Jumping Knowledge Based Spatial-Temporal Graph Convolutional Networks for Automatic Sleep Stage Classification

Xiaopeng Ji, Yan Li, and Peng Wen

***Abstract*—A novel jumping knowledge spatial-temporal graph convolutional network (JK-STGCN) is proposed in this paper to classify sleep stages. Based on this method, different types of multi-channel bio-signals, including electroencephalography (EEG), electromyogram (EMG), electrooculogram (EOG), and electrocardiogram (ECG) are utilized to classify sleep stages, after extracting features by a standard convolutional neural network (CNN) named FeatureNet. Intrinsic connections among different bio-signal channels from the identical epoch and neighboring epochs can be obtained through two adaptive adjacency matrices learning methods. A jumping knowledge spatial-temporal graph convolution module helps the JK-STGCN model to extract spatial features from the graph convolutions efficiently and temporal features are extracted from its common standard convolutions to learn the transition rules among sleep stages. Experimental results on the ISRUC-S3 dataset showed that the overall accuracy achieved 0.831 and the F1-score and Cohen kappa reached 0.814 and 0.782, respectively, which are the competitive classification performance with the state-of-the-art baselines. Further experiments on the ISRUC-S3 dataset are also conducted to evaluate the execution efficiency of the JK-STGCN model. The training time on 10 subjects is 2621s and the testing time on 50 subjects is 6.8s, which indicates its highest calculation speed compared with the existing high-performance graph convolutional networks and U-Net architecture algorithms. Experimental results on the ISRUC-S1 dataset also demonstrate its generality, whose accuracy, F1-score, and Cohen kappa achieve 0.820, 0.798, and 0.767 respectively.**

***Index Terms*—Deep learning, graph convolutional networks, sleep stage classification.**

I. Introduction

SLEEP plays an important role in human life. Sleep disorders, like insomnias, apnea, and circadian rhythm sleep disorders affect our daily life psychologically and physically. Disturbed sleep patterns lead to sleeplessness at night, which affects our mental status and results in poor mental functions [1]. Poor sleep quality also raises risks of cardiovascular diseases and strokes [2]. Bio-signals, including electroencephalography (EEG), electromyogram (EMG), electrooculogram (EOG), and electrocardiogram (ECG), collected through electrodes placed in different locations in humans, such as the brain, chest, and face, called polysomnograms (PSGs), are powerful tools to help experts and researchers to diagnose sleep disorders [3]. These PSGs are segmented into epochs, which are classified into sleep stages by experienced experts according to the sleep staging criteria such as the Rechtschaffen and Kales sleep staging rules (R&K rules) [4] and American Academy of Sleep Medicine (AASM) standards [5]. Although the PSG-based sleep stage classification is a powerful tool for experts to analyze sleep quality and diagnose sleep disorders, this visual inspection-based manual sleep scoring is a tedious and time-consuming task for trained specialists [6].

To identify sleep stages efficiently, many automatic sleep stage classification methods have been reported. Traditional machine learning methods have given reasonably high sleep stage classification performance in past decades. Inputs of traditional machine learning algorithms are usually extracted from the time-domain [7], [8], frequency-domain [9], [10], or time-frequency domain [11], [12], which requires a lot of prior knowledge [8], [13]. For example, a preprocessing phase is required to eliminate cognitive noise and interference among channels. Often principal component analysis is a typical data reduction technique to seek undesired linear correlation among variables [14]. Due to this limitation, the performance of those algorithms heavily depends on feature engineering and feature selections. Compared to traditional machine learning algorithms, deep learning methods can extract higher-level features from original inputs and output classification results directly. Convolutional neural networks (CNNs) have demonstrated their advanced performance in sleep stage classification [15], [16], while other popular deep learning algorithms like recurrent neural networks (RNNs) [17], [18] and deep belief networks [19] have achieved reliable results as well.

CNNs have the capacity of extracting high-level features from raw data, which allows researchers to input raw data directly instead of hand-crafted features. However, these methods require Euclidean inputs and ignore connections among

Manuscript received January 8, 2022; revised May 2, 2022; accepted May 10, 2022. Date of publication May 18, 2022; date of current version June 3, 2022. The work of Xiaopeng Ji was supported by the University of Southern Queensland International Tuition Fee Scholarship. *(Corresponding author: Xiaopeng Ji.)*

This work involved human subjects or animals in its research. The authors confirm that all human/animal subject research procedures and protocols are exempt from review board approval.

Xiaopeng Ji and Yan Li are with the School of Mathematics, Physics and Computing, University of Southern Queensland, Toowoomba, QLD 4350, Australia (e-mail: xiaopeng.ji@usq.edu.au; yan.li@usq.edu.au).

Peng Wen is with the School of Mechanical and Electrical Engineering, University of Southern Queensland, Toowoomba, QLD 4350, Australia (e-mail: paul.wen@usq.edu.au).

Digital Object Identifier 10.1109/TNSRE.2022.3176004

brain regions. Considering the limited understanding of the intrinsic relationship among different channels in different sleep stages, graph-based methods are more advantageous in representing brain connections and their activities. Compared to CNNs, graph convolution networks (GCNs) [20], [21] have the capacity to extract spatial features efficiently on topological data structures, which would provide a potential way to explore the relationship among multiple bio-signal channels during the sleep stage classification.

According to the R&K rules and AASM standards, the transition pattern between neighboring sleep stages is also an essential factor to be considered when sleep stages are identified. However, most of those deep learning algorithms are focused on model development while little attention is paid to the transition mechanism during the sleep process.

To tackle the above challenges, a jumping knowledge spatial-temporal graph convolutional network (JK-STGCN) is proposed to identify sleep stages automatically in this study. With two adaptive graph learning layers and a jumping knowledge graph convolution structure, the JK-STGCN not only learns functional connections among brain regions at each epoch and aggregates temporal functional connections from neighboring epochs but also extracts spatial and temporal features from inputs. The main contributions of this paper are summarized as follows:

- A novel adaptive graph learning method is designed to aggregate the temporal functional relationship among different bio-signal channels from neighboring epochs for the localized spatial graph convolution.
- A novel jumping knowledge spatial-temporal graph convolutional module is proposed to capture the localized spatial correlations and temporal features directly.
- Sleep stage classification experiments are conducted on the ISRUC-S3 and ISRUC-S1 (https://sleeptight.isr.uc.pt/) to test the performance of the JK-STGCN model on healthy subjects and sleep-disordered cases. The experimental results demonstrate that the proposed model achieves the competitive overall performance compared to existing baselines. The experimental results of sleep stage classification on healthy-unhealthy mixed cases indicate that the JK-STGCN model achieves the best performance to classify sleep stages of both healthy and unhealthy cases when the unhealthy samples take around 60% of occupancy.
- Ablation experiments are also carried out on the ISRUC-S3 dataset to explore the effects of different modules on the sleep stage classification performance and the experimental results show that the JK-STGCN model has the best performance when there is a jumping knowledge spatial-temporal graph convolutional module with no attention mechanism.

