EEG-based Intention Recognition from Spatio-Temporal Representations via Cascade and Parallel Convolutional Recurrent Neural Networks

Dalin Zhang, Lina Yao, Xiang Zhang, Sen Wang, Weitong Chen, Robert Boots

Abstract

Brain-Computer Interface (BCI) is a system empowering humans to communicate with or control the outside world with exclusively brain intentions. Electroencephalography (EEG) based BCIs are promising solutions due to their convenient and portable instruments. Motor imagery EEG (MI-EEG) is a kind of most widely focused EEG signals, which reveals a subject's movement intentions without actual actions. Despite the extensive research of MI-EEG in recent years, it is still challenging to interpret EEG signals effectively due to the massive noises in EEG[1] signals (e.g., low signal noise ratio and incomplete EEG signals), and difficulties in capturing the inconspicuous relationships between EEG signals and certain brain activities. Most existing works either only consider EEG as chain-like sequences neglecting complex dependencies between adjacent signals or performing simple temporal averaging over EEG sequences. In this paper, we introduce both cascade and parallel convolutional recurrent neural network models for precisely identifying human intended movements by effectively learning compositional spatio-temporal representations of *raw* EEG streams. The proposed models grasp the spatial correlations between physically neighbouring EEG signals by converting the chain-like EEG sequences into 2D mesh-like hierarchy. A LSTM based recurrent network is able to extract the subtle temporal dependencies of EEG data streams. Extensive experiments on a large scale MI-EEG dataset (108 subjects, 3,145,160 EEG records) have demonstrated that both models achieve high accuracy near 98.3% and outperform a set of baseline methods and most recent deep learning based EEG recognition models, yielding a significant accuracy increase of **18%** in the *cross-subject* validation scenario.

Introduction

Brain-computer interface (BCI) enables users to directly communicate with the outside world or to control instruments using brain intentions alone, thus providing an alternatively practical way to help people who are suffering from severe motor disabilities. Recent research has also found its applications for healthy users, such as BCI games in entertainment industries (Ahn et al. 2014). Scalp-recording electroencephalography (EEG) is considered to be one of the most practical pathways to realize BCI systems due to its portable acquisition system and convenient implementation (Wang et al. 2014). Motor imagery EEG (MI-EEG) is a sort of EEG signals collected when a subject *imagines* performing specific actions (e.g., imagines opening eyes and lifting left arm) while does not make an actual movement. In this way, human intentions can be recognized by analyzing the EEG signals. It has been attracting increasing attentions, and various research has attempted to engage MI-EEG based BCI in real-world applications such as mind controlled wheelchairs (Wang et al. 2014), prosthetic (Bright et al. 2016) and exoskeletons (Qiu et al. 2017).

However, real-world EEG based BCI systems are still immature due to diverse open challenges. First, EEG signals usually have a mass of noises. Apart from the common noises of sensory systems, such as power line interference or inappropriate electrode connections, EEG signals have some unique inevitable noises. During the recording process, physiological activities like eye blinks, muscle activity and heart beat are all harm to collecting high signal-to-noise ratio EEG signals. It is hard to make sure that the participants concentrate on the performing tasks during the whole experiment period. Also, a typical EEG based BCI system usually has 8 to 128 signal channels resulting in limited signal resolution compared to image or video related tasks. Second, the correlations between the EEG signals and their corresponding brain intentions in deep structures are ambiguous. Unlike the body actions which can be easily explained by monitoring accelerometers or gyroscopes, it is not straightforward to infer the brain intentions by directly observing EEG signals. Third, widely utilized brain intention recognition methods heavily rely on handcrafted features, requiring extensive pre-processing before making a prediction (Sun and Zhou 2014). Some methods include signal de-noising (Heydari and Shahbakhti 2015) or feature selection steps (Yin et al. 2017) followed by final recognition model. Such a two-stage model is inconvenient to train and implement, and the whole process is time-consuming and highly dependent on professional knowledge in this domain. Finally, current work mainly targets either intra-subject (test data and train data are from the same subject) or binary EEG signal classification scenarios. Little research has been carried out on both cross-subject and multi-class scenarios. However, the cross-subject and multi-class scenarios are highly desired
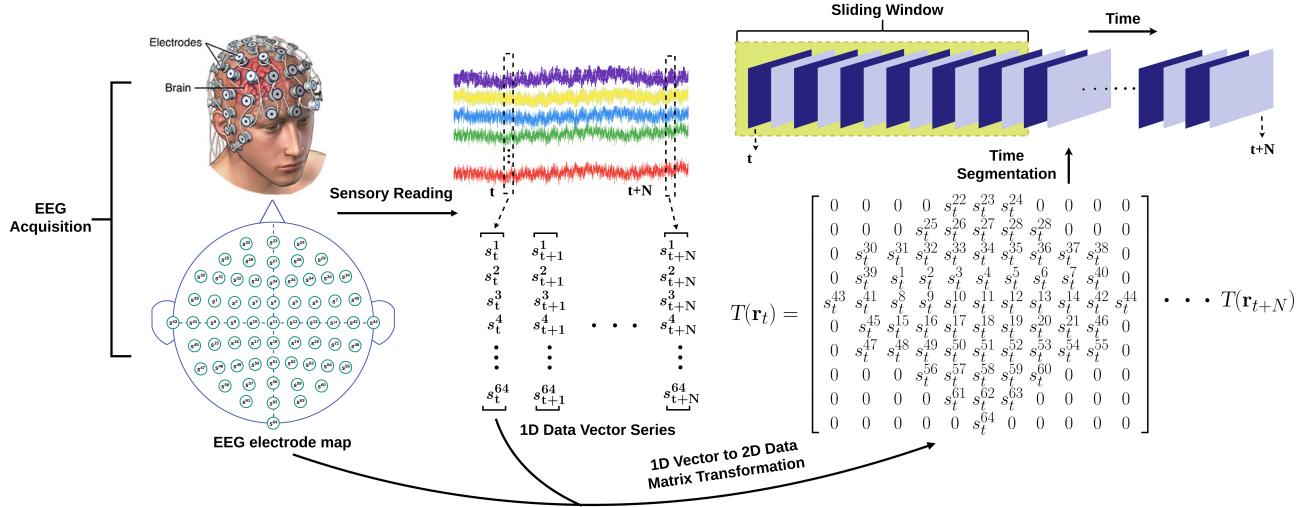
---

[1]In this paper, we will use the terms EEG and MI-EEG interchangeably.

