BIOCYBERNETICS AND BIOMEDICAL ENGINEERING 41 (2021) 1629–1645

Available at www.sciencedirect.com

ScienceDirect

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

Original Research Article

Compact convolutional neural network (CNN) based on SincNet for end-to-end motor imagery decoding and analysis

Tarmizi Ahmad Izzuddin [a,b,*], *Norlaili Mat Safri* [b], *Mohd Afzan Othman* [b]

[a] *Department of Control, Instrumentation and Automation, Faculty of Electrical Engineering, Universiti Teknikal Malaysia Melaka, Melaka, Malaysia*

[b] *Department of Electronic and Computer Engineering, School of Electrical Engineering, Faculty of Engineering, Universiti Teknologi Malaysia, Johor, Malaysia*

ARTICLE INFO

Article history:
Received 9 July 2021
Received in revised form
13 September 2021
Accepted 1 October 2021
Available online 13 October 2021

Keywords:
Brain-Computer Interface (BCI)
Convolutional Neural Network (CNN)
Electroencephalogram (EEG)
Motor imagery

ABSTRACT

In the field of human-computer interaction, the detection, extraction and classification of the electroencephalogram (EEG) spectral and spatial features are crucial towards developing a practical and robust non-invasive EEG-based brain-computer interface. Recently, due to the popularity of end-to-end deep learning, the applicability of algorithms such as convolutional neural networks (CNN) has been explored to achieve the mentioned tasks. This paper presents an improved and compact CNN algorithm for motor imagery decoding based on the adaptation of SincNet, which was initially developed for speaker recognition task from the raw audio input. Such adaptation allows for a compact end-to-end neural network with state-of-the-art (SOTA) performances and enables network interpretability for neurophysiological validation in cortical rhythms and spatial analysis. In order to validate the performance of proposed algorithms, two datasets were used; the first is the publicly available BCI Competition IV dataset 2a, which was often used as a benchmark in validating motor imagery classification algorithms, and the second is a dataset consists of primary data initially collected to study the difference between motor imagery and mental-task associated motor imagery BCI and was used to test the plausibility of the proposed algorithm in highlighting the differences in terms of cortical rhythms. Competitive decoding performance was achieved in both datasets in comparisons with SOTA CNN models, albeit with the lowest number of trainable parameters. In addition, it was shown that the proposed architecture performs a cleaner band-pass, highlighting the necessary frequency bands that were crucial and neurophysiologically plausible in solving the classification tasks.

* Corresponding author at: Department of Control, Instrumentation and Automation, Faculty of Electrical Engineering, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia.
E-mail address: tarmizi@utem.edu.my (T.A. Izzuddin).

https://doi.org/10.1016/j.bbe.2021.10.001

1. Introduction

A Brain-Computer Interface (BCI) can be defined as a system that translates brain activity patterns into messages or commands that represent the user's intention or condition by using a direct brain-to-computer mode of communication [1–3]. There are multiple methods of measuring brain activity ranging from the non-invasive method of measuring Electroencephalography (EEG) signals to a more invasive method called Electrocorticography (ECoG) that places a thin layer of electrodes directly on top of the exposed layer of the brain [4]. In both methods, one particular means of enabling BCI is through the detection of the brain's Motor-imagery (MI) signals, signals that arise due to the synchronization/desynchronization of specific frequency bands during the imagined movement of different body parts [5]. Such BCI systems are often called MI-based BCI or MI-BCI for short.

1.1. *Related works*

Over the past years, numerous decoding techniques and approaches have been developed to achieve good decoding performance from EEG signals. Generally, decoding MI signals follow a particular pipeline, consisting of filtering the raw EEG signals, extracting features from the filtered signals, and classifying them using suitable classifiers. Thus, the MI decoding can be categorized depending on the types of extracted features used and the classification's approach.

One competition-winning approach of decoding MI signal is by obtaining the Common Spatial Pattern (CSP) features to be classified with a classifier such as linear discriminant analysis (LDA), support vector machine (SVM) or similar linear classifier. In this form, a combination of band-pass with spatial filtering is used to emphasizes the MI differences of different body parts. Spatial filtering is commonly achieved using CSP algorithms to maximize an MI class variance while minimising the other [6–8]. One popular and competition-winning variant of CSP is the Filter Bank Common Spatial Pattern (FBCSP) algorithm in which CSP is used in conjunction with a bank of band-pass filters [9–10] and L1-Norm based features selection method [11] in order to obtain the optimal sets of spatial filters in multiple frequency bands. A more recent method uses the Spiking Neural Network (SNN) [12] to classify features extracted from FBCSP, enhancing multiple classes MI classification.

Apart from CSP, a more state-of-the-art approach is through the use of Riemannian Geometry based classifier. The basic concept of a Riemannian Geometry Classifier (RGC) is to map the data directly to a geometric space fitted with an appropriate metric. In this geometric space, manipulating interpolation and classification data can be performed much easily compared to the non-geometric space. The earliest study that proposed the use of this approach is detailed in [13] in which a Riemmanian Distance To Mean (RDTM) classifier was used to classify band-pass covariance features for MI decoding. Apart from RDTM, a more accurate Riemannian geometric approach is highlighted in [14] in which an SVM Riemannian kernel is proposed for classification.

However, decoding the MI signal using previously mentioned techniques requires separately tuning and optimizing the decoding pipeline's modulation, filtering, and classification stage. This process often requires a priori or subject-matter expert about the expected signal outcome. To alleviate this matter, many research studies have explored the use of *Deep Learning* (also known as deep neural networks) towards end-to-end decoding of MI signals. The use of deep Convolutional Neural Networks (CNN), for example, has become highly successful in many applications such as in computer vision and speech recognition, often outperforming conventional engineered methods. Using CNN with many layers, researchers managed to reduce the error rates on the ImageNet image recognition challenge, where 1.2 million images must be classified into 1000 different classes, from 26% to just below 4% in 4 years [15]. Deep CNN also contributes to the success of reducing the error rate on speech recognition and enables the development of current mobile speech recognition technology [16].

Owing to the great success in these fields, the applicability of using deep CNN for EEG-based MI signal classification has been explored by researchers [17–20]. CNN enables end-to-end learning, which is learning directly from raw EEG signals without a priori collection of features, scales to large datasets, and takes the hierarchy of natural signals (learning from simple concepts in its early layer into complex ones in its last layer), unlike the traditional processing and classification system.

One such example is a work reported in [17], in which the authors exhaustively explored the feasibility of using both deep and shallow CNN for EEG motor imagery signal decoding and reported excellent performance even when compared with the conventional FBCSP method. In [21], the authors introduced a compact CNN for BCI purposes called EEGNet, which can be applied to multiple BCI paradigms. Unlike standard methods, which are often tailored to specific BCI paradigms, EEGNet can be applied towards multiple classification tasks (here the author listed four: P300 classification, ERD/ERS, movement-related cortical potential - MRCP, and sensory-motor rhythms -SMR) without changing the network architecture.

A more MI-specific CNN architecture based on EEGNeT was later proposed by authors in [22], which uses the temporal convolutional layer from EEGNeT and spatial feature extraction convolutional layer and activation function from CSP-

Table 1 – Total number of CNN parameters architectures used in this study.

	Shallow ConvNet	EEGNet	TA-CSPNN	Sinc-EEGNet	Sinc-CSPNN
No. of parameters	40,566	1876	978	1380	644

NN. The proposed architecture (called Temporally Adaptive Common Spatial Pattern, TA-CSPNN) uses half the parameters from EEGNet but retains similar accuracy. Another study that explores the use of CNN for MI-based EEG classification is given in [23], in which the 5-layer CNN model is built to classify MI tasks (left- and right-hand movement). Here it was reported that using CNN improved classification accuracy over conventional methods such as SVM and CSP.

1.2. *Proposed method*

This paper introduces a much compact and improved EEG-based MI decoding architecture based on EEGNeT and TA-CSPNN but proposes using a parametrized Sinc-based convolution network using SincNet [24] on the first CNN layer. Initially proposed for the speaker recognition task from raw audio waveform, SincNet allows the first convolutional layer to act as a differentiable band-pass filter and co-joint with standard NN architecture. This is achieved via the convolution of a parameterized Sinc function with the input signals.

Although it was shown that a standard CNN could be trained to act as a Finite-Impulse-Response (FIR) filter on EEG signals [17,21–22,25], the application of SincNet on the first convolutional layer allows the implementation of band-pass filters with a fewer number of high-level tuneable parameters. SincNet thus emphasizes the network on generating band-pass filters that have a better impact on the shape and bandwidth and allowing for better frequency band interpretability. Such architecture was, in part, motivated by the original FBCSP algorithm in which a bank of band-pass filters was used in the early stage of the decoding pipeline.

However, unlike FBCSP, the proposed architecture allows auto-optimization of these band-pass filters during network training since the SincNet layer parameters are differentiable and can be jointed with other CNN layers. This can also be viewed as a form of Differentiable Digital Signal Processing (DDSP) in which the deep learning method is integrated with classical signal processing elements [26]. Such an approach benefits from the inductive bias of using proven signal processing methods while retaining NN's expressive power and end-to-end learning.