II. Related Work

A. Sleep Stage Classification

Traditional machine learning classification algorithms, such as support vector machines [22], [23] and random forest [24], [25], have been used for decades in bio-signal analysis, and many studies have reported their high performance in sleep scoring. However, these algorithms require prior knowledge about signal characteristics and feature engineering. That means that the performance may be severely limited by researchers' understanding of data. Due to the fact that deep learning has brought significant breakthroughs in many research areas, such as image processing [26], [27] and natural language processing [28], more and more researchers apply deep learning to sleep stage classification [29], [30].

Unlike traditional machine learning methods, deep learning algorithms, such as CNNs and RNNs [31], have the capacity to extract abstract and high-level features from raw data directly, which allows researchers to use the raw data instead of hand-picked features. Sors *et al.* [32] proposed a 14-layer CNN to extract features from the original single EEG channel inputs. A two-step training CNN model named DeepSleepNet [15] extracts time-invariant features and bidirectional-long short-term memory to learn transition rules among EEG segments. The combination of a long short-term memory unit and a deep belief network [19] has also been applied to identify sleep stages. The U-Net is a very complex architecture with a multi-scale extraction module, which also demonstrates its performance in sleep stage classification [33], [34].

Although these algorithms can extract spatial features and temporal features manually or automatically, they still failed to explore the functional connections among different brain regions during sleep stage classification.

B. Graph Convolutional Networks

Recently, visibility graphs have been utilized in the bio-signal analysis [35] and the sleep stage classification area [36]. Experimental results indicate that graph features make many contributions to improving the classification accuracies. Combining the graph construction and convolutional operation, also known as GCN, has become popular in many fields, like calculating molecular fingerprints [37], text classification [38], neural machine translation [39], etc. Motivated by its success, many researchers have turned to these non-Euclidean input neural networks in the bio-signal processing area, including motor imagery recognition [40], emotion recognition [41], [42], and epileptic seizure detection [43]. However, for sleep scoring, only a few GCN models have been reported. GraphSleepNet [44] is a spatial-temporal graph convolutional network with a spatial attention layer and a temporal attention layer [45], which inputs differential entropy features [46] extracted from multi-channel bio-signals into a learnable adjacency matrix to calculate the graph convolution and classify sleep stages. Jia *et al.* [47] designed a multi-view spatial-temporal graph convolutional network (MSTGCN) and applied the spatial-temporal graph neural network with domain generalization for sleep stage classification. Although the existing GCN models were claimed to be able to solve the problem of obtaining dynamical functional connections among different brain regions and achieved some higher classification accuracies than traditional methods, they fail to aggregate temporal information from neighboring epochs.

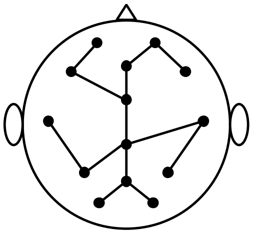

Fig. 1. An example of sleep graph mapped from electrodes at time *t*.

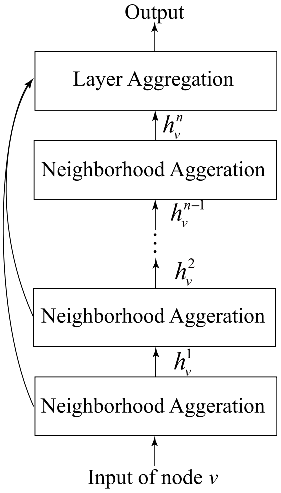

Fig. 2. The structure of a n-layer jumping knowledge network.

III. Preliminaries

In this study, a sleep graph is defined as an undirected graph $\mathcal{G} = (V, E, A)$, where V denotes the set of nodes with the number of $|V| = N$; E denotes the set of edges connecting these nodes; $A \in \mathbb{R}^{N \times N}$ denotes an adjacency matrix of $\mathcal{G}$. At epoch t, attached electrodes will be mapped to a graph as shown in Fig. 1. The connections (edges) between nodes are controlled by a learnable adjacency matrix A.

The raw signal sequences containing L samples are defined as $S = (s_1, s_2, \ldots, s_L) \in \mathbb{R}^{L \times N \times T_s}$, where N denotes the number of channels, T_s denotes sample data points. For each sleep epoch $s_i \in S(i \in \{1, 2, \ldots, L\})$, features are extracted from a CNN named FeatureNet [47] and a N-channel feature matrix of the i-th epoch is defined as $X_i = (x_1^i, x_2^i, \ldots, x_N^i)^T \in \mathbb{R}^{N \times F}$, where $x_n^i \in \mathbb{R}^F(n \in \{1, 2, \ldots, N\})$ denotes features extracted from channel n at epoch i.

The jumping knowledge spatial-temporal graph convolution module is a combination of spatial graph convolution and temporal convolution based on a JK-Net structure [48]. It aggregates both neighborhoods at each independent layer and neighborhoods from previous layers, which increases the size of the influence distribution. As Fig. 2 shows, for each independent node in a graph, the last layer can select from all of those intermediate representations to adapt an effective neighborhood size for each node as needed, and this can lead to a desired adaptivity.

IV. Jumping Knowledge Spatial-Temporal GCN

Fig. 3 illustrates the architecture of the proposed model. There are three key components in this model: 1) Two adaptive graph learning layers are designed to construct adjacency matrixes for the two graph convolutional layers. 2) Based on the JK-Net [48], graph convolutional layers with residual connections are utilized to capture the localized spatial features from neighboring nodes at the same epoch and to aggregate information from different layers. 3) A jumping knowledge spatial-temporal graph convolution module is designed to extract both spatial features and temporal features.

A. Adaptive Graph Learning

Motivated by their high performance of adaptive graph learning methods in the studies of [44], [47], two different graph learning layers are utilized in this study for the localized spatial graph convolution operation.

