. Rish, M. Yeasin, N. Codella, Learning Representations from EEG with Deep Recurrent-Convolutional Neural Networks, ArXiv:1511.06448 [Cs]. (2016).

[6] G. Blanchard, B. Blankertz, BCI Competition 2003—Data Set IIa: Spatial Patterns of Self-Controlled Brain Rhythm Modulations, IEEE Trans. Biomed. Eng. 51 (2004) 1062–1066.

[7] C. Brunner, R. Leeb, G. Mller-Putz, A. Schlögl, G. Pfurtscheller, BCI Competition 2008Graz data set A. Institute for Knowledge Discovery (Laboratory of Brain-Computer Interfaces), Graz University of Technology, Graz, pp 136–142. (2008).

[8] F. Chollet, Deep Learning with Python, Manning Publications, New York, NY, 2017.

[9] T.O. Kvalseth, A Coefficient of Agreement for Nominal Scales: An Asymmetric Version of Kappa, Educ. Psychol. Measure. 51 (1991), 95–101.

[10] R.E. Cowan, B.J. Fregly, M.L. Boninger, L. Chan, M.M. Rodgers, D.J. Reinkensmeyer, Recent trends in assistive technology for mobility, J. NeuroEng. Rehabil. 9 (2012), 20.

[11] B. Crosson, A. Ford, K.M. McGregor, M. Meinzer, S. Cheshkov, X. Li, D. Walker-Batson, R.W. Briggs, Functional imaging and related techniques: An introduction for rehabilitation researchers, J. Rehabil. Res. Develop. 47 (2010), vii- xxxiii.

[12] P. Dangeti, Statistics for machine learning: build supervised, unsupervised, and reinforcement learning models using both Python and R, Packt Publishing, Birmingham, UK, 2017.

[13] J. Decety, M. Lindgren, Sensation of effort and duration of mentally executed actions, Scand. J. Psychol. 32 (1991), 97–104.

[14] R. Djemal, A. Bazyed, K. Belwafi, S. Gannouni, W. Kaaniche, Three-Class EEG-Based Motor Imagery Classification Using Phase-Space Reconstruction Technique, Brain Sci. 6 (2016), 36.

[15] H. Dose, J.S. Møller, H.K. Iversen, S. Puthusserypady, An end-to-end deep learning approach to MI-EEG signal classification for BCIs, Expert Syst. Appl. 114 (2018), 532–542.

[16] T.T. Erguzel, S. Ozekes, S. Gultekin, N. Tarhan, Ant Colony Optimization Based Feature Selection Method for QEEG Data Classification, Psych. Invest. 11 (2014), 243.

[17] M. Fryz, Conditional linear random process and random coefficient autoregressive model for EEG analysis, in: 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON), IEEE, Kiev, 2017: pp. 305–309.

[18] R. Fu, Y. Tian, T. Bao, Z. Meng, P. Shi, Improvement Motor Imagery EEG Classification Based on Regularized Linear Discriminant Analysis, J. Med. Syst. 43 (2019), 169.

[19] F.A. Gers, J. Schmidhuber, F. Cummins, Learning to Forget: Continual Prediction with LSTM, Neural Comput. 12 (2000), 2451–2471.

[20] T. Fawcett, An introduction to ROC analysis, Pattern Recogn. Lett. 27 (2006), 861–874.

[21] I. Goodfellow, Y. Bengio, A. Courville, Deep learning, The MIT Press, Cambridge, MA, 2017.

[22] K.-S. Hong, M.J. Khan, Hybrid Brain–Computer Interface Techniques for Improved Classification Accuracy and Increased Number of Commands: A Review, Front. Neurorobot. 11 (2017), 35.

[23] J. Hua, Z. Xiong, J. Lowey, E. Suh, E.R. Dougherty, Optimal number of features as a function of sample size for various classification rules, Bioinformatics. 21 (2005), 1509–1515.

[24] Y. Jeon, C.S. Nam, Y.-J. Kim, M.C. Whang, Event-related (De)synchronization (ERD/ERS) during motor imagery tasks: Implications for brain–computer interfaces, Int. J. Ind. Ergon. 41 (2011), 428–436.

[25] H. Ji, J. Li, R. Lu, R. Gu, L. Cao, X. Gong, EEG Classification for Hybrid Brain-Computer Interface Using a Tensor Based Multiclass Multimodal Analysis Scheme, Comput. Intell. Neurosci. 2016 (2016), 1732836.

[26] S. Kanoga, Y. Mitsukura, Review of Artifact Rejection Methods for Electroencephalographic Systems, in: W. Sittiprapaporn (Ed.), Electroencephalography, InTech, 2017.