2. Methodology

2.1. *SincNet adaptation*

In a standard CNN architecture, time-domain convolution is performed between the input waveform and a set of learned kernels. In the case of using 1-D time-series data such as EEG signal as an input, this kernel act as a Finite Impulse Response (FIR) filter. The original convolution process is defined as follows:

$$y[n] = (x * h)[n] = \sum_{l=0}^{L-1} x[l] \cdot h[n-l] \tag{1}$$

here $y[n]$ is the filtered output, $x[n]$ is the input signal and $h[n]$ is the kernel of length L. Since an EEG data with C number of channels with trial length T can be viewed as a 2D input, we proposed the following 2D convolution by modifying Eq. (1):

$$y_{c,j}[n] = (x_c * h_j)[n] = \sum_{l=0}^{L-1} x_c[l] \cdot h_j[n-l] \tag{2}$$

Where $c \in C$ and $j \in F_K$. Here hyperparameter F_K denotes the total number of band-pass filters to be used with CNN. Generally, decoding EEG signals using CNN architecture end-to-end enforces optimization of first layer kernel (which generally act as a band-pass filter) during CNN training [17,21,23], with all L elements of the filter are learned during this process. On the other hand, SincNet [24] proposed the use of a predefined Sinc function for a kernel that depends on few learnable parameters. This is motivated by standard filtering in digital signal processing, in which the idea is to design a function that acts as a rectangular band-pass filter. Since in the frequency domain, a band-pass filter can be viewed as a difference between two low-pass filters, the proposed kernel of equation (2) can therefore be viewed in frequency domain as:

$$H_j[f, f_1, f_2] = rect\left(\frac{f}{2f_2}\right) - rect\left(\frac{f}{2f_1}\right) \tag{3}$$

where f_1 and f_2 are the low and high cut-off frequencies respectively, and $rect(.)$ is the rectangular function. The Inverse Fourier Transform (IFT) of this equation then becomes:

$$\begin{aligned} h_j[n, f_1, f_2] &= 2f_2 \frac{\sin(2\pi f_2 n)}{2\pi f_2 n} - 2f_1 \frac{\sin(2\pi f_1 n)}{2\pi f_1 n} \\ &= 2f_2 \mathrm{sinc}(2\pi f_2 n) - 2f_1 \mathrm{sinc}(2\pi f_1 n) \end{aligned} \tag{4}$$

Hence, instead of learning a L sized kernel during CNN training of, the proposed method allows learning of only two parameters, f_1 and f_2, which defines the kernel structure for a convolution process in the time domain. Since equation (4) shows that the band-pass filter can be constructed using a set of differentiable Sinc functions, then parameters f_1 and f_2 can be jointly optimized with other CNN parameters using Stochastic Gradient Descent (SGD) or any gradient-based optimization method. Formally, given $\mathscr{L}$ as the loss-function of the CNN, then using chain-rule, gradient calculation to parameters f_1 and f_2 is possible:

$$\frac{\partial \mathscr{L}}{\partial [f_1, f_2]} = \frac{\partial \mathscr{L}}{\partial h} \cdot \frac{\partial h}{\partial [f_1, f_2]} \tag{5}$$

where $\frac{\partial \mathscr{L}}{\partial [f_1 f_2]}$ is the partial derivative of $\mathscr{L}$ with respect to f_1 and f_2. Moreover, an update to the parameters is obtained by:

$$[f_1, f_2]' = [f_1, f_2] + \eta \frac{\partial \mathscr{L}}{\partial [f_1, f_2]} \tag{6}$$

here η is the learning rate used during CNN training. To ensure that update towards f_1 and f_2 follow $f_1 \geq 0$ and $f_2 \geq f_1$ during training, as suggested by the original SincNet authors, these parameters are fed by the following parameters:

$$f_1^{abs} = |f_1| \tag{7}$$

$$f_2^{abs} = f_1 + f_{band} \tag{8}$$

where $f_{band} = |f_2 - f_1|$ denotes the filter's band size. Consequently, only f_1^{abs} and f_{band} are the only parameters updated

during network training. Again, per the original SincNet authors' suggestion, this convolutional filter is windowed using the popular Hamming windows to mitigate the issue of filter truncation. Given $w[n]$ as the window function, proposed windowed filter $h_{j-windowed}[n,f_1,f_2]$ then becomes:

$$h_{j-windowed}[n,f_1,f_2] = h_j[n,f_1,f_2] \cdot w[n] \quad (9)$$

Where:

$$w[n] = 0.54 - 0.46\cos\left(\frac{2\pi n}{L}\right) \quad (8)$$

Reasoning from this fact, the construction of a filter-bank is possible using proposed Sinc-based band-pass filters. Although the use of filter-bank is inspired by a previous competitive algorithm such as the FBCSP, the proposed filter-bank can be viewed as a form of adaptive filter-bank that automatically adapts the low cut-off frequency and filter's bandwidth in order to minimize classification error, hence, improving model accuracy. Therefore, from the perspective of using CNN for EEG decoding, replacing the first layer with a Sinc-based convolutional filter allows adaptive filtering of particular EEG wave pertaining to the decoding task with fewer parameters.

Apart from this, since the proposed filter is fully differentiable, all standard CNN pipelines (such as pooling, dropout, and normalisation) can be employed together with the proposed filter. Standard convolutional or fully connected CNN layers can be used stacked together to perform EEG classification. In this study, two compact CNN motor image decoding architectures, the EEGNet (code adapted from https://github.com/vlawhern/arl-eegmodels) and the TA-CSPNN (code adapted https://github.com/mahtamsv/TA-CSPNN), were adapted to the proposed SincNet layer, i.e., the original first layer was removed and replaced with a SincNet layer (Hence, we named it as Sinc-EEGNet and Sinc-CSPNN, respectively) such as shown in Fig. 1.

In both architectures, a depthwise convolutional layer was employed after the SincNet layer to generate the frequency-specific spatial-filters F_s, for the network [17,21,22]. The weights and neurons in this layer are arranged in such a way as to mimics the spatial filters in motor imagery decoding using the CSP algorithm [6,9], which helps in discriminating between EEG signals belonging to a particular task. It was then constrained using the norm $\|w\|_2 \leq 1$ since spatial filters in the CSP algorithm are eigenvectors with a norm of 1.

Similar to the original EEGNet, in the proposed Sinc-EEGNet, a separable convolution layer (which consists of a depthwise convolution followed by pointwise convolutions) was employed to decouple the relationship within and across feature maps by summarising each feature map then merging it at the output. However, such a convolution layer was not employed in TA-CSPNN and thus was not used in our Sinc-CSPNN.

Another difference between both architectures is that the CSPNN uses a squared activation function instead of an exponential linear unit (ELU) since it was claimed that ERS/ERD features are variations in the power of EEG signal. However, this study found that the squared activation function's use

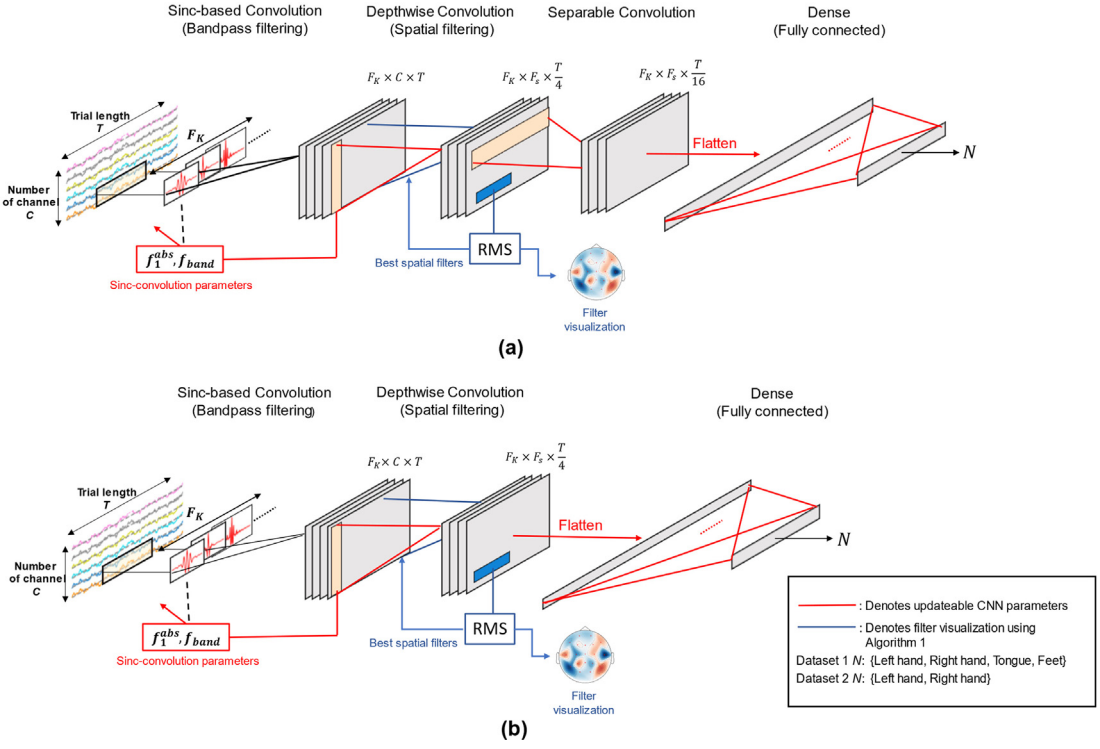


Fig. 1 – Proposed SincNet adapted CNN architecture: (a) Sinc-EEGNet; (b) Sinc-CSPNN.

causes the network not to learn during training. A simple experiment was conducted to replace the activation function by comparing the performance of the widely used ELU and rectified linear unit (RELU) for replacement, and both activation functions were found to alleviate the problem mentioned above. We then chose RELU for the Sinc-CSPNN's activation function.

The final layer of both Sinc-EEGNet and Sinc-CSPNN consists of dense NN connected to a softmax classifier with an output N which aggregates the features generated from the previous layer. Softmax is generally chosen to reduce the free parameters, such as shown in [27]. A detailed structure of the proposed architecture, along with its hyperparameters, are listed in Table A1 in the appendix.

2.2. *Feature visualization and interpretation*

Since CNN-based EEG decoding performs end-to-end learning on raw EEG data, visualization of the learned CNN filters is necessary to validate and understand how proposed Sinc-based architectures distinguish between tasks. The development of CNN visualization has been an important part of the research at this time. As an essential component of the validation process, it was proposed to ensure that the relevant features drive classification performance rather than random noise or artifacts in EEG data [17,21].

Thus, due to the structure of the proposed Sinc-based CNN, two forms of interpretation and visualization of the network are proposed. The first is made possible by the network's spatial filtering (depthwise convolution) layer, which allows for visualization of the topographic spatial patterns by extracting the trained weights in this layer. The second is the visualization of the frequency band that the network focuses on through the Sinc-convolutions layer. The subsequent section explains each form of proposed visualization and interpretation techniques.

2.2.1. *Task-related spatial pattern extraction*

In the original CSP algorithm, spatial patterns relating to one task can be visualized by obtaining the columns of the inverse projection matrix $\mathbf{W}^{-1}$, where the first and last columns are the most important spatial patterns which explain the greatest variance in one task and the smallest variance in the other [6]. In this study, through depth-wise convolution, such patterns can be obtained by visualizing the trained network's spatial filters.

Here, we introduce an algorithm to obtained task-related spatial patterns by identifying the spatial feature map with the highest activation value and backtracking the corresponding filters connected to this feature map. The highest activation was obtained by calculating the root-mean-square (RMS) on the i^{th} feature map (the i^{th} output of the spatial filters) while inputting the network with EEG trials data pertaining to task to be analyzed using proposed algorithm 1 listed below. Again, here F_K refers to the total number of band-pass while F_s refers to frequency-specific spatial filter. Due to the use of the Sinc convolutional layer and the structure of the proposed network, backtracking and extracting a set of spatial filter weights W_c, which gave the maximum activation of the task-related feature map is possible. The corresponding spatial patterns during task n can be obtained by mapping such weights onto an EEG topographic head plot for cortical activity analysis.