1) Function-Based Adaptive Graph Learning: As proposed in [44], this connection A_{mn} between node n and node m in an adaptive graph is defined by a non-negative function:

$$\begin{aligned} A_{mn} &= g(x_m, x_n) \\ &= \frac{\exp(\mathrm{ReLU}(\boldsymbol{\omega}^T\,|x_m - x_n|))}{\sum_{n=1}^{N} \exp(\mathrm{ReLU}(\boldsymbol{\omega}^T\,|x_m - x_n|))} \end{aligned} \tag{1}$$

where x_m and x_n are the nodes of the adaptive graph, $\boldsymbol{\omega} = (\omega_1, \omega_2, \ldots, \omega_F)^T \in \mathbb{R}^{F \times 1}$ is a learnable parameter set. The activation function ReLU guarantees that A_{mn} is non-negative. The softmax operation normalizes each row of A. Weight vector $\boldsymbol{\omega}$ is updated by minimizing the following loss function,

$$\mathcal{L}_{\text{graph\_learning}} = \sum_{m,n=1}^{N} \|x_m - x_n\|_2^2\, A_{mn} + \lambda\, \|A_F\|^2 \tag{2}$$

where $\lambda \geq 0$ is a regularization parameter.

2) Temporal-Information-Based Graph Learning: A function-based adaptive graph learning method can learn the intrinsic connections among different bio-signal channels at one epoch. However, it fails to aggregate functional connections from neighboring epochs, which means that, for each node, the temporal influences from its neighboring nodes of previous epochs and coming epochs are ignored. A temporal-information-based graph learning method considers the intrinsic connections from both temporal and spatial view. A $2d + 1$ time steps temporal-information-based adaptive graph is defined as

$$A_T = avg(X \cdot W) \tag{3}$$

where $X = (x_{t-d}, \ldots, x_t, \ldots, x_{t+d}) \in \mathbb{R}^{(2d+1) \times N \times F}$ is a feature set. $W = (w_{t-d}, \ldots, w_t, \ldots, w_{t+d}) \in \mathbb{R}^{(2d+1) \times F \times N}$ is a learnable parameter set. The avg function calculates the mean values of $2d + 1$ adjacency matrixes from time step $t - d$ to $t + d$, which helps to aggregate connections from $2d + 1$ neighboring epochs. The loss of this temporal-information-based graph learning will be considered during calculating the overall loss which is defined as in equation (4):

$$\mathcal{L}_{\text{loss}} = \mathcal{L}_{\text{cross\_entropy}} + \mathcal{L}_{\text{graph\_learning}} + \beta\, \|A_T\|^2 \tag{4}$$

where β denotes the strength of L2 regularization for temporal-information-based adjacency matrix A_T, and $\mathcal{L}_{\text{graph\_learning}}$ is the loss of the function-based adaptive graph learning as

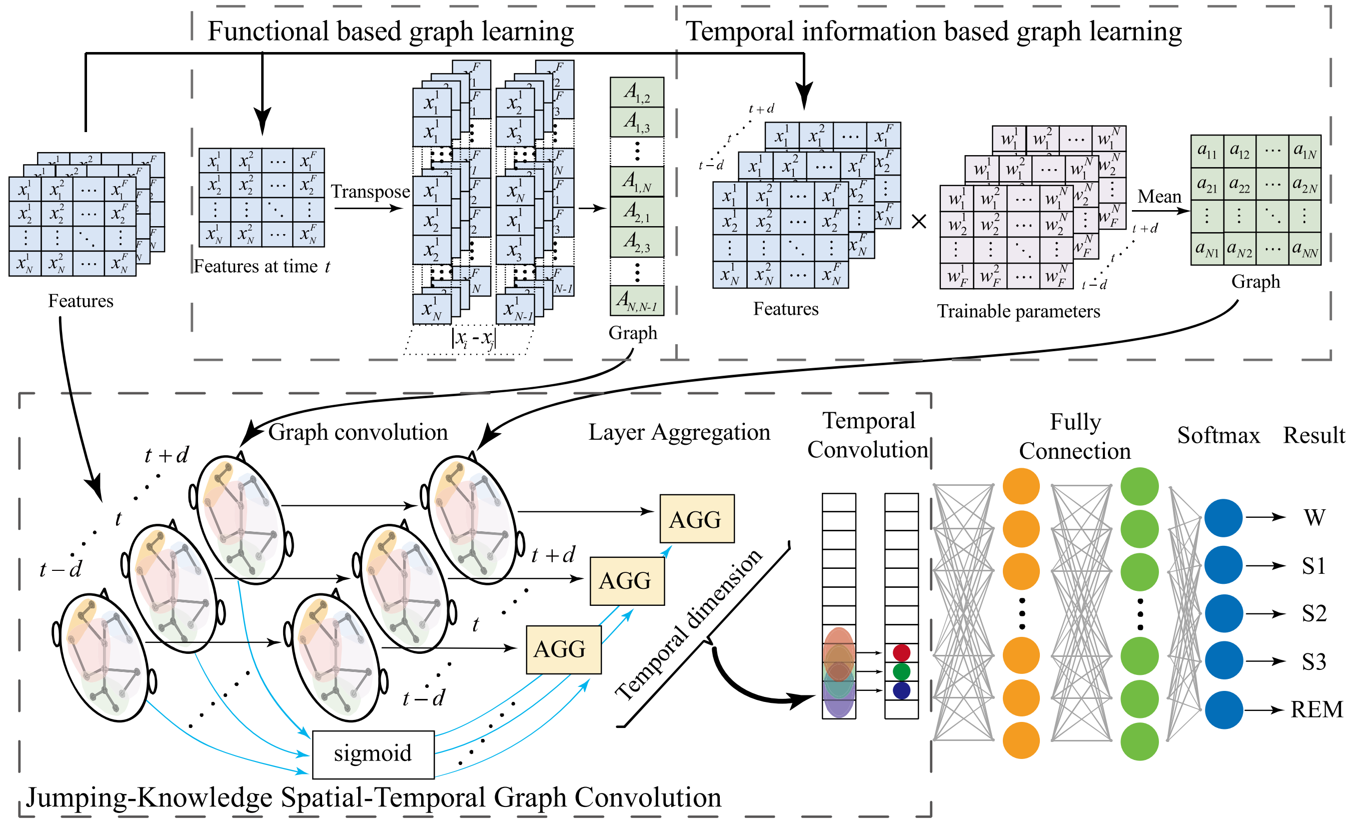

Fig. 3. The structure of the JK-STGCN model. The features are used to generate two adaptive graphs for the jumping knowledge spatial-temporal graph convolution module. The features of $t+2d$ time steps are utilized for the spatial features extraction. The temporal features are extracted by the 2D standard convolution.

defined in the equation (4). $\mathcal{L}_{\text{cross\_entropy}}$ denotes the original loss function as defined in equation (5):

$$\mathcal{L}_{\text{cross\_entropy}} = -\frac{1}{L}\sum_{i=1}^{L}\sum_{r=1}^{R} y_{i,r}\log\hat{y}_{i,r} \tag{5}$$

where L denotes the number of samples, R denotes the number of classes. y is the true label and $\hat{y}$ is the predicted value.