Figure 1: EEG data acquisition and preprocessing. EEG signals are first captured using a BCI headset with multiple electrodes and recorded as time series 1D data vectors. These data vectors are then converted to 2D data meshes according to the electrode map of the data acquisition headset of the BCI system. The converted 2D meshes are finally segmented to clips using sliding window techniques.

for implementing real-world applications. Furthermore, even under intra-subject or binary classification scenarios, many existing works suffer poor performance near 80% accuracy.

In recent years, deep learning's revolutionary advances in audio and visual signals recognition have gained significant attentions (LeCun, Bengio, and Hinton 2015). Some recent deep learning based EEG classification approaches have enhanced the recognition accuracy (Bashivan et al. 2016; Tabar and Halici 2016). However, these approaches either focus on complex pre-processing, such as converting raw EEG signals to images (Bashivan et al. 2016), or neglecting the subtle spatial and temporal information contained within EEG signals. Hence, current methods still have limited capabilities in dealing with cross-subject and multi-class scenarios, with limited deployment in real-world applications.

To tackle the above obstacles for further developing EEG-based BCIs, we present in this paper, two kinds of convolutional recurrent neural networks, which we call cascade and parallel models, to detect human intentions through learning the effective compositional spatio-temporal dynamics from *raw* EEG streaming signals without preprocessing. In particular, we build a mesh-like raw EEG signal hierarchy from 1D chain-like EEG vectors by mapping the EEG recordings with the spatial information of EEG acquisition electrodes, to align the correlations between neighbouring EEG signals and corresponding brain areas. Next, both cascade and parallel convolutional recurrent network models are developed to decode robust EEG representations from both space and time dimensions in sequence or in parallel respectively. The proposed models are unified end-to-end trainable models, simultaneously learning the robust feature representations and classifying the EEG raw signals to detect intentional control thoughts. The proposed models have good generalization in more complex and practical scenarios (both cross-subject and

multi-class). Both the cascade and parallel models achieve high accuracy of near 98.3%, significantly outperforming state-of-the-art methods by near 18%.

The Proposed Method

In this section, we describe the detailed architectures of the proposed cascade and parallel convolutional recurrent network approaches.

Converting 1D EEG Sequences to 2D EEG Meshes

The overall EEG data acquisition and preprocessing flowchart of our proposed method is shown in Figure 1. The EEG based BCI system uses a wearable headset with multiple electrodes to capture the EEG signals. When a subject *imagines* performing a certain instruction, the electrodes of the headset acquire the fluctuations of the voltages from the scalp. The EEG electrode map in Figure 1 depicts the electrodes placement of an example BCI headset. The electrode map varies from different BCI systems according to the different number of recording channels. The sensory readings from the EEG acquisition system represent time series data at the acquiring frequency. Typically, the raw data from EEG signal acquisition system at time index t is a one-dimensional (1D) data vector $\mathbf{r}_t = [s_t^1, \ s_t^2, \ ... \ , s_t^n]^T$, where s_t^n is the reading data of the nth electrode channel and the acquisition system totally contains n channels. For the observation period $[t, \ t+N]$, there are $(N+1)$ 1D data vectors, each of which contains n elements corresponding to n electrodes of the acquisition headset.

From the EEG electrode map, it is observed that each electrode is physically neighboring multiple electrodes which measures the EEG signals in a certain area of brain, while the elements of the chain-like 1D EEG data vectors are restricted to two neighbors. Furthermore, different brain regions

correspond to different brain activities. From this conceptualization, we convert the 1D EEG data vectors to 2D EEG data meshes according to the spatial information of the electrode distribution of the acquisition system. The transformation function of the 1D data vector $\mathbf{r}_t$ at time stamp t for its corresponding 2D data mesh $\mathbf{m}_t$ is denoted as follows:

$$T(\mathbf{r}_t) = \begin{bmatrix} 0 & 0 & 0 & 0 & s_t^{22} & s_t^{23} & s_t^{24} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & s_t^{25} & s_t^{26} & s_t^{27} & s_t^{28} & s_t^{28} & 0 & 0 & 0 \\ 0 & s_t^{30} & s_t^{31} & s_t^{32} & s_t^{33} & s_t^{34} & s_t^{35} & s_t^{36} & s_t^{37} & s_t^{38} & 0 \\ 0 & s_t^{39} & s_t^{1} & s_t^{2} & s_t^{3} & s_t^{4} & s_t^{5} & s_t^{6} & s_t^{7} & s_t^{40} & 0 \\ s_t^{43} & s_t^{41} & s_t^{8} & s_t^{9} & s_t^{10} & s_t^{11} & s_t^{12} & s_t^{13} & s_t^{14} & s_t^{42} & s_t^{44} \\ 0 & s_t^{45} & s_t^{15} & s_t^{16} & s_t^{17} & s_t^{18} & s_t^{19} & s_t^{20} & s_t^{21} & s_t^{46} & 0 \\ 0 & s_t^{47} & s_t^{48} & s_t^{49} & s_t^{50} & s_t^{51} & s_t^{52} & s_t^{53} & s_t^{54} & s_t^{55} & 0 \\ 0 & 0 & 0 & s_t^{56} & s_t^{57} & s_t^{58} & s_t^{59} & s_t^{60} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & s_t^{61} & s_t^{62} & s_t^{63} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & s_t^{64} & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (1)$$