Algorithm 1: Spatial pattern extration

Input	:	i^{th} output of the spatial filters x_i pertaining to input data during task n, where $n \in N$
Output	:	A set of spatial filter weights W_c where $c \in C$
Step 1	:	**For** i in the total number of F_K : **For** j in the total number of F_S :
Step 2	:	Calculate the RMS for x_i feature map: $RMS_{i,j} = \sqrt{\frac{1}{T}\sum (x_{i,j})^2}$ Where T is the trial's length.
Step 3	:	**If** $RMS_{i,j} > RMS_{i-1,j-1}$: Save the index $[i_{max}, j_{max}] \leftarrow [i, j]$
Step 4	:	**For** c in the total number of EEG channels C:
	:	**If** W_c is connected to $[i_{max}, j_{max}]$:
		Return $W_{c-max\_i} \leftarrow W_c$

2.2.2. *Frequency band visualization over sinc-kernels*

Since the first layer of the network performs sinc-based convolution, which acts as a band-pass filter, visualization of its frequency response gives an intuition on the frequency band that the network focuses on to solve the classification task. Such information is crucial to establishing the validity of the proposed CNN architecture with known cortical rhythms such as the sensorimotor rhythm (SMR), a common control signal for oscillatory-based BCI [28]. Hence, for this purpose, we proposed two ways of visualizing the frequency response:

1) Visualization of individual kernel's frequency response, which gave an intuition on the frequency band that the i-th sinc-filter (which in turns connected to [i-th, j-th] spatial filter) focuses on. This was performed by performing discrete Fourier transform (DFT) over the individual Sinc kernel $h[n]$ using the standard DFT equation:

$$y[k] = \sum_{n=0}^{N-1} e^{-2\pi j \frac{kn}{N}} \cdot h[n] \tag{9}$$

2) The cumulative frequency response of all learned kernels provided insights on the frequency that the whole network focuses on. This can be obtained by performing by summing the frequency response of all kernels using the equation below:

$$Filter\ Sum = \frac{1}{F_K} \sum_{i=1}^{F_k} \sum_{k=1}^{T'} y_i[k]) \tag{10}$$

here T' denotes the window size, and F_k is the number of filters being used.

2.3. *Datasets*

We perform the validation of the proposed method using two datasets. The first Dataset is the publicly available BCI Com-

petition VI (2a) [29] dataset, often used as a benchmark to gauge classifier performance in decoding EEG-based motor imagery signals. Another dataset consists of our primary data that we initially used to study the performance of motor imagery associated with mental imagery tasks.

2.3.1. *Dataset I*

Initially used for BCI competition, this publicly available Dataset (http://www.bbci.de/competition/iv/#datasets) has been used in numerous studies concerning motor imagery BCI [17,21–22]. Dataset consists of EEG recordings of 9 participants performing left hand, right hand, tongue, and both feet motor imagery in two sessions; training and evaluation. Each session recording consists of 228 trials. Data was recorded using 22 Ag/AgCl electrodes (see Fig. 2 for electrode position) with a sampling frequency of 250 Hz. However, in this study, all data were resampled to 125 Hz following the pre-processing procedure that was described in the original data description. A band-pass filter in the range 4–40 Hz was employed, and data was epoch from 0.5 to 2.5 s (in the MI region) following Dataset's description. All data recorded from training sessions was used to train all classifiers, while data from evaluation sessions were used to evaluate and test (50% each from the total number of data) (Fig. 3).

2.3.2. *Dataset II*

This Dataset consists of EEG recordings from 11 participants (10 males, one female) who all voluntarily agreed to participate and signed a given consent form approved by the Ministry of Health Malaysia. Data were recorded using the medical-grade NVX-52 EEG amplifier from MKS utilizing 19 channels with electrodes (AgCl) placed according to the international 10–20 system. Recorded initially to study differences between the Motor Imagery (MI) and Motor Imagery associated with the mental rotation task (we denote this as MI + MR from now on), this Dataset consists of 8 trials per subject (4 trials MI, 4 trials MI + MR) with each trial lasted 7 s. The use of this Dataset is partly motivated in assessing proposed sinc based CNN as a tool for interpreting and visualizing differences between MI and MI + MR task in terms of features learned by CNN.

During MI tasks, participants were required to perform right and left-hand imagery movement. A short training session was then conducted in which participants were required to perform a virtual 3D object manipulation task by associating hand movements with the rotation of a 3D star-shaped object on a computer screen through the use of an accelerometer attached to the participant's hand. After training, participants were again required to perform a motor imagery task, but this time around while simultaneously mind visualizing the 3D object rotation seen and manipulated during the training session (hence such task was denoted as MI + MR). All EEG recordings were sampled at 500 Hz downsampled to 125 Hz and notch filtered at 60 Hz.

Since the number of trials per subject is fewer, albeit with a longer trial duration when compared to Dataset I, a sliding window strategy similar to [17] was employed to augment the number of trials and provide more training data for the network, in this strategy, "multiple crops" of EEG data are used to increase the EEG decoding accuracy. Formally, given original trial $X^j \in \mathbb{R}^T$ with T as timesteps, a set of crops with window size T' as time slices with hop size h of the trial is given as follows:

$$C^j = \left\{ X^j_{t+h\cdots t+h+T'} | t \in 1..T - T' \right\} \tag{11}$$

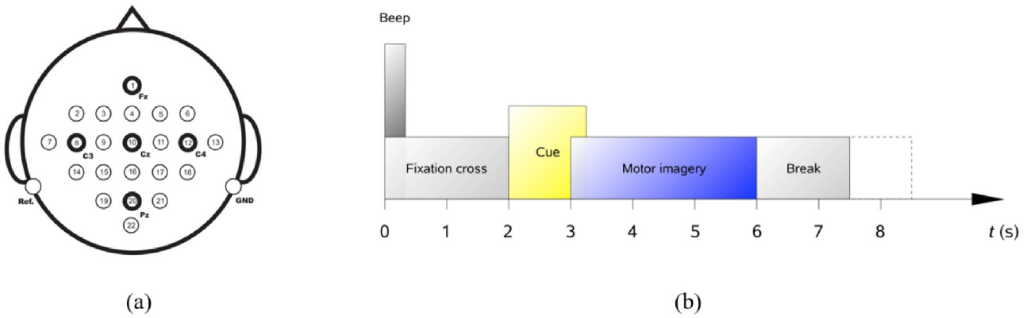
All of the T- T' crops are used as the new training data for our CNN classifier. A window size of 250 (around 1 s) with a hop size of 3 was chosen, yielding a total of 393 crops per trial. This hop size was chosen in such a way to ensure that total crops n fit within the trial length T (hence n must be a natural number) according to the following equation:

$$T' + nh = Tn \in \mathbb{N} \tag{12}$$

3. Results

3.1. *Parameters and architecture setup*

In order to gauge the performance of proposed NN architectures, proposed architectures were compared with their non-sinc counterpart (EEGNeT and TA-CSPNN) as well as the state-of-the-art (SOTA) ShallowConvNet from [17]. In addition, since Dataset I was used in the BCI competition IV 2a, a comparison with the competition-winning FBCSP approach was also made in this study. Same parameters were used for both dataset 1 and 2. Except for the first convolution layer in EEGNeT and TA-CSPNN, all parameters in the layers that follow remain the same with our Sinc-based architectures. Since both recordings were downsampled to 125 Hz, a temporal kernel sized $K = 63$ was chosen. This was chosen to allow the proposed NN to collect information in the 2 Hz and above



(a) (b)

Fig. 2 – (a) Electrode montage used in the dataset I; (b) its timing scheme [27].

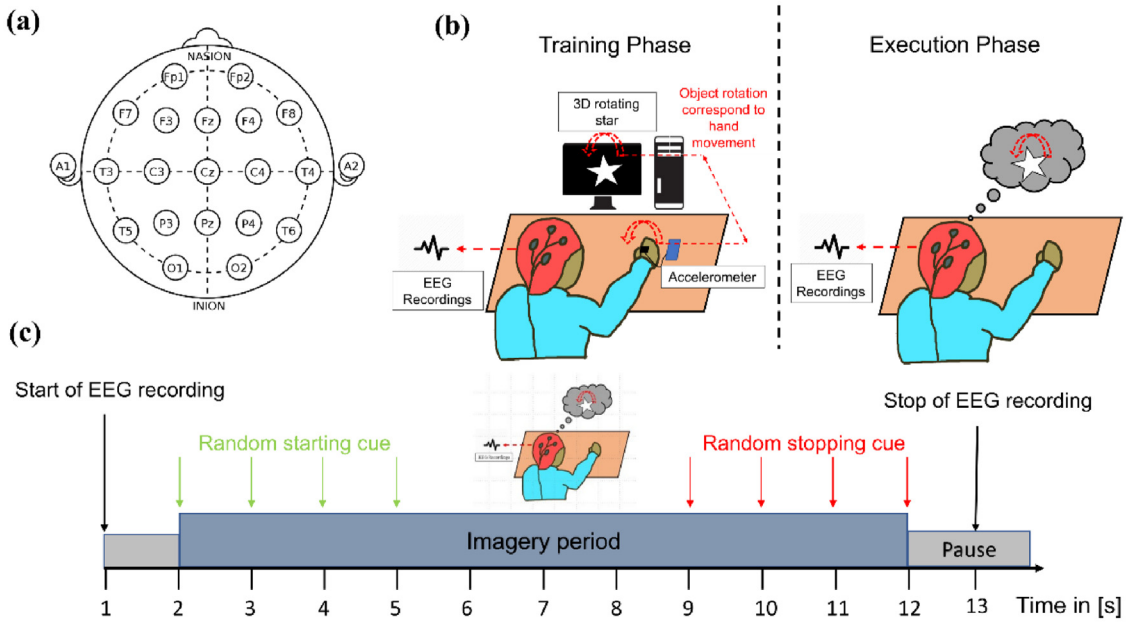


Fig. 3 – Depiction of BCI training protocol and execution phase of Dataset II. (a).

frequency region, as this number is about half the downsampled frequency.

The number of Sinc-based convolution filters F_k were set to 10 with each filter has a band-pass width f_{band} to be at 4 Hz. All band-pass filters are initialised in such a way that they are equally distributed in the range 4 Hz and a maximum of 40 Hz (i.e. starting with [4,8,9,12]… [36,40]. Such is partly inspired by the construction of filter banks constructed in the original FBCSP algorithm. However, it should be noted that since parameters f_{band} and f_1 are differentiable, this act as an adaptive band-pass filter during network training, unlike the static filters in FBCSP. The number of spatial filters F_s was set 2 following the original parameter in EEGNet and TA-CSPNN, while the number of pointwise convolution filters F_p in Sinc-EEGNet is set to 16 (Table A1 in the appendix section shows the overall architecture of the proposed. Sinc-EEGNeT and Sinc-CSPNN).