B. Jumping Knowledge Spatial-Temporal Graph Convolution

The jumping knowledge spatial-temporal graph convolution module is a combination of spatial graph convolution and temporal convolution based on the JK-Net structure as mentioned previously, and the spatial graph convolution has the ability to capture spatial features from neighboring graph nodes at the same epoch and the temporal convolution exploits temporal dependencies from nearby epochs.

1) Spatial Graph Convolution: In this study, a GCN is utilized from the perspective of spectral graph theory, and the $K-1$ order Chebyshev polynomials is adopted to reduce the computational complexity

The Laplacian matrix is defined as [42]:

$$L = D - A \tag{6}$$

where A is an adjacency matrix learned based on equation (1) or equation (3), and $D \in \mathbb{R}^{N\times N}$ denotes the diagonal degree matrix of A.

The graph convolution on input x is defined as [49]:

$$g_\theta *_G x = g_\theta(L)x = \sum_{k=0}^{K-1}\theta_k T_k(\widetilde{L})x \tag{7}$$

where g_θ denotes the convolution kernel, $*_G$ is the graph convolutional operation, $\theta \in \mathbb{R}^K$ is a vector of polynomial coefficients. $\widetilde{L} = 2/\lambda_{\max}L - I_N$, where $\lambda_{\max}$ denotes the Laplacian matrix's maximum eigenvalue, I_N denotes the unit matrix. The $K-1$ order Chebyshev polynomials is recursively defined as:

$$T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x) \tag{8}$$

where $T_0(x) = 1$, $T_1(x) = x$.

2) Jumping Knowledge Graph Convolution: Based on the JK-Net, a jumping knowledge module is used to extract the spatial information from each node and to aggerate features from different layers. This aggregating layer can be formulated as:

$$\begin{aligned}\text{AGG}^h &= \text{ReLu}(g_\theta *_G \chi^{(l-1)}) \\ &\quad + \text{sigmoid}(\text{ReLu}(g_{\theta'} *_{G'} \chi^{(l-2)}))\end{aligned} \tag{9}$$

where g_θ and $g_{\theta'}$ are different convolution kernels defined by equation (7), $*G$ and $*G'$ are graph convolution based on two adaptive graphs learned through equation (1) and equation (3), $\chi^{(l-1)}$ and $\chi^{(l-2)}$ are inputs of graph convolution layer $l-1$ and graph convolution layer $l-2$, ReLu and

TABLE I

Numbers of Epochs for Each Sleep Stage From ISRUC-S1 and ISRUC-S3 Datasets

	W	N1	N2	N3	REM	Total
ISRUC-S3	1651	1215	2609	2014	1060	8549
ISRUC-S1	20098	11062	27511	17251	11265	87187

sigmoid are activation functions as defined below:

$$\mathrm{ReLu}(x) = \max(0, x) \tag{10}$$

$$\mathrm{sigmoid}(x) = \frac{1}{1+e^{-x}} \tag{11}$$

3) Temporal Convolution: Common standard 2D convolution layers are utilized to extract temporal features after a spatial graph convolution layer. Based on the combination of sufficient extracted localized spatial features and aggregated localized spatial features at each epoch, the temporal convolution learns the transition rules from the neighboring epochs of the current sleep stages. The temporal convolution of the l-th 2D convolution layer is defined as:

$$\chi^{(l)} = \sigma(\Phi * (\sigma(\mathrm{AGG}^{(l-1)}))) \in \mathbb{R}^{N\times C_l \times T_l} \tag{12}$$

where σ is the activation function, Φ denotes the convolution kernel, $*$ is the standard convolution operation, and AGG denotes the output of the aggregate layer defined as the equation (9), C_l is the number of channels, and T_l is the l-th layer's temporal dimension.

V. Experiments

A. Datasets Used and Experiment Setting

In this study, experiments are conducted on two subsets of the ISRUC-Sleep database [50]: 1) Both the ISRUC-S3 and the ISRUC-S1 data are utilized to evaluate the classification performance of the proposed model. The ISRUC-S3 subgroup contains 10 healthy adults (9 males and 1 female, aged from 30 to 58). The ISRUC-S1 subgroup contains 100 adults with evidence of having sleep disorders (55 males and 45 females, aged from 20 to 85). Each recording from these two subgroups contains 2 EOG channels (LOC-A2 and ROC-A1), 6 EEG channels (F3-A2, C3-A2, O1-A2, F4-A1, C4-A1, and O2-A1), 3 EMG channels (Chin EMG, left leg movements and right leg movements), and 1 ECG channel, and all signals were sampled at 200Hz. The PSG was segmented into 30-second-length epochs and annotated by two experts according to the AASM standards. 2) The ISRUC-S1 data is also used to test the generality of the proposed method. The distribution of sleep stages is shown in TABLE I.

The inputs to the proposed model are extracted from a standard CNN named FeatureNet. This feature extractor aims to extract high-level features from the raw input feature matrix, which means that 3000 original data points from each channel at each epoch will be transferred into a 256-dimension feature vector. 2 EOG channels, 6 EEG channels, 1 EMG channel (Chin EMG), and 1 ECG channel are fed into the CNN to extract features. After that, these extracted features are fed into the proposed model for classifying sleep stages. Detailed hyper-parameters are shown in TABLE II, where the parameter 'neighboring epoch size' means the number of neighboring epochs to aggregate temporal functional connections among brain regions, and the parameter 'Order of Chebyshev polynomials' is set to 9 to aggerate the spatial information from all nine neighboring channels at each epoch.