where the positions of the *null* electrodes are designated as "0", which has no effect in neural network. Through this transformation, the raw 1D data vector series $[\mathbf{r}_t, \ \mathbf{r}_{t+1}, \ ... \ \mathbf{r}_{t+N}]$ is converted to the 2D data mesh series $[\mathbf{m}_t, \ \mathbf{m}_{t+1}, \ ... \ \mathbf{m}_{t+N}]$, where for the observation duration $[t, \ t+N]$, the number of 2D data meshes is still $(N+1)$. After 2D data mesh transformation, the data mesh is normalized across the non-zero elements using Z-score normalization. Each of the resulted 2D data meshes contains the spatial information of the brain activity at its recording time. During the recording process, some EEG readings are variably missing largely due to issues of electrical conductivity and subjects movement, resulting in all channels recording zeros. This issue is unavoidable in sensor-based systems, and it might not be tolerated by BCIs. From the application point of view, smooth manipulation of the BCI system provides improved user experience. For this reason, a BCI system should preferably translate brain activities to the output information continuously without interruption. As missing information is a clinical reality, in this work we preserve the incomplete recordings which are discarded in previous work (Kim et al. 2016) to maintain the integrity of EEG signals. The experimental results show our model performed well despite these "missing readings".

Up to this point, we apply the sliding window approach to divide the streaming 2D meshes to individual clips as shown in the last step of Figure 1. Each clip has fixed length of time series 2D data meshes with 50% overlapping between continuous neighbors. The data meshes segment $\mathbf{S}_j$ is created as follows:

$$\mathbf{S}_j = [\mathbf{m}_t, \ \mathbf{m}_{t+1}... \ \mathbf{m}_{t+S-1}]$$

where S is the window size and $j = 1, 2, ..., q$ with q segments during the observation period. Our goal is to develop an effective model to recognize a set of human intentions $\mathbf{A} = [a_1, \ a_2, \ ... \ a_K]^T$ from each windowed data meshes segment $\mathbf{S}_j$. The recognition approach tries to predict the human intention $\mathbf{Y}_t \in \mathbf{A}$ performed during this windowed period.

Cascade Convolutional Recurrent Network

The cascade human intention recognition model represents a deep convolutional recurrent neural network framework
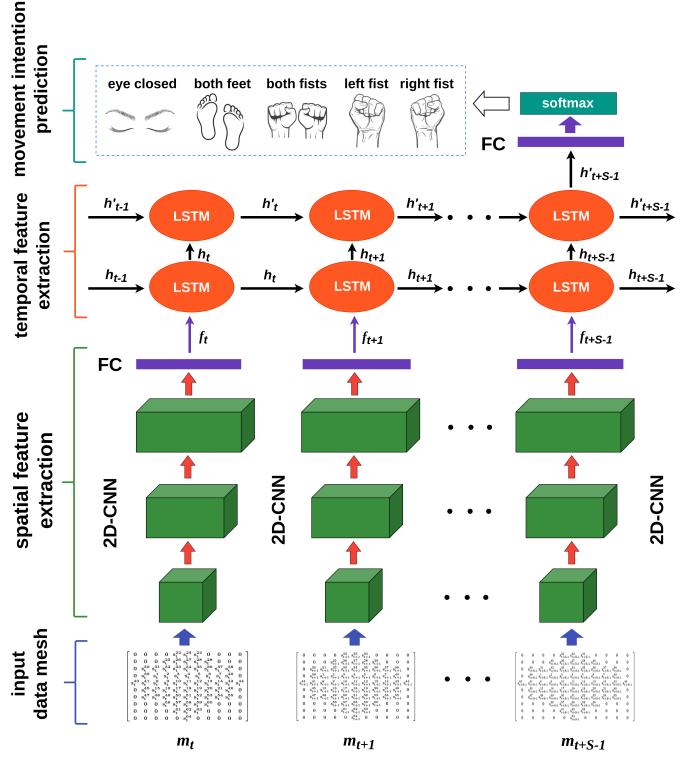

Figure 2: Cascade convolutional recurrent neural network architecture.

illustrated in Figure 2, capturing the spatial and temporal features in sequence. The input to the model is the preprocessed segment of 2D data meshes (e.g., $\mathbf{S}_j$), creating a 3D data architecture containing both spatial and temporal information. We first extract the spatial features of each data mesh, and then feed the sequence of the extracted spatial features into the RNN to extract temporal features. One fully connected layer receives the output of the last time step of the RNN layers, and feeds the softmax layer for final intention prediction.