[27] R.J. Korhonen, J.C. Hernandez-Pavon, J. Metsomaa, H. Mäki, R.J. Ilmoniemi, J. Sarvas, Removal of large muscle artifacts from transcranial magnetic stimulation-evoked EEG by independent component analysis, Med. Biol. Eng. Comput. 49 (2011), 397–407.

[28] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, Commun. ACM. 60 (2017), 84–90.

[29] A. Kumar, M. Singh, Optimal Selection of Wavelet Function and Decomposition Level for Removal of ECG Signal Artifacts, J. Med. Imaging Health Inform. 5 (2015), 138–146.

[30] Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE. 86 (1998), 2278–2324.

[31] D. Lee, H.-J. Lee, S.-G. Lee, Motor Imagery EEG Classification Method Using EMD and FFT, J. KIISE, 41 (2014), 1050–1057.

[32] D. C. Leonard, Learning theories, A to Z, Oryx Press, Westport, CT, 2002,

[33] J. Li, Parallel two-class 3D-CNN classifiers for video classification, in: 2017 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), IEEE, Xiamen, China, 2017: pp. 7–11.

[34] Y. Li, G. Zhou, D. Graham, A. Holtzhauer, Towards an EEG-based brain-computer interface for online robot control, Multimed. Tools Appl. 75 (2016), 7999–8017.

[35] M. Lokman, A. Dabag, N. Ozkurt, S. Miqdad, M. Najeeb, Feature Selection and Classification of EEG Finger Movement Based on Genetic Algorithm, in: 2018 Innovations in Intelligent Systems and Applications Conference (ASYU), IEEE, Adana, 2018: pp. 1–5.

[36] Y. Lu, H. Jiang, W. Liu, Classification of EEG Signal by STFT-CNN Framework: Identification of Right-/left-hand Motor Imagination in BCI Systems, in: Proceedings of The 7th International Conference on Computer Engineering and Networks — PoS(CENet2017), Sissa Medialab, Shanghai, China, 2017: p. 001.

[37] Z. Mahmood, S. Khan, On the Use of K-Fold Cross-Validation to Choose Cutoff Values and Assess the Performance of Predictive Models in Stepwise Regression, Int. J. Biostat. 5 (2009), 25.

[38] J.N. Mak, J.R. Wolpaw, Clinical Applications of Brain-Computer Interfaces: Current State and Future Prospects, IEEE Rev. Biomed. Eng. 2 (2009), 187–199.

[39] A.H. Marblestone, G. Wayne, K.P. Kording, Toward an Integration of Deep Learning and Neuroscience, Front. Comput. Neurosci. 10 (2016), 94.

[40] K.L. Masita, A.N. Hasan, S. Paul, Pedestrian Detection Using R-CNN Object Detector, in: 2018 IEEE Latin American Conference on Computational Intelligence (LA-CCI), IEEE, Gudalajara, Mexico, 2018: pp. 1–6.

[41] J. Metsomaa, J. Sarvas, R.J. Ilmoniemi, Multi-trial evoked EEG and independent component analysis, J. Neurosci. Meth. 228 (2014), 15–26.

[42] S.D. Muthukumaraswamy, High-frequency brain activity and muscle artifacts in MEG/EEG: a review and recommendations, Front. Hum. Neurosci. 7 (2013), 138.

[43] F. Nasiriyan, H. Khotanlou, Sparse Connectivity and Activity Using Sequential Feature Selection in Supervised Learning, Appl. Artif. Intell. 32 (2018), 568–581.

[44] L.F. Nicolas-Alonso, J. Gomez-Gil, Brain Computer Interfaces, a Review, Sensors. 12 (2012), 1211–1279..

[45] H. Nolan, R. Whelan, R.B. Reilly, FASTER: Fully Automated Statistical Thresholding for EEG artifact Rejection, J. Neurosci. Meth. 192 (2010), 152–162.

[46] N. Padfield, J. Zabalza, H. Zhao, V. Masero, J. Ren, EEG-Based Brain-Computer Interfaces Using Motor-Imagery: Techniques and Challenges, Sensors. 19 (2019), 1423.

[47] M.P. Paulraj, C.R. Hema, R. Nagarajan, S. Yaacob, A.H. Adom, EEG Classification using Radial Basis PSO Neural Network for Brain Machine Interfaces, in: 2007 5th Student Conference on Research and Development, IEEE, Selangor, Malaysia, 2007: pp. 1–5.

[48] F. Pichiorri, G. Morone, M. Petti, J. Toppi, I. Pisotta, M. Molinari, S. Paolucci, M. Inghilleri, L. Astolfi, F. Cincotti, D. Mattia, Brain-computer interface boosts motor imagery practice during stroke recovery: BCI and Motor Imagery, Ann. Neurol. 77 (2015), 851–865.