TABLE II

Hyper-Parameters of JK-STGCN

Hyperparameter	Value
Neighboring epoch size	5
Layer number of functional graph learning	1
Layer number of temporal graph learning	1
Layer number of graph convolution	2
Temporal convolution kernels	10
Order of Chebyshev polynomials	9
Regularization parameter of graph learning	0.0005
Dropout probability	0.5
Number of training epochs	80
Batch size	64
Learning rate	0.0001
Optimizer	Adam

To evaluate the classification performance of the proposed method, we compare it with traditional machine learning methods, Euclidean-inputs deep learning algorithms like CNNs, RNNs, U-Nets, and existing GCN models on the ISRUC-S3 subgroup and further evaluation experiments for deep learning methods are carried out on the ISRUC-S1 subset. For a fair comparison with the MSTGCN model proposed in [47], we use the same features extracted from the FeatureNet to test the performance, due to the fact that the inputs of both the proposed model and MSTGCN are extracted from the FeatureNet, which means the performance of these two methods may be influenced by the CNN. Moreover, the code is uploaded on Github (https://github.com/XiaopengJi-USQ/JK-STGCN).

The evaluation measures accuracy (ACC), Cohen's kappa (κ), precision (PR), recall (RE) and F1-score (F1) are defined as below:

$$Accuracy = \frac{TP+TN}{TP+FN+FP+TN}\% \tag{13}$$

$$\kappa = \frac{p_0 - p_e}{1-p_e} \tag{14}$$

where p_0 is the overall accuracy of the model and p_e is the hypothetical probability of chance agreement.

$$precision = \frac{TP}{TP+FP}\% \tag{15}$$

$$F1 = \frac{2\times RE\times precision}{RE+precision} \tag{16}$$

All these experiments are conducted in a computer with an Intel I9-10900K CPU, 64 GB Memory and a Nvidia 2080ti GPU.

B. Comparison With the State-of-the-Art Methods

The details of the performance comparison with these baselines on the ISRUC-S3 subgroup data are presented in TABLE III.

The performance of traditional machine learning algorithms heavily depends on researchers' prior knowledge and feature engineering, which means both the spatial features and temporal features cannot be extracted effectively. As a result, their

TABLE III
COMPARISON BETWEEN JK-STGCN AND OTHER DEEP LEARNING METHODS ON ISRUC-S1 SUBGROUP

	Method	Overall Metrics			Per-class F1-score (F1)				
		ACC	F1	κ	W	N1	N2	N3	REM
Alickovic et al. [22]	SVM	0.733	0.721	0.657	0.868	0.523	0.699	0.786	0.731
Memar et al. [25]	RF	0.729	0.708	0.648	0.858	0.473	0.704	0.809	0.699
Supratak et al. [15]	CNN+BiLSTM	0.788	0.779	0.730	0.887	**0.602**	0.746	0.858	0.808
Supratak & YiKe [18]	CNN+RNN	0.746	0.736	0.672	0.797	0.540	0.741	0.833	0.768
Jia et al. [34]	U^2-Net	0.799	0.786	0.742	0.860	0.589	0.793	0.886	0.802
Perslev et al. [33]	U-Net	0.770	0.764	-	0.900	0.550	0.780	0.740	0.850
Jia et al. [44]	STGCN	0.799	0.787	0.741	0.878	0.574	0.776	0.864	0.841
Jia et al. [47]	MSTGCN	0.821	0.808	0.769	0.894	0.596	0.806	0.890	**0.856**
proposed model	JK-STGCN	**0.831**	**0.814**	**0.782**	**0.900**	0.598	**0.826**	**0.901**	0.845

* W=wake. N1, N2 and N3 are sleep stage 1, 2, 3, separately, and are non-rapid eye movement. REM= rapid eye movement.

performance cannot be as high as those by deep learning methods.

In terms of deep learning algorithms, both CNNs and RNNs have the ability to extract spatial features or temporal features from original data effectively. However, they ignore the importance of potential connections (relationships) among different channels, which also limits their performances. Because of their special architecture, the U-Net model and U^2-Net model perform as well as the traditional CNNs and RNNs on the ISRUC-S3 dataset. However, their complex architecture and large size training set requirements limit their application.

Although the GraphSleepNet model and the MSTGCN model consider the functional connection among different brain regions and reach higher performance than most Euclidean-inputs deep learning methods like CNNs, RNNs, and the U-Net model, they fail to consider the effects of neighboring nodes from neighboring epochs.

For the classification results, the JK-STGCN can identify most of the corresponding stages. The stage of Wake, N2, and N3 achieve the highest performance among all the algorithms. The reason that stage N1 has a lower classification is because N1 is a transitional stage between the Wake and N2 stages, which means its characteristic is not as clear as the deep sleep stages. From the classification results in TABLE IV, we can find that the JK-STGCN model can classify most Wake stage and most deep sleep stages successfully. The non-symmetric confusion matrix indicates that these misclassifications are caused by the imbalanced class data.

The classification performance of deep learning algorithms can be affected by the dataset size. To further evaluate the classification performance of non-Euclidean inputs models and Euclidean inputs models on large dataset size, the classification experiments were also conducted on the ISRUC-S1 subgroup. 50 subjects are randomly selected from the ISRUC-S1 subset for 25-fold cross validation. The results in TABLE V demonstrate that the JK-STGCN model has more reliable performance compared with other models.

TABLE IV
CONFUSION MATRIX OBTAINED FROM 10-FOLD VALIDATION ON ISRUC-S3 DATASET

	Predicted					Per-class Metrics	
	W	N1	N2	N3	REM	PR	RE
W	1499	106	30	8	8	0.891	0.908
N1	143	656	273	3	140	0.670	0.540
N2	30	142	2258	138	41	0.790	0.865
N3	3	1	234	1776	0	0.921	0.882
REM	7	74	62	3	914	0.829	0.862

The execution time of a model reflects the complexity of its architecture and its efficiency. Under the same computer setting and similar classification accuracy, the shorter time it takes, the higher efficiency the model has. Models with the top three classification performances on both the ISRUC-S1 and ISRUC-S3 are selected to compare their training time and testing time. Considering the different structures of each model, several training parameters, such as the features extracted from the FeatureNet, the training epochs, batch size, and others are set the same for the MSTGCN and the JK-STGCN. However, the architecture of the U^2-Net model is much more complex than these two GCN methods. As a result, all training parameters of the U^2-Net are set the same as those in [34]. As Fig. 4 illustrates, both the training time and the training time plus the feature extraction time of the proposed model are much lower than those by the MSTGCN and the U^2-Net model. The main reason is that the parameter size of MSTGCN or the U^2-Net is much larger than that in JK-STGCN, which means the two models are much more complex than the JK-STGCN model. Fig. 5 illustrates the testing time of the JK-STGCN, the MSTGCN and the U^2-Net model on 50 subjects. The testing time of the proposed method is a little shorter than the MSTGCN's testing time, which are both around 7 seconds, whereas the U^2-Net takes about 37 seconds to complete the same prediction task. The faster prediction speed and smaller storage space requirement of the JK-STGCN model make it possible to deploy this algorithm to some edge artificial intelligence devices, like the smartphone and the smartwatch.