In both Sinc-convolution and depthwise layers, we employed dropout [30] with a dropout rate of $p = 0.5$, to minimise the effects of overfitting. Table 2 summarizes the total number of trainable parameters in comparisons with its non-sinc counterpart and SOTA shallowConvNet. Training of all CNN models were performed on a standard PC with Intel i5 CPU and 8 GB of RAM, equipped with an Nvidia GTX 1050 GPU with 4 GB of NVRAM. The use of a GPU was necessary in order to increase the training speed. During training, a 10-fold cross-validation method was performed to ensure stability and ensure the model's effectiveness.

3.2. *Classification performance on Dataset I*

All models were trained on the train set for Dataset I with 10% of data used as validation. The models were then tested on the evaluation dataset according to the original BCI Competition rules. Ten different weight's initializations were used for all models. For the FBCSP algorithm, filter-banks and feature extraction techniques were adopted from Kan Kai Ang et al. approach [9–10], and standard Linear Discriminant Analysis was used as the feature's classifier. As Dataset I was initially used for competition, results are reported as Kappa-Cohen coefficient κ calculated using the following equation:

$$\kappa = \frac{p_o - p_e}{1 - p_o} \tag{13}$$

where p_o is the classification accuracy p_e is the proportion of times the MI classes are expected to agree by chance. Such a metric was chosen following the metric used by the competition-winning approach, FBCSP. Table 2 shows the classification accuracy while Table 3 shows the obtained kappa coefficient values and its mean for all classifiers. In addition, Table 4 shows the average performance in terms of precision, recall and ROC's AUC scores for each architecture.

A slight increase in κ was observed on models adapted with Sinc-based convolutions (+0.048 for EEGNeT and +0.034 for CSPNN). A mean kappa of 0.598 was obtained for FBCSP, almost identical to the result reported in [9] (0.599). Our implementation of Shallow ConvNet obtained a slightly better mean than FBCSP, EEGNeT and TA-CSPNN; however, not with the Sinc-convolution adapted version of the latter two (difference of +0.036 with Sinc-EEGNeT and +0.025 with Sinc-CSPNN). From the obtained mean κ values, all architectures are found not to be significantly different from each other ($p > 0.3$).

3.2.1. *Network feature visualization and characterization on dataset I*

For cortical activity analysis, algorithm 1 described in section 2.2 was used to extract the spatial patterns learned by the network for a particular task n ($n \in \{Left\ hand, Right$

Table 2 – Classification performance (accuracy) on Dataset I.

Subject	FBCSP	Shallow ConvNet	EEGNet	TA-CSPNN	Sinc-EEGNet	Sinc-CSPNN
A1	0.74	0.68	0.75	0.76	0.86	0.82
A2	0.47	0.56	0.49	0.47	0.54	0.51
A3	0.76	0.79	0.76	0.78	0.85	0.81
A4	0.55	0.53	0.58	0.49	0.52	0.52
A5	0.61	0.65	0.64	0.63	0.67	0.65
A6	0.51	0.55	0.54	0.56	0.52	0.58
A7	0.69	0.71	0.69	0.61	0.78	0.66
A8	0.69	0.68	0.76	0.78	0.83	0.78
A9	0.69	0.7	0.6	0.66	0.72	0.66
Ave.	0.63	0.65	0.65	0.64	0.70	0.67

* Denotes significant improvement over its non-sinc counterpart (paired T-test, $p < 0.05$)).

Table 3 – Classification performance (Kappa coefficient values κ) on Dataset I.

Subject	FBCSP	Shallow ConvNet	EEGNet	TA-CSPNN	Sinc-EEGNet	Sinc-CSPNN
A1	0.739	0.683	0.716	0.749	**0.793**	0.752
A2	**0.475**	0.430	0.455	0.418	0.408	0.395
A3	0.752	0.759	0.736	0.766	0.761	**0.794**
A4	0.484	**0.512**	0.481	0.485	0.500	0.491
A5	0.601	0.581	0.613	0.592	**0.644**	0.627
A6	0.347	0.438	0.336	0.401	0.316	**0.393**
A7	0.64	0.695	0.680	0.590	**0.705**	0.644
A8	0.682	0.688	0.752	0.744	**0.775**	0.749
A9	0.663	**0.669**	0.578	0.631	0.659	0.641
Mean κ	0.598	0.606	0.594	0.597	**0.642**	**0.631**

Table 4 – Average classification performance of all models II on dataset I.

Model	Precision	Recall	AUC Score
FBCSP	0.640	0.621	0.610
ShallowConvNet	0.678	0.638	0.672
EEGNet	0.642	0.629	0.640
TA-CSPNN	0.646	0.645	0.644
Sinc-EEGNet	0.733	0.729	0.732
Sinc-CSPNN	0.680	0.716	0.691

hand, *Tongue*, *Feet*}). Here, trials from each task were fed to the network, and the spatial filter index [*i_max*,*j_max*], which corresponds to weights that causes the highest neuronal RMS activation on the feature map is extracted. Fig. 4a and 5a show an example of the obtained highest index for each task *n* for subject 3, and its corresponding spatial filters plotted as a topographic head map for Sinc-EEGNet and Sinc-CSPNN, respectively. Subject 3 was chosen as an example as it gave the best accuracy (up to 83% decoding accuracy) among all subjects.

The bar graph located at the upper figure represents the top 4 indexes with a high percentage of activation during the trials of a particular task *n*. Since index *i_max* in [*i_max*, *j_max*] also corresponds to the *i-th* Sinc-based filter which performs band-pass on the EEG data, the frequency response of the *i-th* filter is plotted (as shown in Fig. 4b and 5b to identify the frequency bands that the individual spatial kernel focuses using methods that were previously mentioned in section 2.2.2. In addition, as mentioned in the same section, to identify the frequency band that the whole network focuses on, the cumulative frequency response of all Sinc filters and comparisons with its non-Sinc counterpart is shown in Fig. 6.

3.3. *Classification performance on Dataset II*

In this Dataset, out of 4 trials per task, 3 of the trials were used as training with 10% left for validation. This results in a total of 1179 (393×3) of training data since sliding window strategy was used on each trial. Testing on the proposed model was done on the last trial. Again, similar to the Dataset I, all models were trained using ten sets of weight initialization. Table 3 shows the average test accuracy obtained on all MI and MI + MR tasks for all subjects, while Table 4 shows its performance in terms of its Kappa performance. Similar to the previous Dataset, the average performance in precision, recall, and ROC's AUC scores is shown in Table 5. Fig. 7 summarizes each model's average accuracy to highlight its performance between MI and MI + MR tasks (Table 6).

From Fig. 7, the average accuracy across all models for MI tasks does is almost similar. However, during the MI + MR task, an increase in accuracy can be observed in all models with the most apparent increase observed for sinc adapted CNN architectures. Compared with the MI counterpart, an average of 20% increase is observed with Sinc-EEGNeT and a slightly lower 15% increase for Sinc-CSPNN (Table 7).

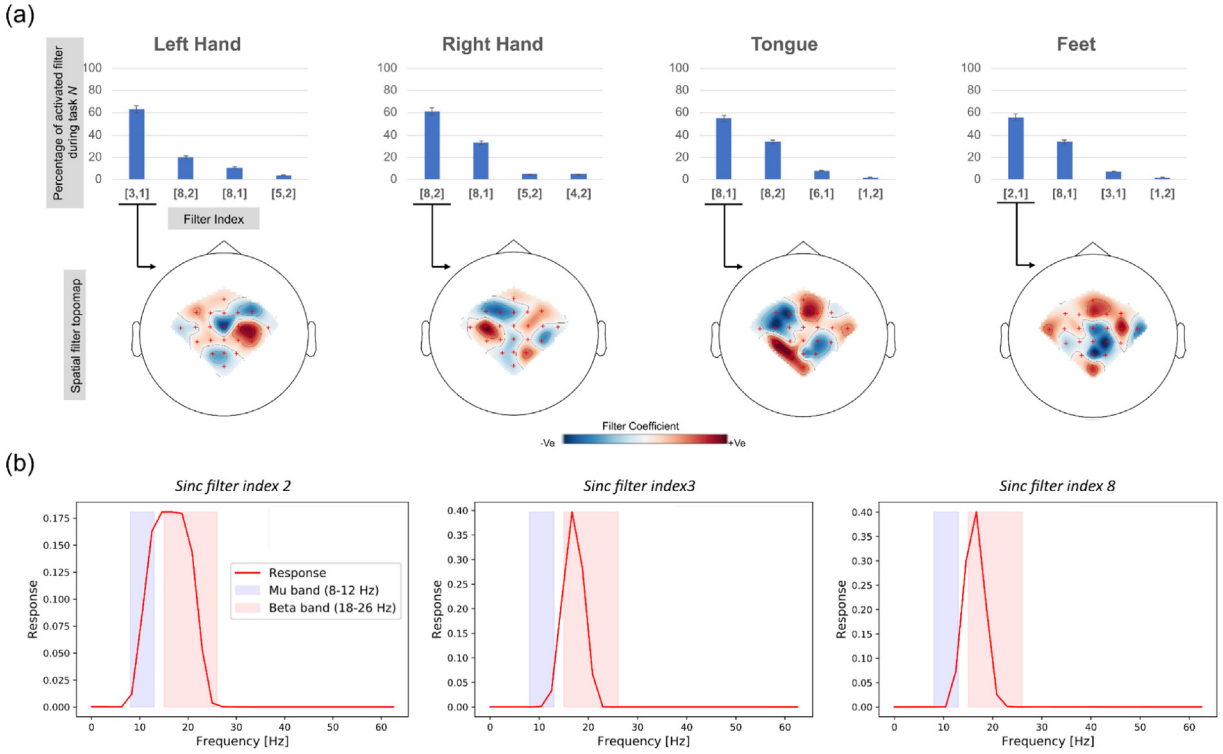


Fig. 4 – Example of subject 3: (a) Percentage of highest obtained index [i_*max*,j_*max*] for particular task N and its corresponding spatial filter obtained using proposed algorithm 1 for Sinc-EEGNet; (b) Index i_*max* individual sincNet filter response.