To extract the spatial features of each data mesh, we apply a mesh-wise deep 2D-CNN as shown in Figure 2. The jth input segment is defined as $\mathbf{S}_j = [\mathbf{m}_t \ ... \ \mathbf{m}_{t+S-1}] \in \mathbb{R}^{S \times h \times w}$, where there are S data meshes denoted as $\mathbf{m}_k$ ($k = t, \ t+1, \ ... \ t+S-1$), and each data mesh is of size $h \times w$. The data meshes are input to a 2D-CNN individually, and each resolves to a spatial feature representation $\mathbf{f}_k$ ($k = t, \ t+1, \ ... \ t+S-1$):

$$\text{CasCNN:} \quad \mathbf{f}_k = C_{2D}(\mathbf{m}_k), \ \ \mathbf{f}_k \in \mathbb{R}^l.$$

The final spatial feature representation $\mathbf{f}_k$ is a feature vector with l elements. Through the 2D-CNN spatial feature extraction step, the input segments are transformed to sequences of spatial feature representations:

$$\text{CasCNN:} \quad \mathbf{S}_j \Rightarrow \mathbf{F}_j, \text{ where } \mathbf{F}_j = [\mathbf{f}_t \ ... \ \mathbf{f}_{t+S-1}] \in \mathbb{R}^{S \times l}.$$

Concretely, there are three 2D convolutional layers with the same kernel size of 3×3 for spatial feature extraction. In each

convolutional operation we use zero-padding techniques to prevent missing the information at the edge of the input data mesh. This creates feature maps with the same size as the raw input EEG data mesh of $h \times w$. We start the first convolutional layer with 32 feature maps, and double the feature maps in each of the following convolutional layers. As a result, there are 128 feature maps in the last convolutional layer. After these three convolutional layers, a fully connected layer with 1024 neurons is applied to convert the 128 feature maps to the final spatial feature representation $\mathbf{f}_k \in \mathbb{R}^{1024}$. This fully connected layer is optional for feeding the 2D-CNN results to RNN. However, we observe that this layer is essential in helping with convergence and marginally improvement of the performance of the whole framework.

The spatial feature representation sequence $\mathbf{F}_j$ is input to a RNN to computes the temporal features. We use Long Short-Term Memory (LSTM) units to construct two stacked RNN layers. LSTM is a modified RNN cell addressing the gradient vanishing and exploding problem. There are S LSTM units in each layer, and the input to the second RNN layer is the output time sequence of the previous RNN layer. The hidden state of the LSTM unit of the first RNN layer at current time step t is denoted as h_t, and the h_{t-1} is the hidden state of the previous time step $t-1$. The information from the previous time step is conveyed to the current step, and influence the final output. We use the hidden state of the LSTM unit as the output of the LSTM unit. Therefore, the input sequence of the second LSTM layer, is the hidden state sequence of the first LSTM layer $[\mathbf{h}_t, \ \mathbf{h}_{t+1}, \ ... \ \mathbf{h}_{t+S-1}]$. Since we are interested in what the brain is directing during the whole segment period, the extracted features when the LSTM has observed the entire samples of the sliding window are used for further analysis. Only the output of the last time step LSTM, $\mathbf{h}'_{t_S-1}$, is fed into the next fully connected layer as shown in the final stage of Figure 2. The temporal feature representation $\mathbf{h}'_{t+S-1}$ of the segment $\mathbf{S}_j$ is:

$$\text{CasRNN: } \mathbf{h}'_{t+S-1} = R_{lstm}(\mathbf{F}_j), \ \ \mathbf{h}'_{t+S-1} \in \mathbb{R}^d,$$

where d is the size of the hidden state of an LSTM unit. On top of the fully connected layer is the final softmax layer yielding final probability prediction of each class:

$$\text{FC-softmax: } \mathbf{P}_j = S_m(\mathbf{h}'_{t+S-1}), \ \ \mathbf{P}_j \in \mathbb{R}^K,$$

where the framework aims to classify K categories. We induce dropout operations as a form of regularization after the fully connected layers in both the 2D-CNN stage and the final classification stage.

Overall, the framework convert and split the EEG recording streams to segments of 2D data meshes, and classify each segment to one of the K categories. Each segment $\mathbf{S}_j$ contains S EEG data recordings, which have been converted to S 2D meshes $[\mathbf{m}_t, \ \mathbf{m}_{t+1}, \ ... \ \mathbf{m}_{t+S-1}]$. A 2D-CNN is applied mesh-wise in a segment to extract spatial features $[\mathbf{f}_t, \ \mathbf{f}_t \ ... \ \mathbf{f}_{t+S-1}]$, and a RNN is consequently applied to extract the temporal features $\mathbf{h}'_{t+S-1}$ across the data meshes. Softmax classifier finally computes the classification probabilities over K brain intentions for each individual segment. The proposed model can be depicted as: $I(S \times n)$-$T(S \times h \times w)$-$C(S \times h \times w \times 32)$-$C(S \times h \times w \times 64)$-

$C(S \times h \times w \times 128)$-$FC(S \times l)$-$R(S \times d)$-$R(S \times d)$-$FC(l)$-$Sm(K)$, where $I(S \times n)$ denotes the input EEG data window of S 1D data recordings with n elements each, $T(S \times h \times w)$ means transforming input EEG recordings to data meshes of size $h \times w$, $C(S \times h \times w \times m)$ denotes a mesh-wise covolutional layer with m feature maps, $FC(l)$ is a fully connected layer with l neurons, $R(S \times d)$ is a RNN layer with the hidden state size of d for each LSTM cell and totally S LTSM cells, and $Sm(K)$ denotes the softmax layer predicting K classes.
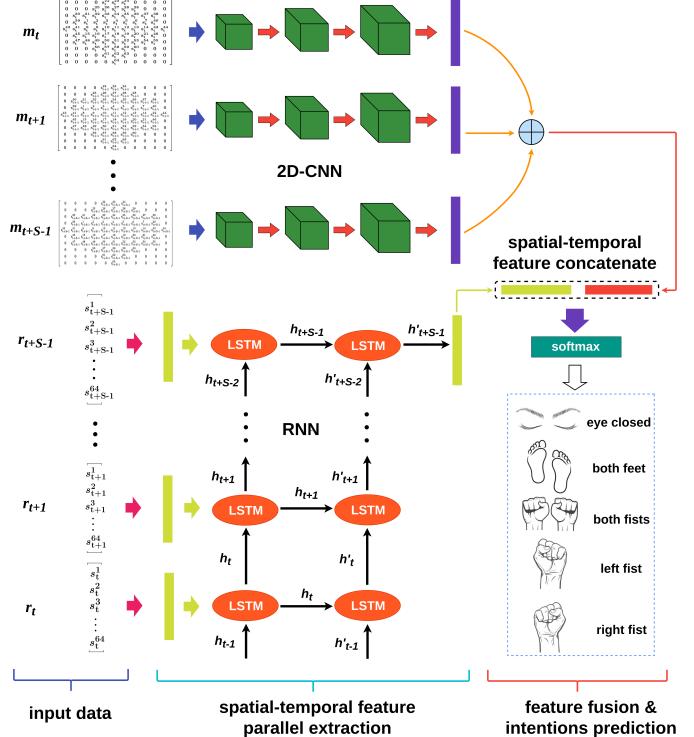