C. Model Analysis

Experiments above demonstrate that the JK-STGCN model has the capacity to classify sleep stages on both healthy subjects and unhealthy cases. The results in TABLE III and

TABLE V
COMPARISON BETWEEN JK-STGCN AND OTHER DEEP LEARNING METHODS ON ISRUC-S1 SUBGROUP

	Method	Overall Metrics			Per-class F1-score (F1)				
		ACC	F1	κ	W	N1	N2	N3	REM
Supratak et al. [15]	CNN+BiLSTM	0.717	0.691	0.638	0.823	0.466	0.738	0.809	0.621
Supratak & YiKe [18]	CNN+RNN	0.778	0.758	0.714	0.883	0.532	0.764	0.848	0.763
Jia et al. [34]	U^2-Net	0.815	**0.801**	0.762	**0.899**	**0.570**	0.800	0.878	0.857
Perslev et al. [33]	U-Net	0.770	0.770	-	0.890	0.520	0.790	0.770	**0.880**
Jia et al. [44]	STGCN	0.786	0.754	0.723	0.884	0.437	0.775	0.838	0.835
Jia et al. [47]	MSTGCN	0.804	0.785	0.748	0.887	0.545	0.791	0.872	0.832
proposed model	JK-STGCN	**0.820**	0.798	**0.767**	0.895	0.550	**0.811**	**0.883**	0.850

* W=wake. N1, N2 and N3 are sleep stage 1, 2, 3, separately, and are non-rapid eye movement. REM= rapid eye movement.

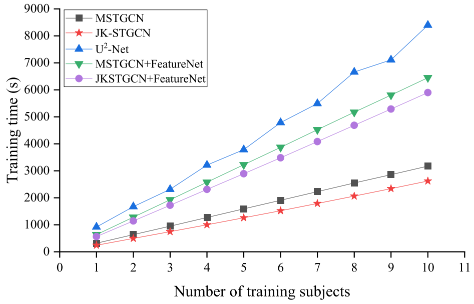

Fig. 4. Training time for the U^2-Net, MSTGCN and JK-STGCN based on *k*-fold training.

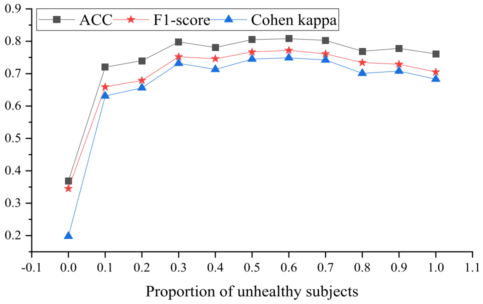

Fig. 6. The trend of accuracy, F1-score and Cohen kappa with different proportion of unhealthy subjects in training set.

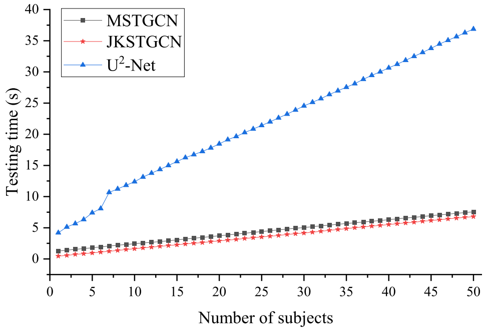

Fig. 5. Testing time for the U^2-Net, MSTGCN and JK-STGCN based on 50 subjects.

TABLE V imply that the classification performance may be affected by two factors, one being the proportion of unhealthy subjects in the training set and the other being the size of the training set. To further evaluate the effects of these two factors on the classification performance, two more experiments are conducted on the ISRUC-S1 subset which contains 100 patients with sleep disorders like sleep apnea obstructive syndrome, periodic limb movements of sleep, etc.

1) Effects of the Proportion of the Unhealthy Subjects in Training Set: The testing set contains 10 patients which are randomly selected from the ISRUC-S1 subset. The initial training set contains ten healthy subjects selected from the ISRUC-S3. The proportion of the unhealthy subjects in the training set is changed by removing one healthy subject from the training set randomly and adding a new random unhealthy subject from the rest subjects in ISRUC-S1 and this operation repeats until all healthy subjects are removed.

Fig. 6 illustrates the changing trend of accuracy, F1-score, and Cohen kappa of the experiment above. The classification accuracy of the JK-STGCN model is much lower in this disordered sleep stage classification task when all the training data is from the healthy subjects. However, the classification accuracies rise rapidly when there are 10% unhealthy subjects. Then the performance improves slowly as the proportion of unhealthy subjects in the training set increases. The JK-STGCN achieves the best performance to classify disordered sleep stages when the unhealthy subjects reach 60%. After that, the performance reduces slightly as the proportion of unhealthy data increases. It is believed that the classification performance improves as the unhealthy data increase at first mainly because the JK-STGCN model starts to learn and recognize the features in abnormal bio-signals and this leads to the improvement of abnormal bio-signals classification. However, the JK-STGCN model starts to misclassify the normal bio-signals, when abnormal bio-signals and abnormal transition ratio reach a high level, resulting in a performance reduction. Even though the classification accuracies are heavily affected by the ratio of the healthy subjects to unhealthy patients, this negative effect may be eliminated by increasing the training set size.

2) Effects of the Size of the Training Set: The ISRUC-S1 subset is randomly divided into four disjoint subgroups, and each subgroup contains 10, 20, 30, and 40 patients respectively. One-subject validation is carried out on each subgroup to validate the influence of the train set size on the JK-STGCN model. As shown in Fig. 7, the classification performance keeps rising with the training set size increases. It is believed that the JK-STGCN model has the capacity to learn and

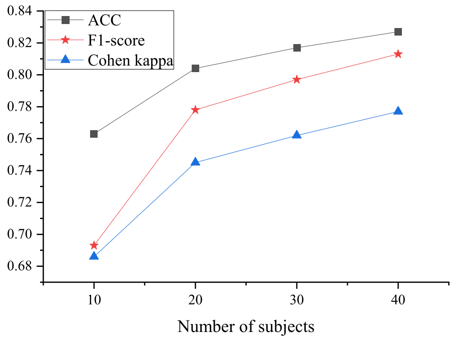

Fig. 7. The performance of JK-STGCN on four disjoint groups from ISRUC-S1.