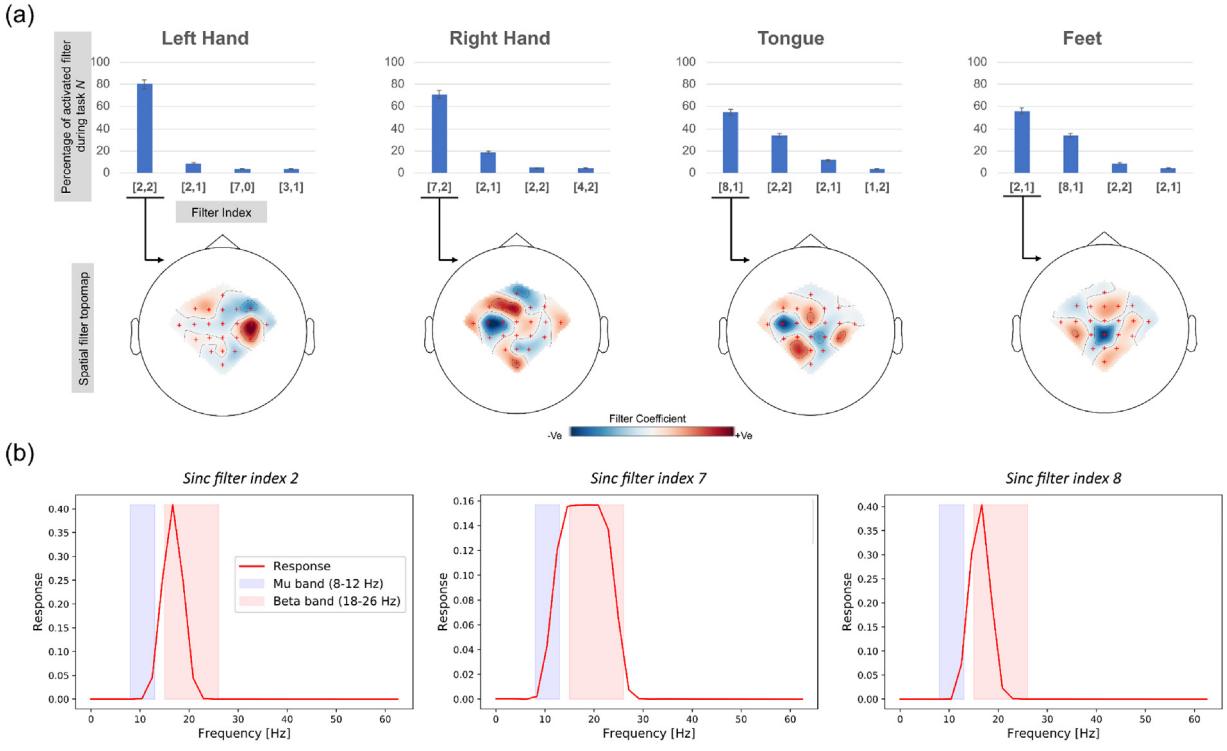


Fig. 5 – Example of subject 3: (a) Percentage of highest obtained index [i_*max*,j_*max*] for particular task N and its corresponding spatial filter obtained using proposed algorithm 1 for Sinc-CSPNN; (b) Index i_*max* individual sincNet filter response.

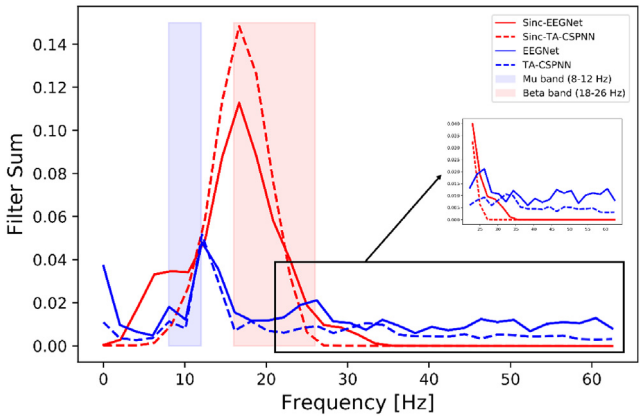


Fig. 6 – Cumulative frequency response of sinc based CNN models versus its non-sinc counterpart.

3.3.1. *Network feature visualization and characterization on Dataset II*

Similar method described in section 3.3 was again used for cortical activity analysis on Dataset II. Unlike Dataset, I, feature visualization and characterization on this Dataset were used to evaluate the proposed network performance in highlighting differences in cortical activity between MI and MI + MR tasks, such as shown in Fig. 8 for subject 12. Since the number on task n is now 2 ($n \in \{Left\ hand, Right\ hand\}$), the highest obtained index for MI and MI + MR task corresponds to the index that gives the highest activation during *Left hand* or *Right-hand* movement. Similar to the method mentioned in section 3.3 again, spatial filters corresponding to the obtained index is plotted as the topographic head map, such as shown in the lower part of Fig. 6, highlighting areas that the proposed network emphasized during the particular task.

Another important characteristic that should be identified in order to differentiate between MI and MI + MR task is the difference in cortical rhythms that appear during the executions of each task. For this purpose, it is necessary to perform frequency analysis on the learned network's sincNet filters to emphasize the cortical frequency band that the network focuses. Such is depicted in Fig. 9, in which the average frequency response of all Sinc filters from all subjects is plotted for both MI and MI + MR tasks.

3.4. *Training performance and characteristics*

For all CNN models, the use of early stopping was adopted. This strategy allows stopping the NN training once the loss on the validation set worsens (i.e., the network's performance stops improving on the validation data) and to prevent overfitting. Fig. 10 shows the average validation-loss curves obtained during training. It can be observed that the validation accuracy improves after each epoch, while the validation loss decreases as the number of epoch increases. This indicates the initialize networks (including the proposed) converge as training progresses. It also can be observed that, due to the early stopping strategy employed, the number of epochs needed for training differs on each architecture.

Since it is assumed that the reduction of parameters led to a reduction in the network's total training time, to verify this, the total number of epochs requires for each architecture (to achieve plateau on validation loss and accuracy) is averaged across all subjects and all folds, such as shown in Table 8 below. From the table, it can be observed that the earlier hypothesis is invalidated as the reduction in the number of parameters does not lead to a reduction in the number of epochs required for the network to achieve steady accuracy. Such is obvious for the case of ShallowConvNet, in which this benchmarking network contains the largest number of trainable parameters but having the least number of epochs required for training. In the case for EEGNeT and CSPNN, however, adopting the proposed sinc convolution layer does in fact cause a slight reduction in the number of epochs required for training, especially in the case of EEGNeT.

3.5. *Ablation: classification of FBCSP's extracted feature using SincNet*

To validate the effectiveness of the adaptive band-pass filtering of the earlier sincNet layer, the sinc convolutional and depthwise convolutional layers of the proposed Sinc-EEGNeT were removed, and the remaining layers were used to classify features extracted from FBCSP algorithm. For this ablation experiment, Sinc-EEGNeT was chosen as it gave the best performance among the proposed sinc based CNN architectures. The sinc convolutional and depthwise convolutional layers were removed as both of these layers mimic FBCSP, albeit being an adaptive one. The resulting architecture consists of only a CNN with a separable convolution layer and a fully connected layer. Unlike the originally proposed Sinc-EEGNeT architecture, output features of the FBCSP algorithm is a matrix with a size of $C \times T$ (because no downsampling was performed) hence, a kernel of size $K = 63$ was used on the separable convolution layer with the number of kernel F_c is maintained at 16. Due to changes in kernel size, the network was re-trained with data from Dataset I and II.

Table 9 below summarizes the obtained average performance metrics when classifying features extracted using FBCSP with the ablated Sinc-EEGNeT. Based on the obtained results, it can be concluded that such classification scheme resulted only in a performance almost similar to the reported classification of using FBCSP with LDA. This outcome is indicative that adapting the sinc convolutional layer with CNN architectures such as EEGNeT or CSPNN enables a better adaptation towards the relevant frequency bands, providing better classification performance.

4. Discussion

4.1. *SOTA performance with a fewer number of parameters*

Based on the obtained results presented in section 3.2 and section 3.3, a slight improvement in decoding accuracy can be observed with the proposed model compared to state-of-the-art (SOTA) models such as ShallowNet and EEGNet. In this case, replacing the first CNN layers with proposed SincNet results in slight improvement with fewer trainable parameters, such as shown in Table 1 27% reduction in the number

Table 5 – Classification performance(percentage) of all CNN models on Dataset II.

	Mental imagery task (MI)					Mental rotation associated motor imagery task (MI+MR)					
Subject	FBCSP	Shallow ConvNet	EEGNet	TA-CSPNN	**Sinc-EEGNet**	**Sinc-CSPNN**	FBCSP	ShallowConvNet	EEGNet	TA-CSPNN	**Sinc-EEGNet**	**Sinc-CSPNN**
1	0.551	0.554	0.542	0.499	**0.574**	0.535	0.681	0.723	0.751	0.718	**0.789**	0.752
2	0.722	0.787	0.721	0.827	**0.849**	0.846	0.541	0.504	0.577	0.541	0.575	**0.629**
3	0.523	0.452	0.464	0.478	0.459	**0.675**	0.585	0.508	0.643	0.462	**0.815**	0.703
4	0.487	0.472	0.476	0.454	**0.631**	0.564	0.479	0.512	0.678	0.581	0.819	**0.828**
5	**0.710**	0.666	0.585	0.494	0.668	0.496	0.841	0.869	0.907	0.609	**0.909**	0.639
6	0.790	0.828	0.783	0.805	0.826	**0.845**	0.442	0.507	0.469	**0.573**	0.548	0.434
7	**0.482**	0.412	0.293	0.415	0.392	0.367	0.477	0.337	0.532	0.648	**0.663**	0.556
8	0.255	0.156	0.324	0.218	0.334	**0.495**	0.514	0.568	0.522	**0.605**	0.555	0.554
9	0.271	0.222	0.256	0.653	**0.424**	0.401	0.471	0.542	0.388	0.482	0.615	0.534
10	0.491	**0.492**	0.481	0.359	0.327	0.442	0.334	0.276	0.344	**0.352**	0.338	0.272
11	0.214	0.202	0.273	0.278	**0.346**	0.169	0.209	0.237	0.274	0.252	0.507	**0.603**
12	0.470	0.395	0.422	**0.483**	0.357	0.457	0.572	0.656	0.754	0.741	0.816	**0.865**
13	0.491	0.493	0.613	0.492	0.529	**0.549**	0.327	0.346	**0.759**	0.614	0.602	0.687
Mean.	0.477	0.472	0.484	0.497	0.517	0.526	0.498	0.507	0.584	0.552	0.658	0.620

Bold indicates highest recorded kappa for a particular subject.