Figure 3: **Parallel recurrent convolutional neural network architecture.** The concatenate operation in the final spatio-temporal fusion part is used for an example.

Parallel Convolutional Recurrent Network

The structure of the parallel convolutional recurrent network is illustrated in Figure 3. It also contains two parts, CNN and RNN, for spatial and temporal feature extraction respectively. However, different from the cascade model, the parallel model extracts the spatial and temporal features of EEG signals in parallel and fuses the extracted features at last for final intention recognition. Particularly, the RNN part of the parallel model receives the data from the same segments to that feed the corresponding CNN part. While due to the RNN part responsible for the temporal feature extraction, the raw EEG data vectors are not converted to 2D-mesh hierarchies. The jth input windowed segment to the RNN part is:

$$\mathbf{R}_j = [\mathbf{r}_t, \ \mathbf{r}_{t+1} \ ... \ \mathbf{r}_{t+S-1}],$$

where r_t is the data vector at time step t, and S denotes the window size. The RNN part of the parallel model also

has two LSTM layers, each containing the same number of LSTM units with that of the cascade model due to the same window size we use. The hidden state of the last time step in one segment is used for further analysis as well:

$$\mathbf{h}'_{t+S-1} = R_{lstm}(\mathbf{R}_j), \ \ \mathbf{h}'_{t+S-1} \in \mathbb{R}^v,$$

where v is the hidden state size of the LSTM unit. A fully connected layer is applied both before and after LSTM layers to enhance the temporal information representation capabilities. Thus the final temporal features from the parallel RNN part is denoted as:

$$\text{ParaRNN:} \mathbf{O}_j = \text{FC}(\mathbf{h}'_{t+S-1}), \ \ \mathbf{O}_j \in \mathbb{R}^l,$$

where l is the size of the finally fully connected layer of the parallel RNN part. The parallel CNN part which is responsible for extracting spatial features, receives the segment of 2D data meshes $\mathbf{S}_j$ as input, and applies mesh-wise convolutional operations as the CNN part of the cascade model does. The CNN structure of the parallel model is the same with that of the cascade model as well. To be comparable to the temporal features in terms of size, the extracted spatial features $\mathbf{f}_k$ at each time step in one segment are added up to a single feature vector $\mathbf{L}_j$:

$$\text{ParaCNN:} \ \mathbf{L}_j = \sum_{k=t}^{t+S-1} \mathbf{f}_k \ \ (\mathbf{L}_j, \ \mathbf{f}_k \in \mathbb{R}^l),$$

where l is the size of the fully connected layer in the CNN part, which is the same with that of the RNN part.

The concurrently extracted spatial and temporal features are fused to a joint spatio-temporal feature vector. Various fusion approaches are developed, and the detailed results are shown in the following experiment section. A softmax layer receives the fused spatio-temporal features to finally predict the human intentions:

$$\text{softmax:} \ \mathbf{P}_j = S_m([\mathbf{L}_j, \ \mathbf{O}_j]), \ \mathbf{P}_j \in \mathbb{R}^K.$$

According to the previous denotation rules, the overall structure of the parallel model is: $I(S{\times}n)\text{-}T(S{\times}h{\times}w)||I(S{\times}n)\text{-}C(S{\times}h{\times}w{\times}32)||FC(S{\times}l)\text{-}C(S{\times}h{\times}w{\times}64)||R(S{\times}v)\text{-}C(S{\times}h{\times}w{\times}128)||R(S{\times}v)\text{-}FC\&add(l)||FC(l)\text{-}\text{Fusion-}S(K)$, where $||$ denotes data flows in parallel.

In the 2D-CNN part of both the cascade and parallel models, convolutional layers are not followed by a pooling operation. Although in CNN architecture a convolutional operation is often coupled with a pooling operation, it is not mandated. The pooling operation is usually used for reducing data dimensions at the cost of missing some information, However, in this EEG data analysis problem, the data dimension is much smaller than that used in computer vision research, so in order to keep all information, we concatenate three CNN layers directly without pooling operations.

Experiments and Result Summary

We focus on the MI-EEG Dataset of cross-subject, multi-class scenario on PhysioNet (Goldberger et al. 2000) to evaluate the developed cascade and parallel convolutional recurrent network frameworks for movement intention recognition. We compare our model against those previous reported to show our superior performance. Meanwhile, we also investigate the influence of the spatial and temporal information by comparison experiments. At last, different variants of both cascade and parallel models are systematically studied.

Dataset and Model Implementation

The movement intention EEG data is collected using BCI2000 instrumentation (Schalk et al. 2004) with 64 electrode channels and 160Hz sampling rate. To the best of our knowledge, this dataset is so far the largest EEG-based human movement intention dataset with 109 subjects. But in the data preprocessing stage, we found that the recordings of the #89 subject were severely damaged, so this participant's record was removed from further analysis. We used the EEG data from 108 subjects to build the cross-subject dataset. The dataset contains five brain activities with eye closed (baseline), imagining opening and closing both feet, imagining opening and closing both fists, imagining opening and closing left fist and imagining opening and closing right fist.