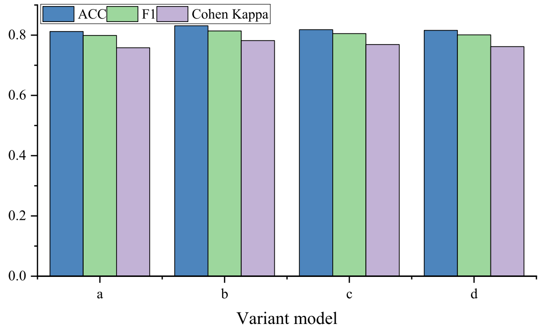

Fig. 8. Comparison of the designed variant models.

recognize both the normal data and the abnormal data if there are sufficient training data.

3) Ablation Experiment: To explore the effects of each module used in the proposed model, four variant models are designed and evaluated using the ISRUC-S3 database. The details are described below:

1. *variant a (basic model)*: The basic model is an independent adaptive graph learning STGCN model without the jumping knowledge spatial-temporal graph convolutional module.

2. *variant b (+ jumping knowledge spatial-temporal graph convolutional module)*: The jumping knowledge spatial-temporal graph convolutional module is added to the basic model to form a JK-STGCN model.

3. *variant c (+spatial attention)*: A spatial attention layer is added to the JK-STGCN to indicate the importance of different channels.

4. *variant d (+spatial attention and temporal attention)*: A spatial attention layer and a temporal attention layer are both added to the JK-STGCN to learn the importance of different channels and different sleep epochs.

As Fig. 8 illustrates, the basic model has the lowest performance among all these variant models. The main reason may be that the parameter size is too small to learn such complex spatial-temporal features, even the adaptive graph learning algorithm provides the optimal connections among channels.

The performance improves when the jumping knowledge spatial-temporal graph convolutional module is added to the basic STGCN model. According to [48], GCN models can achieve the best accuracies when there are two graph convolutional layers with residual connections. The classification results also demonstrate that the two-layer GCN model with residual connections may extract sufficient spatial features, which are more important than global information. The classification accuracies decrease when the attention mechanisms are added. The reason is that the attention mechanisms pay more attention to the important channels and sleep EEG segment sequences, which means some unimportant factors that are related to channels and epochs are ignored, resulting in an inefficient information extraction for sleep stage classification.

VI. Conclusion

In this paper, a JK-STGCN model is proposed to classify sleep stages. The JK-STGCN model contains two adaptive graph learning layers that explore intrinsic connections and relationships among multi-channel bio-signals during sleep stage classification. A jumping knowledge spatial-temporal graph convolution module is designed to extract spatial features and temporal features, which helps the model learn transitional rules among epochs. The experimental results on the ISRUC-S3 subset show that the overall accuracy, the F1-score, and Cohen kappa reached 0.831, 0.814, and 0.782, respectively, which is much better in performance compared to those Euclidean-input deep learning methods and the existing STGCN methods. The experimental results on the ISRUC-S1 subset demonstrate its high performance of sleep stage classification on unhealthy subjects, compared to other deep learning baselines. In addition, extensive experiments are carried out to evaluate the training time and testing time among the top three models. The fastest training speed and prediction speed imply that the proposed model has the ability to be deployed on edge artificial intelligence devices. Moreover, the effects of the distribution of the datasets on the classification performance are explored. The results indicate that the proposed model has reliable robustness to classify both normal data and abnormal data when there is sufficient training data. The ablation experiment is also conducted to find the most important module of the proposed model. Even though the JK-STGCN demonstrates its high performance on sleep stage classification, there is still some space to improve. One drawback is that the GCN model is a multi-channel-based classification algorithm, which means the storage space of the dataset it requires is larger than single-channel-based classification algorithms. One solution is to use the connections among frequency bands instead of the connections among channels and this change allows GCN to classify sleep stages by using a single channel bio-signal, which can decrease the storage space and accelerate the training speed and testing speed. Another improvement that may be considered in the future is the jumping-knowledge module. In the proposed model, the jumping-knowledge operation only happens in each epoch, rather than happens among neighboring epochs. It is believed that the neighboring-epoch-crossed jumping operation would help the aggregate layers to aggregate both spatial and temporal information from the graph convolutional layers, and it would also help the standard temporal convolutional layers to learn the transition rules effectively.

References

[1] T. Young, P. E. Peppard, and D. J. Gottlieb, "Epidemiology of obstructive sleep apnea: A population health perspective," *Amer. J. Respiratory Crit. Care Med.*, vol. 165, no. 9, pp. 1217–1239, May 2002.

[2] H. J. Cho, H. Lavretsky, R. Olmstead, M. J. Levin, M. N. Oxman, and M. R. Irwin, "Sleep disturbance and depression recurrence in community-dwelling older adults: A prospective study," *Amer. J. Psychiatry*, vol. 165, no. 12, pp. 1543–1550, Dec. 2008.

[3] N. A. Siuly, Y. Li, and P. Wen, "Identification of motor imagery tasks through CC-LR algorithm in brain computer interface," *Int. J. Bioinf. Res. Appl.*, vol. 9, no. 2, p. 156, 2013.

[4] E. A Wolpert, "A manual of standardized terminology, techniques and scoring system for sleep stages of human subjects," *Arch. Gen. Psychiatry*, vol. 20, no. 2, pp. 246–247, 1969.

[5] R. B. Berry *et al.*, "The AASM manual for the scoring of sleep and associated events," *Rules, Terminology Tech. Specifications, Darien, Illinois, Amer. Acad. Sleep Med.*, vol. 176, pp. 2012, 2012.

[6] A. Roebuck *et al.*, "A review of signals used in sleep analysis," *Physiol. Meas.*, vol. 35, no. 1, pp. R1–R57, Jan. 2014.

[7] M. Diykh, Y. Li, and P. Wen, "EEG sleep stages classification based on time domain features and structural graph similarity," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 24, no. 11, pp. 1159–1168, Nov. 2016.