[49] D. Rathee, H. Raza, G. Prasad, H. Cecotti, Current Source Density Estimation Enhances the Performance of Motor-Imagery-Related Brain–Computer Interface, IEEE Trans. Neural Syst. Rehabil. Eng. 25 (2017), 2461–2471.

[50] K.S.-T. Salo, T.P. Mutanen, S.M.I. Vaalto, R.J. Ilmoniemi, EEG Artifact Removal in TMS Studies of Cortical Speech Areas, Brain Topogr. 33 (2020), 1–9.

[51] R.T. Schirrmeister, J.T. Springenberg, L.D.J. Fiederer, M. Glasstetter, K. Eggensperger, M. Tangermann, F. Hutter, W. Burgard, T. Ball, Deep learning with convolutional neural networks for EEG decoding and visualization: Convolutional Neural Networks in EEG Analysis, Hum. Brain Mapp. 38 (2017), 5391–5420.

[52] A. Shahrjooihaghighi, H. Frigui, X. Zhang, X. Wei, B. Shi, A. Trabelsi, An ensemble feature selection method for biomarker discovery, in: 2017 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), IEEE, Bilbao, 2017: pp. 416–421.

[53] D. Shen, G. Wu, H.-I. Suk, Deep Learning in Medical Image Analysis, Annu. Rev. Biomed. Eng. 19 (2017), 221–248.

[54] Siuly, H. Wang, Y. Zhang, Detection of motor imagery EEG signals employing Naïve Bayes based learning process, Measurement. 86 (2016), 148–158.

[55] M. Tangermann, K.-R. Müller, A. Aertsen, N. Birbaumer, C. Braun, C. Brunner, R. Leeb, C. Mehring, K.J. Miller, G.R. Müller-Putz, G. Nolte, G. Pfurtscheller, H. Preissl, G. Schalk, A. Schlögl, C. Vidaurre, S. Waldert, B. Blankertz, Review of the BCI Competition IV, Front. Neurosci. 6 (2012), 55.

[56] W.O. Tatum, B.A. Dworetzky, D.L. Schomer, Artifact and Recording Concepts in EEG, J. Clinic. Neurophysiol. 28 (2011), 252–263.

[57] M.A. Uusitalo, R.J. Ilmoniemi, Signal-space projection method for separating MEG or EEG into components, Med. Biol. Eng. Comput. 35 (1997), 135–140.

[58] C. Uyulan, T.T. Erguzel, Analysis of Time - Frequency EEG Feature Extraction Methods for Mental Task Classification, Int. J. Comput. Intell. Syst. 10 (2017), 1280-1288.

[59] C. Uyulan, T.T. Ergüzel, N. Tarhan, Entropy-based feature extraction technique in conjunction with wavelet packet transform for multi-mental task classification, Biomed. Eng. (Biomed. Techn.) 64 (2019), 529–542.

[60] M. Volker, J. Hammer, R.T. Schirrmeister, J. Behncke, L.D.J. Fiederer, A. Schulze-Bonhage, P. Marusic, W. Burgard, T. Ball, Intracranial Error Detection via Deep Learning, in: 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), IEEE, Miyazaki, Japan, 2018: pp. 568–575.

[61] Y. Wang, T.-P. Jung, Improving Brain–Computer Interfaces Using Independent Component Analysis, in: B.Z. Allison, S. Dunne, R. Leeb, J. Del R. Millán, A. Nijholt (Eds.), Towards Practical Brain-Computer Interfaces, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012: pp. 67–83.

[62] Z. Wei, C. Wu, X. Wang, A. Supratak, P. Wang, Y. Guo, Using Support Vector Machine on EEG for Advertisement Impact Assessment, Front. Neurosci. 12 (2018), 76.

[63] I. Winkler, S. Haufe, M. Tangermann, Automatic Classification of Artifactual ICA-Components for Artifact Removal in EEG Signals, Behav. Brain Funct. 7 (2011), 30.

[64] J. Yang, S. Yao, J. Wang, Deep Fusion Feature Learning Network for MI-EEG Classification, IEEE Access. 6 (2018), 79050–79059.

[65] T. Yu, J. Xiao, F. Wang, R. Zhang, Z. Gu, A. Cichocki, Y. Li, Enhanced Motor Imagery Training Using a Hybrid BCI With Feedback, IEEE Trans. Biomed. Eng. 62 (2015), 1706–1717.

[66] X. Zhao, H. Zhang, G. Zhu, F. You, S. Kuang, L. Sun, A Multi-Branch 3D Convolutional Neural Network for EEG-Based Motor Imagery Classification, IEEE Trans. Neural Syst. Rehabil. Eng. 27 (2019), 2164–2177.