For time series model (proposed cascade and parallel models, baseline 3D-CNN and RNN) evaluation, we use a sliding window of fixed length to segment the data. The length of the window is 62.5 ms with a step size of 31.25 ms, so the window size S is set 10. After segmenting, the number of instances is 624,259 from 108 subjects. During the experimental process, it is observed that larger or smaller length of the window size decreases the model performance. This is possibly due to the periodic length of the brain activity. According to the EEG data recording process of the evaluation dataset, the 2D data meshes are transformed with the size of 10×11 as shown in Figure 1. We randomly select 75% instances to train the model and 25% instances to validate the trained model.

All neural networks were implemented with the Tensor-Flow framework and trained on a Nvidia Titan X pascal GPU from scratch in a fully-supervised manner. The stochastic gradient descent with Adam update rule (Kingma and Ba 2015) is used to minimize the cross-entropy loss function. The network parameters are optimized with a learning rate of 10^{-4}. The keep probability of the dropout operation is 0.5. The hidden states of the LSTM cell for cascade model d and parallel model v are 64 and 16 respectively. All fully connected layers have the same size of 1024.

Comparison Models

State-of-the-arts　We will give a brief introduction of the compared state-of-the-art models. All the models are based on the same dataset with our work.

- (Major and Conrad 2017) researches independent component analysis (ICA) to reduce noises and feed the result to a neural network for final prediction on intra-subject binary MI-EEG classification;

- (Shenoy, Vinod, and Guan 2015) uses shrinkage regularized filter bank common spatial patterns (SR-FBCSP) for intra-subject binary MI-EEG classification;

- (Pinheiro et al. 2016) focuses on one-against-all EEG classification using SVM, nearest neighbour and C4.5 algorithms;

- (Kim et al. 2016) extracts EEG features with strong uncorrelating transform complex common spatial patterns (SUTCCSP) algorithm, and make final predictions with random forest classifier for the cross-subject binary classification;

- (Zhang et al. 2017) uses autoencoder for EEG feature extraction and XGboost for final classification on five-category, cross-subject motor imagery scenario;

- (Bashivan et al. 2016) extracts the frequency features of EEG data, and converts the extracted features to images to feed into deep neural networks. We reproduce their method on the same MI-EEG dataset with this work using their open access code.

Baseline models Apart from a set of state-of-the-arts, we also compare our model with the variants of CNN- and RNN-based models. We use 1D-CNN (without spatial or temporal information), 2D-CNN (only with spatial information) and 3D-CNN (with both spatial and temporal information) models for comparison and investigating the influence of spatial and temporal information on brain intention recognition. The 1D-CNN model just uses the raw EEG vectors as input.The 2D-CNN model is fed with the data meshes transformed by equation (1), but without sliding window segmentation. The 3D-CNN model uses the same input data with that fed into the cascade model. Each of the three CNN models has three convolutional layers without subsampling layers, one fully connected layer with 1024 nurons and one softmax output layer. The kernel size of the models are 3, 3×3 and $3 \times 3 \times 3$ for 1D, 2D and 3D, respectively, and the stride keeps constant of 1. The number of feature maps of each CNN layer are 32, 64 and 128 for all baseline models. For comparison purpose, we keep all the hyper-parameters of the baseline CNN models the same with the CNN part of our proposed method, except for their own particular hyper-parameters.

To make a fair comparison with both cascade model and parallel model, we keep the RNN baseline models with two LSTM layers between two fully connected layers and choose 64 and 16 as the hidden state size of LSTM cells respectively.

Experimental Results

In this section, we will present the overall performance of our proposed model and the comparison experimental results. The influence of spatial and temporal information will also be systematically analyzed.

Overall Performance The overall performance of our proposed models and the comparison models are summarized in Table 1. It is observed that both our cascade and parallel models achieve high accuracy near 98.3%, consistently outperforming the state-of-the-art methods and the baseline models. Even though some work is focused on simple scenarios, such as instra-subject or binary classification, our method surpasses their method significantly. Furthermore, our 3D-CNN baseline model also achieves competitive results to the state-of-the-art work. This implies that it is crucial to use the

Table 1: Comparison with the state-of-the-art methods and baseline methods. All the methods are based on the same dataset. RNN(64) and RNN(16) denote RNN models with hidden state size of 64 and 16 respectively.

Method	Multi-class	Validation	Acc
(Major and Conrad 2017)	Binary	Intra-Sub	0.72
(Shenoy, Vinod, and Guan 2015)	Binary	Intra-Sub	0.8206
(Pinheiro et al. 2016)	Bianry	Cross-Sub(10)	0.8505
(Kim et al. 2016)	Binary	Cross-Sub(105)	0.805
(Zhang et al. 2017)	Multi(5)	Cross-Sub(20)	0.794
(Bashivan et al. 2016)[2]	Multi(5)	Cross-Sub(108)	0.6731
1D-CNN	Multi(5)	Cross-Sub(108)	0.8622
2D-CNN	Multi(5)	Cross-Sub(108)	0.8841
3D-CNN	Multi(5)	Cross-Sub(108)	0.9238
RNN(64)	Multi(5)	Cross-Sub(108)	0.8493
RNN(16)	Multi(5)	Cross-Sub(108)	0.7468
Cascade model	Multi(5)	Cross-Sub(108)	**0.9824**
Parallel model	Multi(5)	Cross-Sub(108)	**0.9828**