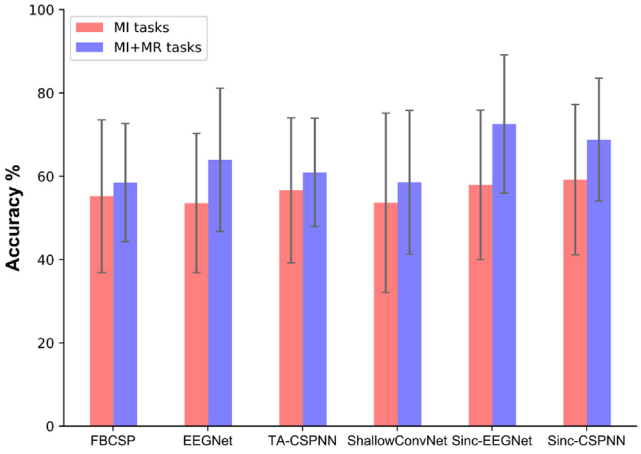


Fig. 7 – Average accuracy on all subjects for both MI and MI + MR tasks.

of trainable parameters compared to SOTA EEGNet, and 35% for TA-CSPNN).

Comparison between the proposed sinc based CNN architecture with the non-NN FBCSP approach on Dataset I yielded a slight improvement in mean kappa value, although such improvement is not statistically significant. Such a slight improvement is also notable in Dataset II classification results. It should be noted that although the proposed approach does not differ much in terms of classification performance, the NN based approach allows for end-to-end classification, that is, direct classification from the raw EEG data bypassing the needs for filtration, feature extraction and classification stage typically described in conventional MI decoding pipeline. This, in turn, eliminates the needs for manually tuning each element in the decoding pipeline as a NN based approach relies on automatically optimizing each network's layer during network training.

Although almost similar MI decoding performance can be observed in Dataset I and II, introducing mental rotation tasks during MI task (MI + MR) results in a general improvement in decoding accuracy in all models. This improvement is expected since it is hypothesized that such task allows the subject to engage in kinaesthetic motor imagery movement needed to ensure good performance while using a BCI system [31,32]. However, the best improvement in decoding accuracy during the MI + MR task can be observed in proposed Sinc based architectures (up to almost 90% accuracy in some subjects) with an average 20% improvement in Sinc-EEGNet and 15% for Sinc-CSPNN. Such performance may be attributed to SincNet's ability to focus and "notch" at a frequency band that only contains relevant information pertaining to a given task.

Besides, such an ability to highlight the necessary frequency band in the first layer of the network allows for the applicability of algorithm 1, which is used to extract learned spatial filters for neurophysiological interpretation and visualization. Another factor that may contribute to better classification in Dataset II is that unlike trial-wise decoding used in Dataset I, a sliding window over trial strategy may allow the CNN models to better adapt to only necessary frequency bands that contain relevant information. Such performance

Table 6 – Classification performance (Kappa coefficient).

	Mental imagery task (MI)						Mental rotation associated motor imagery task (MI+MR)					
Subject	FBCSP	Shallow ConvNet	EEGNet	TA-CSPNN	Sinc-EEGNet	Sinc-CSPNN	FBCSP	Shallow ConvNet	EEGNet	TA-CSPNN	Sinc-EEGNet	Sinc-CSPNN
1	0.551	0.554	0.542	0.499	**0.574**	0.535	0.681	0.723	0.751	0.718	**0.789**	0.752
2	0.722	0.787	0.721	0.827	**0.849**	0.846	0.541	0.504	0.577	0.541	0.575	**0.629**
3	0.523	0.452	0.464	0.478	0.459	**0.675**	0.585	0.508	0.643	0.462	**0.815**	0.703
4	0.487	0.472	0.476	0.454	**0.631**	0.564	0.479	0.512	0.678	0.581	0.819	**0.828**
5	**0.710**	0.666	0.585	0.494	0.668	0.496	0.841	0.869	0.907	0.609	**0.909**	0.639
6	0.790	0.828	0.783	0.805	0.826	**0.845**	0.442	0.507	0.469	**0.573**	0.548	0.434
7	**0.482**	0.412	0.293	0.415	0.392	0.367	0.477	0.337	0.532	0.648	**0.663**	0.556
8	0.255	0.156	0.324	0.218	0.334	**0.495**	0.514	0.568	0.522	**0.605**	0.555	0.554
9	0.271	0.222	0.256	0.653	**0.424**	0.401	0.471	0.542	0.388	0.482	0.615	0.534
10	0.491	**0.492**	0.481	0.359	0.327	0.442	0.334	0.276	0.344	**0.352**	0.338	0.272
11	0.214	0.202	0.273	0.278	**0.346**	0.169	0.209	0.237	0.274	0.252	0.507	**0.603**
12	0.470	0.395	0.422	**0.483**	0.357	0.457	0.572	0.656	0.754	0.741	0.816	**0.865**
13	0.491	0.493	0.613	0.492	0.529	**0.549**	0.327	0.346	**0.759**	0.614	0.602	0.687
Mean.	0.477	0.472	0.484	0.497	0.517	0.526	0.498	0.507	0.584	0.552	0.658	0.620

Bold indicates highest recorded kappa for a particular subject.

was also reported in [33] in which the performance of deep CNN models for motor imagery decoding is generally better for a slice-wise windowed approach compared to a trial-wise decoding approach.

While the performance of both Sinc-EEGNet and Sinc-CSPNN are comparable, it should be noted that the former generally performs better that the latter. Such performance maybe attributed to relatively large size and the addition of extra layer in Sinc-EEGnet thus allowing for more information to be learned by the network. In comparison with its non-sinc counterpart, the use of sinc based convolutions reduces the amount of overfitting, which are prone to occur in a network with larger number of parameters.

4.2. *Neurophysiological interpretability of adapted SincNet and spatial filters*

As mentioned earlier, spectral analysis on SincNet filters allows for neurophysiological interpretation of cortical rhythms that the learned network focuses on distinguishing between tasks. Here, spectral analysis on the Sinc layers was performed using the method described in section 2.2.1. Based on results shown in the lower part of Figs. 4 and 5, spectral analysis on the individual Sinc filters reveals that the adapted filters focus on bandpassing frequency located in the *mu* band (8–13 Hz) and the *beta* band (15–26 Hz). Such bands are known to neurophysiologically related to motor imagery movement, imagination, and visualization [34,35].

Apart from analysing sinc kernel individually, obtaining the cumulative frequency response of proposed architecture and comparing it with its non-sinc counterpart shows that the sincNet layer performed a "cleaner" band-pass with a notch-shaped filter starting at the *mu* band and peaked at the earlier *beta* band region, such as shown in Fig. 6. Such band-highlighting characteristics were also reported in the original SincNet literature [24] in which SincNet successfully adapted its characteristics to address the speaker identification task from the raw voice signal.

This characteristic was again validated in Dataset II in which sincNet was used to analysed differences in cortical rhythm between MI and MI + MR tasks. As shown in Fig. 9 section 3.3.1, Both models show a decrease in the SincNet filter's peak value, which lies in the *mu* band region for the MI + MR task. Another observation is that apart from decreasing, this peak is now shifted towards a higher *mu* region near the *beta* band region. The decreased in the filter's peak value during the MI + MR task is expected since subjects are hypothesized to suppress the mu rhythm better while performing these tasks.

It is interesting to note that apart from clearly highlighting the decrease in peak value in the *mu* band, visualization of SincNet's filter allows for the observation of a slight shift in the band-pass peak's position towards the *beta* region, which often associated with focused mental state, high arousal, and high alertness [36–38]. Such a state is hypothesized to be apparent during mental rotation tasks. In addition, beta waves have been shown to increase during concentration and immersion, particularly in frontal or occipital lobes [39], which is expected during MI + MR task.

Apart from spectral visualization of SincNet filters, this study also proposes algorithm 1 to extract and visualize the

Table 7 – Average classification performance of all models on Dataset II.

Model	MI Task			MI+MR Task		
	Precision	Recall	AUC Score	Precision	Recall	AUC Score
FBCSP	0.542	0.571	0.551	0.593	0.572	0.584
ShallowConvNet	0.533	0.525	0.528	0.677	0.611	0.610
EEGNet	0.601	0.556	0.556	0.606	0.448	0.651
TA-CSPNN	0.545	0.498	0.542	0.651	0.578	0.624
Sinc-EEGNet	0.661	0.694	0.625	0.780	0.766	0.751
Sinc-CSPNN	0.580	0.766	0.601	0.698	0.680	0.693

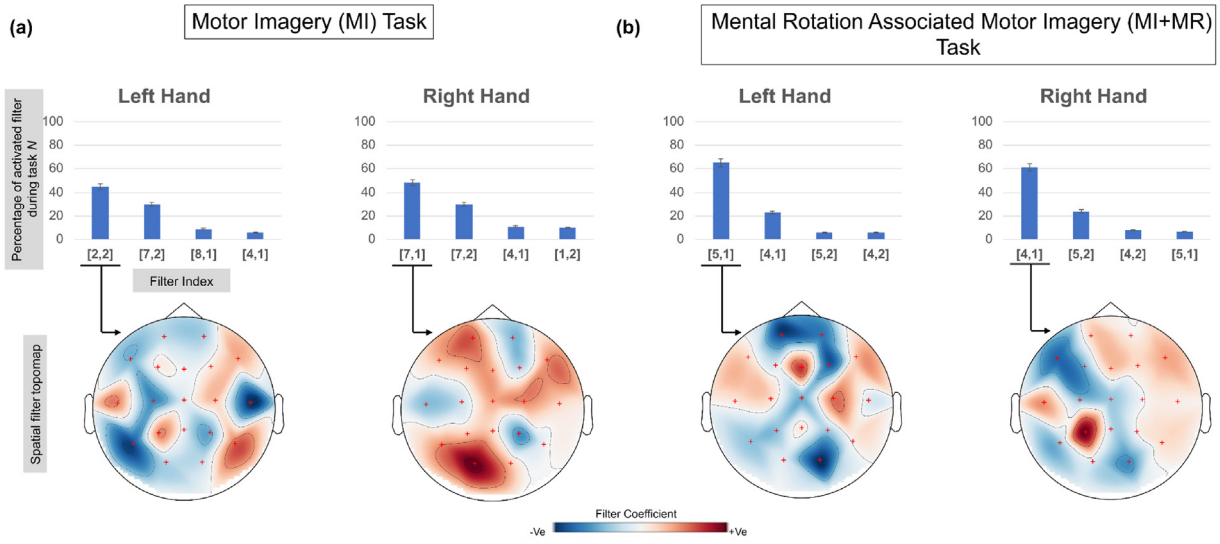


Fig. 8 – Subject 12 as an example: (a) highest obtained index [*i_max,j_max*] and its corresponding spatial filter visualization for MI task; (b) MI + MR task.