spatial and temporal information to boost EEG-based intention recognition and analysis. Bashivan et al. also proposed to use CNN and RNN for EEG signal analysis (Bashivan et al. 2016). However, they used complex preprocessing by extracting the frequency features of EEG signals and converting to 2D images instead of using the raw signal data. To make a comparison, we reproduced their method on our dataset using their open access code on github, and the results are also shown in Table 1. Our approach outperforms Bashavan's models by some 30%. This is probably because the frequency feature extraction needs a large continuous sampling period, while the motor imagery tasks are periodic short duration brain activities. So extracting the frequency features may damage the temporal information. While in their report, they use the method for brain working load prediction, which is a relatively long time static activity, so their method gives acceptable results in their experiments. In our study, a larger sliding window also decreases the model performance dramatically. Our method has the flexibility to adapt different kinds of EEG-based intention recognition by varying sliding window size over an unrestricted range. Compared with previous studies, our model requires less preprocess on raw data making it more suitable for real-time applications, such as BCI.

Impact of Temporal and Spatial Information To investigate the influence of spatial and temporal information on intention recognition and compare the developed models with alternative neural networks, we build up three CNN baseline models as as depicted above, and their performance is summarized in Table 1. When comparing the results of 1D-CNN

---

[2]We reproduce the approach on our dataset using the open access code on github https://github.com/pbashivan/EEGLearn

and 2D-CNN baseline models, the 2D-CNN outperforms the 1D-CNN on overall accuracy. This indicates that the spatial information helps the brain intention recognition problem, and extracting the correlations between physically adjacent sensory signals is more effective than simple chain-like signals. In addition, the brain intentions can also be defined as a series periodic brain motions. The 3D convolution is widely used to extract local spatial-temporal information (Ji et al. 2013). In Table 1, the 3D-CNN model shows consistently improved performance compared with both the 1D-CNN and 2D-CNN baseline models. Moreover, the cascade model uses RNN to extract the global temporal dynamics after spatial feature extraction dramatically promotes the performance, and surpasses the 3D-CNN model. However, when only considering the temporal information with RNN, the recognition accuracy descends to around 80%, indicating the combination of spatio-temporal features is crucial to successfully analyzing EEG signals. When extracting the spatial and temporal features in parallel, we observe similar results. The parallel convolutional recurrent network consistently outperforms the models separately considering spatial information (2D-CNN model) or temporal information (RNN(16) model). Both the cascade model and the parallel model perform excellent and similar recognition accuracy of around 98.3%. These comparison results demonstrate that the spatio-temporal features obviously enhance the intention recognition tasks from EEG signals, and both feature fusion approaches have the powerful capabilities to depict the spatio-temporal representations effectively.

Table 2: Comparison of different structures of cascade convolutional recurrent network model

Cascade structure	Acc
1-layer CNN+FC+2-layer RNN+FC	0.9310
2-layer CNN+FC+2-layer RNN+FC	0.9712
3-layer CNN+FC+2-layer RNN+FC	**0.9831**
3-layer CNN+2-layer RNN+FC	0.9217
3-layer CNN+FC+2-layer RNN	0.9801
3-layer CNN+FC+1-layer RNN+FC	0.9813

Cascade model variants   Since it is impossible to exhaustively investigate the neural network architectures, here we study the effects of the key components of the cascade model. In the experiment process, it is observed that the cascade architecture produces best performance when the hidden state size of LSTM cells is 64. This is probably because it is hard to well trained the network with larger size and smaller hidden state size has limited representation capabilities. Hence the following cascade model variants are all based on LSTM hidden state size of 64. The results are summarized in Table 2. It is shown that more CNN or RNN layers results better accuracy. However this performance improvement is at the cost of computational resources, thus we choose three CNN layers and two RNN layers by trade-off between performance and resources. Fully connected layers are also critical components, especially the layer linking the CNN part and RNN part. This illustrates reconstructing the outputs before feeding another type of neural network facilitates following

feature extraction. However, the final fully connected layer does not contributes much. This might be due to the spatio-temporal features after RNN layers are robust enough for final prediction.

Parallel model variants   Table 3 shows the comparison of different structures of the parallel model. The hidden state size of the LSTM cells in RNN components is set as 16 empirically. It is also observed that the performance can be improved by more CNN layers or RNN layers. The most different character of the parallel model and the cascade model is the fusion approach of spatial and temporal features. Tn the parallel model the data flows the CNN and RNN concurrently. Hence, there are diverse methods to fuse the parallel data. Here two basic fusion approaches, (concatenate and add) as well as two improved fusion approaches (concatenate joint fully connected layer and concatenate joint pointwise convolutional operation (Chollet 2017)) are studied. It is interesting to find that the basic fusion methods perform better results with accuracy higher than 98%. Complex or advanced neural network needs careful training and parameter tuning to achieve better performance, thus it is redundant to add more operations when basic approaches are capable to achieve satisfactory results.

Table 3: Comparison of different structures of parallel convolutional recurrent network model. Cat is short for concatenate, and Conv is short for point-wise convolutional operation.

Parallel structure	Fusion method	Acc
1-layer CNN+FC FC+2-layer RNN+FC	Concatenate	0.9487
2-layer CNN+FC FC+2-layer RNN+FC	Concatenate	0.9727
3-layer CNN+FC FC+2-layer RNN+FC	**Concatenate**	**0.9828**
3-layer CNN+FC FC+1-layer RNN+FC	Concatenate	0.9821
3-layer CNN+FC FC+2-layer RNN+FC	Add	0.9813
3-layer CNN+FC FC+2-layer RNN+FC	Cat+FC	0.9696
3-layer CNN+FC FC+2-layer RNN+FC	Cat+Conv	0.9666

Conclusions

In this paper, we propose the use of spatio-temporal representations to enhance EEG-based intention recognition in a more practical scenario of cross-subject and multi-class, and develop two unified end-to-end trainable deep learning models for movement intention recognition. A large-scale dataset of 108 participants on five categories is used to evaluate the proposed models. The experimental results show that both the cascade and parallel architectures could achieve very competitive accuracy around 98.3%, considerably superior to the state-of-the-art methods. Moreover, we build a set of baseline models to investigate the influence of both spatial and temporal information on brain intention recognition, and

study the key variant architectures of the proposed frameworks. It is demonstrated that the performance improvement is largely due to the spatial and temporal information used in intention recognition process. Our model is also robust where sensor data is missing.

References

[Ahn et al. 2014] Ahn, M.; Lee, M.; Choi, J.; and Jun, S. C. 2014. A review of brain-computer interface games and an opinion survey from researchers, developers and users. *Sensors* 14(8):14601–14633.

[Bashivan et al. 2016] Bashivan, P.; Rish, I.; Yeasin, M.; and Codella, N. 2016. Learning representations from eeg with deep recurrent-convolutional neural networks. In *International conference on learning representations*.

[Bright et al. 2016] Bright, D.; Nair, A.; Salvekar, D.; and Bhisikar, S. 2016. Eeg-based brain controlled prosthetic arm. In *Advances in Signal Processing (CASP), Conference on*, 479–483. IEEE.

[Chollet 2017] Chollet, F. 2017. Xception: Deep learning with depthwise separable convolutions. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Goldberger et al. 2000] Goldberger, A. L.; Amaral, L. A.; Glass, L.; Hausdorff, J. M.; Ivanov, P. C.; Mark, R. G.; Mietus, J. E.; Moody, G. B.; Peng, C.-K.; and Stanley, H. E. 2000. Physiobank, physiotoolkit, and physionet. *Circulation* 101(23):e215–e220.

[Heydari and Shahbakhti 2015] Heydari, E., and Shahbakhti, M. 2015. Adaptive wavelet technique for eeg de-noising. In *Biomedical Engineering International Conference (BMEiCON), 2015 8th*, 1–4. IEEE.

[Ji et al. 2013] Ji, S.; Xu, W.; Yang, M.; and Yu, K. 2013. 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence* 35(1):221–231.

[Kim et al. 2016] Kim, Y.; Ryu, J.; Kim, K. K.; Took, C. C.; Mandic, D. P.; and Park, C. 2016. Motor imagery classification using mu and beta rhythms of eeg with strong uncorrelating transform based complex common spatial patterns. *Computational intelligence and neuroscience* 2016:1.

[Kingma and Ba 2015] Kingma, D., and Ba, J. 2015. Adam: A method for stochastic optimization. In *3rd International Conference for Learning Representations*.

[LeCun, Bengio, and Hinton 2015] LeCun, Y.; Bengio, Y.; and Hinton, G. 2015. Deep learning. *Nature* 521(7553):436–444.

[Major and Conrad 2017] Major, T. C., and Conrad, J. M. 2017. The effects of pre-filtering and individualizing components for electroencephalography neural network classification. In *SoutheastCon, 2017*, 1–6. IEEE.

[Pinheiro et al. 2016] Pinheiro, O. R.; Alves, L. R.; Romero, M.; and de Souza, J. R. 2016. Wheelchair simulator game for training people with severe disabilities. In *Technology and Innovation in Sports, Health and Wellbeing (TISHW), International Conference on*, 1–8. IEEE.

[Qiu et al. 2017] Qiu, S.; Li, Z.; He, W.; Zhang, L.; Yang, C.; and Su, C.-Y. 2017. Brain–machine interface and visual compressive sensing-based teleoperation control of an exoskeleton robot. *IEEE Transactions on Fuzzy Systems* 25(1):58–69.

[Schalk et al. 2004] Schalk, G.; McFarland, D. J.; Hinterberger, T.; Birbaumer, N.; and Wolpaw, J. R. 2004. Bci2000: a general-purpose brain-computer interface (bci) system. *IEEE Transactions on biomedical engineering* 51(6):1034–1043.

[Shenoy, Vinod, and Guan 2015] Shenoy, H. V.; Vinod, A. P.; and Guan, C. 2015. Shrinkage estimator based regularization for eeg motor imagery classification. In *Information, Communications and Signal Processing (ICICS), 2015 10th International Conference on*, 1–5. IEEE.

[Sun and Zhou 2014] Sun, S., and Zhou, J. 2014. A review of adaptive feature extraction and classification methods for eeg-based brain-computer interfaces. In *Neural Networks (IJCNN), 2014 International Joint Conference on*, 1746–1753. IEEE.

[Tabar and Halici 2016] Tabar, Y. R., and Halici, U. 2016. A novel deep learning approach for classification of eeg motor imagery signals. *Journal of neural engineering* 14(1):016003.

[Wang et al. 2014] Wang, H.; Li, Y.; Long, J.; Yu, T.; and Gu, Z. 2014. An asynchronous wheelchair control by hybrid eeg–eog brain–computer interface. *Cognitive Neurodynamics* 8(5):399.

[Yin et al. 2017] Yin, Z.; Wang, Y.; Liu, L.; Zhang, W.; and Zhang, J. 2017. Cross-subject eeg feature selection for emotion recognition using transfer recursive feature elimination. *Frontiers in neurorobotics* 11.

[Zhang et al. 2017] Zhang, X.; Yao, L.; Zhang, D.; Wang, X.; Sheng, Q.; and Gu, T. 2017. Multi-person brain activity recognition via comprehensive eeg signal analysis. In *14th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*.