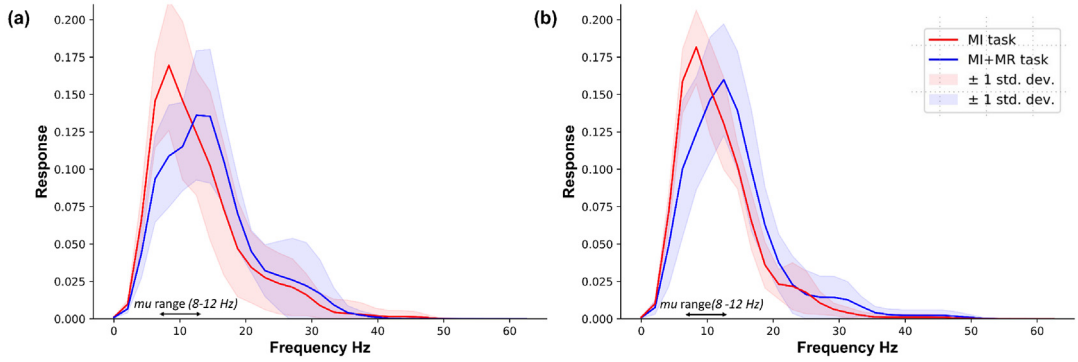


Fig. 9 – Average cumulative Sinc filters frequency response of all subjects: (a) Sinc-EEGNet; (b) Sinc-CSPNN.

proposed CNN model's spatial filters layer. By visualizing learned spatial filter weights as a topographic head map, analysis can be performed since these weights highlight cortical areas that the network focuses on to solve the classification task. In both Dataset I and Dataset II, areas corresponding to weights emphasized by the network (weights with relatively large coefficient value) were neurophysiologically related to the task being performed. For example, in Dataset I (Figs. 4 and 5, both Sinc-EEGNet and Sinc-CSPNN emphasize cortical motor areas C_3 and C_4 during *right* and *left* hand imagery movement, respectively. Similar results can also be observed in Dataset II, albeit with a different topographic EEG layout. An interesting observation in this dataset is that, compared to MI task, the proposed Sinc-EEGNet was able to emphasize weights corresponding to the prefrontal areas in the MI + MR task. Such emphasis was previously mentioned to occur during this task since the beta wave presence is ordinarily apparent in frontal and occipital lobes.

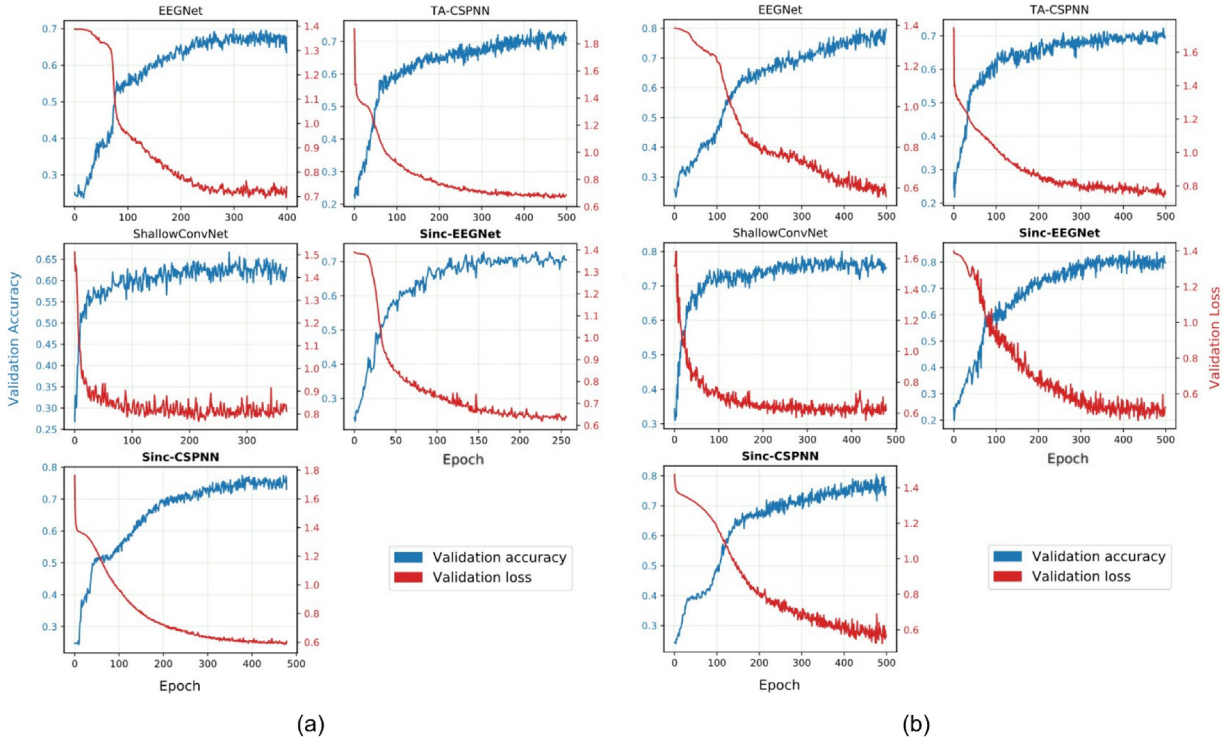


Fig. 10 – Average validation accuracy-loss curves for: (a) **Dataset I (secondary data)** (b) **Dataset II (primary data).**

Table 8 – Training time characteristics of tested CNN architectures.

	Shallow-ConvNet	EEGNeT	TA-CSPNN	**Sinc-EEGNeT**	**Sinc-CSPNN**
Average number of epochs	336 (± 98)	341 (± 88)	350 (± 99)	328 (± 88)	347 (± 89)
Time per epoch [ms]	46.42	30.55	20.86	22.42	21.65

Table 9 – Average performance metrics across all subjects for FBCSP-CNN classification.

Dataset	Accuracy	Precision	Recall	Kappa	AUC
Dataset I	0.61	0.631	0.625	0.517	0.611
Dataset II (MI task)	0.52	0.545	0.565	0.467	0.549
Dataset II (MI+MR task)	0.55	0.587	0.575	0.512	0.552

5. Conclusion

This paper presented two compact CNN architectures for decoding motor imagery signals based on SincNet. Compared to other SOTA models, the proposed models perform at par or, in some cases better, with the least number of trainable parameters. The proposed architecture was also shown to better adapt to the necessary cortical rhythms related to the given task and perform a cleaner band-pass filtering over these rhythms. Besides spectral visualization and analysis, SincNet also enables the development algorithm 1, in which spatial filters in the CNN's depthwise convolution layers are extracted for tasks related to cortical analysis.

In the future, the applicability of proposed Sinc based models will be expanded to other biomedical domains. One example is similar to the work in [40] in which SincNet model

is used for emotion classification from EEG signals. Another interesting application that could be explored is the use of proposed model towards epileptic seizure detection [41,42] in which the interpretability of SincNet could benefits clinicians and healthcare practitioners towards providing accurate epilepsy diagnostic. Classification of other BCI paradigms such as the recent concurrent SSVEP and P300 EEG features [43] can also be explored using the proposed model. Apart from EEG, the proposed algorithm can also be applied towards other biosignal classification, such as detecting heart failures from electrocardiogram (ECG) signals [44–46], detecting amytrophic lateral sclerosis (ALS) disease from EMG [47] and distinguishing audiovisual inputs for cocktail party problem [48], just to name a few.

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.

Acknowledgement

The author would like to thank Universiti Teknikal Malaysia Melaka (UTeM) Center for Robotics and Industrial Automation (CERIA), and Universiti Teknologi Malaysia (UTM), Biomedical Instrumentation and Electronic (bMIE) research group for the financial and facilities support.

Compliance with ethical standards

All procedures performed in this study were in accordance with the ethical standards of the institutional and national research committee and in a compliance with the 1964 Helsinki Declaration or its later amendments. For Dataset II (primary data) ethical approval was obtained from the Ministry of Health Malaysia (**NMRR-19-1671-47228)** and informed consent was given and retrieved from all individuals participating in the study.

Appendix A

Table A1 – Summary of parameters of the proposed Sinc based architecture.

Sinc-EEGNet			Sinc-CSPNN		
Layer	Filter size	Output	Layer	Filter size	Output
Input		$(1, C, T)$	Input		$(1, C, T)$
Sinc Convolution 2D	$F_K \times 2$	(F_K, C, T)	Sinc Convolution 2D	$F_K \times 2$	(F_K, C, T)
Batch Normalisation		(F_K, C, T)	Batch Normalisation		(F_K, C, T)
Depthwise Convolution	$C \times F_s$	$(F_K \times F_s, 1, T)$	Depthwise Convolution	$C \times F_s$	$(F_K \times F_s, 1, T)$
Activation: ELU		$(F_K \times F_s, 1, T)$	Activation: RELU		$(F_K \times F_s, 1, T)$
Average pooling		$(F_K \times F_s, 1, T/4)$	Average pooling		$(F_K \times F_s, 1, T/4)$
Dropout		$(F_K \times F_s, 1, T/4)$	Dropout		$(F_K \times F_s, 1, T/4)$
Separable Convolution	$(16 \times F_K \times F_s) + (F_C \times F_K \times F_s)$	$(F_C, 1, T/4)$	Flatten		$(F_K \times F_s)$
BatchNorm		$(F_C, 1, T/4)$	Dense		N
Activation: ELU		$(F_C, 1, T/4)$	Softmax		N
Average pooling		$(F_C, 1, T/32)$			
Dropout		$(F_C, 1, T/32)$			
Flatten		$(F_C \times (T/32))$			
Dense		N			
Softmax		N			

REFERENCES

[1] Rao RPN. Brain-computer interfacing: an introduction. Cambridge University Press; 2011. 10.1017/CBO9781139032803.

[2] Lebedev MA, Nicolelis MAL. Brain-machine interfaces: from basic science to neuroprostheses and neurorehabilitation. Physiol Rev 2017;97:767–837. https://doi.org/10.1152/physrev.00027.2016.

[3] Abdulkader SN, Atia A, Mostafa M-S. Brain computer interfacing: applications and challenges. Egypt Informatics J 2015;16(2):213–30. https://doi.org/10.1016/j.eij.2015.06.002.

[4] Amiri S, Fazel-Rezai R, Asadpour V. A review of hybrid brain-computer interface systems. Adv Human-Computer Interact 2013;2013:1–8. https://doi.org/10.1155/2013/187024.

[5] Brodu N, Lotte F, Lecuyer A. Comparative study of band-power extraction techniques for Motor Imagery classification. In: IEEE Symp. Comput. Intell Cogn. Algorithms, Mind, Brain. IEEE; 2011. p. 1–6. 10.1109/CCMB.2011.5952105.

[6] Wang Y, Gao S, Gao X. Common spatial pattern method for channel selelction in motor imagery based brain-computer. Interface 2006:5392–5. https://doi.org/10.1109/iembs.2005.1615701.

[7] Jin J, Miao Y, Daly I, Zuo C, Hu D, Cichocki A. Correlation-based channel selection and regularized feature optimization for MI-based BCI. Neural Networks 2019;118:262–70. https://doi.org/10.1016/j.neunet.2019.07.008.

[8] Jin J, Fang H, Daly I, Xiao R, Miao Y, Wang X, et al. Optimization of model training based on iterative minimum covariance determinant in motor-imagery BCI. Int J Neural Syst 2021;31:2150030. https://doi.org/10.1142/S0129065721500301.

[9] Ang KK, Chin ZY, Wang C, Guan C, Zhang H. Filter bank common spatial pattern algorithm on BCI competition IV datasets 2a and 2b. Front Neurosci 2012;6:1–9. https://doi.org/10.3389/fnins.2012.00039.

[10] Kai Keng Ang, Zheng Yang Chin, Haihong Zhang, Cuntai Guan, Filter Bank Common Spatial Pattern (FBCSP) in Brain-Computer Interface, in: 2008 IEEE Int. Jt. Conf. Neural Networks (IEEE World Congr. Comput. Intell., 2008: pp. 2390–2397. 10.1109/IJCNN.2008.4634130.

[11] Jin J, Xiao R, Daly I, Miao Y, Wang X, Cichocki A. Internal feature selection method of CSP based on L1-norm and dempster-shafer theory. IEEE Trans Neural Networks Learn Syst PP 2020:1–12. https://doi.org/10.1109/TNNLS.2020.3015505.

[12] Wang H, Tang C, Xu T, Li T, Xu L, Yue H, et al. An approach of one-vs-rest filter bank common spatial pattern and spiking neural networks for multiple motor imagery decoding. IEEE Access 2020;8:86850–61. https://doi.org/10.1109/ACCESS.2020.2992631.

[13] Barachant A, Bonnet S, Congedo M, Jutten C. Multiclass brain-computer interface classification by riemannian geometry. IEEE Trans Biomed Eng 2012;59(4):920–8. https://doi.org/10.1109/TBME.2011.2172210.

[14] Barachant A, Bonnet S, Congedo M, Jutten C. Classification of covariance matrices using a Riemannian-based kernel for BCI applications. Neurocomputing 2013;112:172–8. https://doi.org/10.1016/j.neucom.2012.12.039.

[15] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: IEEE Conf. Comput. Vis. Pattern Recognit. IEEE; 2016. p. 770–8. 10.1109/CVPR.2016.90.

[16] Sainath TN, Kingsbury B, Saon G, Soltau H, Mohamed AR, Dahl G, et al. Deep convolutional neural networks for large-scale speech tasks. Neural Networks 2015;64:39–48. https://doi.org/10.1016/j.neunet.2014.08.005.

[17] Schirrmeister RT, Springenberg JT, Fiederer LDJ, Glasstetter M, Eggensperger K, Tangermann M, et al. Deep learning with convolutional neural networks for EEG decoding and visualization. Hum Brain Mapp 2017;38(11):5391–420. https://doi.org/10.1002/hbm.23730.

[18] Tayeb Z, Fedjaev J, Ghaboosi N, Richter C, Everding L, Qu X, et al. Validating deep neural networks for online decoding of motor imagery movements from eeg signals. Sensors (Switzerland) 2019;19. https://doi.org/10.3390/s19010210.

[19] Amin SU, Alsulaiman M, Muhammad G, Bencherif MA, Hossain MS. Multilevel weighted feature fusion using convolutional neural networks for EEG motor imagery classification. IEEE Access 2019;7:18940–50. https://doi.org/10.1109/ACCESS.2019.2895688.

[20] Tang X, Wang T, Du Y, Dai Y. Motor imagery EEG recognition with KNN-based smooth auto-encoder. Artif Intell Med 2019;101. https://doi.org/10.1016/j.artmed.2019.101747 101747.

[21] Lawhern VJ, Solon AJ, Waytowich NR, Gordon SM, Hung CP, Lance BJ. EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces. J Neural Eng 2018;15. https://doi.org/10.1088/1741-2552/aace8c 056013.

[22] Mousavi M, de Sa VR. Temporally adaptive common spatial patterns with deep convolutional neural networks. In: 41st Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. IEEE; 2019. p. 4533–6. 10.1109/EMBC.2019.8857423.

[23] Tang Z, Li C, Sun S. Single-trial EEG classification of motor imagery using deep convolutional neural networks. Optik (Stuttg) 2017;130:11–8. https://doi.org/10.1016/j.ijleo.2016.10.117.

[24] Ravanelli M, Bengio Y, Recognition S. Speaker recognition from raw waveform with SincNet. In: IEEE Spok. Lang. Technol. Work. IEEE; 2018. p. 1021–8. 10.1109/SLT.2018.8639585.

[25] Olivas-Padilla BE, Chacon-Murguia MI. Classification of multiple motor imagery using deep convolutional neural networks and spatial filters. Appl Soft Comput J 2019;75:461–72. https://doi.org/10.1016/j.asoc.2018.11.031.

[26] JEL (Hanoi) H, CG, Roberts A. DDSP:Differentiable Digital Signal Processing, in: Int. Conf. Learn. Represent., 2020: pp. 1–19.

[27] Springenberg JT, Dosovitskiy A, Brox T, Riedmiller M. Striving for simplicity: The all convolutional net, in: 3rd Int. Conf. Learn. Represent. ICLR 2015 – Work. Track Proc., 2015.

[28] Wolpaw JR, McFarland DJ, Vaughan TM. Brain-computer interface research at the Wadsworth Center. IEEE Trans Rehabil Eng 2000. https://doi.org/10.1109/86.847823.

[29] Tangermann M, Müller K-R, Aertsen A, Birbaumer N, Braun C, Brunner C, et al. Review of the BCI competition IV. Front Neurosci 2012;6. https://doi.org/10.3389/fnins.2012.00055.

[30] Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 2014;15:1929–58.

[31] Jeunet C, N'Kaoua B, Subramanian S, Hachet M, Lotte F, Friedman D. Predicting mental imagery-based BCI performance from personality, cognitive profile and neurophysiological patterns. PLoS ONE 2015;10(12):e0143962. https://doi.org/10.1371/journal.pone.0143962.

[32] Kubler A, Holz E, Kaufmann T, Zickler C. A User Centred Approach for Bringing BCI Controlled Applications to End-Users, in: Brain-Computer Interface Syst. - Recent Prog. Futur. Prospect., InTech, 2013. https://doi.org/10.5772/55802.

[33] Ma X, Qiu S, Wei W, Wang S, He H. Deep channel-correlation network for motor imagery decoding from the same limb. IEEE Trans Neural Syst Rehabil Eng 2020;28(1):297–306. https://doi.org/10.1109/TNSRE.2019.2953121.

[34] Garcia-Rill E. The 10Hz Fulcrum, in: Waking Reticular Act. Syst. Heal. Dis., Elsevier, 2015: pp. 157–170. 10.1016/B978-0-12-801385-4.00008-2.
[35] Kübler A, Mattia D. Brain-computer interface based solutions for end-users with severe communication disorders. In: Neurol Conciousness. Elsevier; 2016. p. 217–40. 10.1016/B978-0-12-800948-2.00014-5.
[36] Abhang PA, Gawali BW, Mehrotra SC. Technical Aspects of Brain Rhythms and Speech Parameters, in: Introd. to EEG-Speech-Based Emot. Recognit., Elsevier, 2016: pp. 51–79. https://doi.org/10.1016/B978-0-12-804490-2.00003-8.
[37] Satapathy SK, Dehuri S, Jagadev AK, Mishra S. Introduction. In: EEG Brain Signal Classif Epileptic Seizure Disord Detect. Elsevier; 2019. p. 1–25. 10.1016/B978-0-12-817426-5.00001-6.
[38] Izzuddin TA, Safri NM, Othman MA. Mental imagery classification using one-dimensional convolutional neural network for target selection in single-channel BCI-controlled mobile robot. Neural Comput Appl 2020. https://doi.org/10.1007/s00521-020-05393-6.
[39] Lim S, Yeo M, Yoon G. Comparison between concentration and immersion based on EEG analysis. Sensors 2019;19:1669. https://doi.org/10.3390/s19071669.
[40] Zeng H, Wu Z, Zhang J, Yang C, Zhang H, Dai G, et al. EEG emotion classification using an improved sincnet-based deep learning model. Brain Sci 2019;9. https://doi.org/10.3390/brainsci9110326.
[41] Emami A, Kunii N, Matsuo T, Shinozaki T, Kawai K, Takahashi H. Seizure detection by convolutional neural network-based analysis of scalp electroencephalography plot images. NeuroImage Clin 2019;22. https://doi.org/10.1016/j.nicl.2019.101684 101684.
[42] Zhou M, Tian C, Cao R, Wang B, Niu Y, Hu T, et al. Epileptic seizure detection based on EEG signals and CNN. Front Neuroinform 2018;12. https://doi.org/10.3389/fninf.2018.00095.
[43] Xu M, Han J, Wang Y, Jung T-P, Ming D. Implementing Over 100 command codes for a high-speed hybrid brain-computer interface using concurrent P300 and SSVEP features. IEEE Trans Biomed Eng 2020;67(11):3073–82. https://doi.org/10.1109/TBME.2020.2975614.
[44] Acharya UR, Oh SL, Hagiwara Y, Tan JH, Adam M, Gertych A, et al. A deep convolutional neural network model to classify heartbeats. Comput Biol Med 2017;89:389–96. https://doi.org/10.1016/j.compbiomed.2017.08.022.
[45] Porumb M, Iadanza E, Massaro S, Pecchia L. Biomedical signal processing and control a convolutional neural network approach to detect congestive heart failure. Biomed Signal Process Control 2020;55. https://doi.org/10.1016/j.bspc.2019.101597 101597.
[46] Abdul-Kadir NA, Mat Safri N, Othman MA. Atrial fibrillation classification and association between the natural frequency and the autonomic nervous system. Int J Cardiol 2016;222:504–8. https://doi.org/10.1016/j.ijcard.2016.07.196.
[47] Sengur A, Akbulut Y, Guo Y, Bajaj V. Classification of amyotrophic lateral sclerosis disease based on convolutional neural network and reinforcement sample learning algorithm. Heal Inf Sci Syst 2017;5:9. https://doi.org/10.1007/s13755-017-0029-6.
[48] Li Y, Wang F, Chen Y, Cichocki A, Sejnowski T. The effects of audiovisual inputs on solving the cocktail party problem in the human brain: an fMRI study. Cereb Cortex 2018;28:3623–37. https://doi.org/10.1093/cercor/bhx235.