[8] R. Sharma, R. B. Pachori, and A. Upadhyay, "Automatic sleep stages classification based on iterative filtering of electroencephalogram signals," *Neural Comput. Appl.*, vol. 28, no. 10, pp. 2959–2978, Oct. 2017.

[9] L. Zoubek, S. Lesecq, F. Chapotot, S. Charbonnier, and A. G. C. Buguet, "Feature selection for sleep/wake stages classification using data driven methods," *Biomed. Signal Process. Control*, vol. 2, no. 3, pp. 171–179, Jul. 2007.

[10] A. Stochholm, K. Mikkelsen, and P. Kidmose, "Automatic sleep stage classification using ear-EEG," in *Proc. 38th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Aug. 2016, pp. 4751–4754.

[11] O. Tsinalis, P. M. Matthews, and Y. Guo, "Automatic sleep stage scoring using time-frequency analysis and stacked sparse autoencoders," *Ann. Biomed. Eng.*, vol. 44, no. 5, pp. 1587–1597, May 2016.

[12] W. Al-Salman, Y. Li, and P. Wen, "Detection of EEG K-complexes using fractal dimension of time frequency images technique coupled with undirected graph features," *Frontiers Neuroinform.*, vol. 13, p. 45, Jun. 2019.

[13] M. Diykh, Y. Li, and S. Abdulla, "EEG sleep stages identification based on weighted undirected complex networks," *Comput. Methods Programs Biomed.*, vol. 184, no. 6, Feb. 2020, Art. no. 105116.

[14] M. T. Sadiq, X. Yu, and Z. Yuan, "Exploiting dimensionality reduction and neural network techniques for the development of expert brain–computer interfaces," *Expert Syst. Appl.*, vol. 164, Feb. 2021, Art. no. 114031.

[15] A. Supratak, H. Dong, C. Wu, and Y. Guo, "DeepSleepNet: A model for automatic sleep stage scoring based on raw single-channel EEG," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 11, pp. 1998–2008, Nov. 2017.

[16] L. Wei, Y. Lin, J. Wang, and Y. Ma, "Time-frequency convolutional neural network for automatic sleep stage classification based on single-channel EEG," in *Proc. IEEE 29th Int. Conf. Tools With Artif. Intell. (ICTAI)*, Nov. 2017, pp. 88–95.

[17] E. Bresch, U. Großekathöfer, and G. Garcia-Molina, "Recurrent deep neural networks for real-time sleep stage classification from single channel EEG," *Frontiers Comput. Neurosci.*, vol. 12, p. 85, Oct. 2018.

[18] A. Supratak and Y. Guo, "TinySleepNet: An efficient deep learning model for sleep stage scoring based on raw single-channel EEG," in *Proc. 42nd Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC) Conjunction With 43rd Annu. Conf. Can. Med. Biol. Eng. Soc.*, Jul. 2020, pp. 641–644.

[19] I. N. Yulita, M. I. Fanany, and A. M. Arymuthy, "Bi-directional long short-term memory using quantized data of deep belief networks for sleep stage classification," *Proc. Comput. Sci.*, vol. 116, pp. 530–538, Jan. 2017.

[20] J. Zhou *et al.*, "Graph neural networks: A review of methods and applications," 2018, *arXiv:1812.08434*.

[21] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.

[22] E. Alickovic and A. Subasi, "Ensemble SVM method for automatic sleep stage classification," *IEEE Instrum. Meas.*, vol. 67, no. 6, pp. 1258–1265, Jun. 2018.

[23] Y. L. Siuly and W. Peng, "Classification of EEG signals using sampling techniques and least square support vector machines," in *Rough Sets and Knowledge Technology*. Berlin, Germany: Springer-Verlag, 2009.

[24] L. Fraiwan, K. Lweesy, N. Khasawneh, H. Wenz, and H. Dickhaus, "Automated sleep stage identification system based on time-frequency analysis of a single EEG channel and random forest classifier," *Comput. Methods Programs Biomed.*, vol. 108, no. 1, pp. 10–19, 2012.

[25] P. Memar and F. Faradji, "A novel multi-class EEG-based sleep stage classification system," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 1, pp. 84–95, Jan. 2018.

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[27] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[28] A. W. Yu, H. Lee, and Q. V. Le, "Learning to skim text," 2017, *arXiv:1704.06877*.

[29] N. Michielli, U. R. Acharya, and F. Molinari, "Cascaded LSTM recurrent neural network for automated sleep stage classification using single-channel EEG signals," *Comput. Biol. Med.*, vol. 106, pp. 71–81, Mar. 2019.

[30] H. Sun *et al.*, "Sleep staging from electrocardiography and respiration with deep learning," *Sleep*, vol. 43, no. 7, p. zsz306, Jul. 2020.

[31] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.

[32] A. Sors, S. Bonnet, S. Mirek, L. Vercueil, and J.-F. Payen, "A convolutional neural network for sleep stage scoring from raw single-channel EEG," *Biomed. Signal Process. Control*, vol. 42, pp. 107–114, Apr. 2018.

[33] M. Perslev, S. Darkner, L. Kempfner, M. Nikolic, P. J. Jennum, and C. Igel, "U-sleep: Resilient high-frequency sleep staging," *npj Digit. Med.*, vol. 4, no. 1, pp. 1–12, Dec. 2021.

[34] Z. Jia, Y. Lin, J. Wang, X. Wang, P. Xie, and Y. Zhang, "SalientSleepNet: Multimodal salient wave detection network for sleep staging," 2021, *arXiv:2105.13864*.

[35] G. Zhu, Y. Li, P. Wen, and S. Wang, "Analysis of alcoholic EEG signals based on horizontal visibility graph entropy," *Brain Inform.*, vol. 1, nos. 1–4, pp. 19–25, 2014.

[36] G. Zhu, Y. Li, and P. P. Wen, "An efficient visibility graph similarity algorithm and its application on sleep stages classification," in *Proc. Int. Conf. Brain Inform.*, 2012, pp. 185–195.

[37] D. Duvenaud *et al.*, "Convolutional networks on graphs for learning molecular fingerprints," 2015, *arXiv:1509.09292*.

[38] W. L. Hamilton, R. Ying, and J. Leskovec, "Inductive representation learning on large graphs," 2017, *arXiv:1706.02216*.

[39] A. Vaswani *et al.*, "Attention is all you need," 2017, *arXiv:1706.03762*.

[40] Y. Hou *et al.*, "Deep feature mining via attention-based BiLSTM-GCN for human motor imagery recognition," 2020, *arXiv:2005.00777*.

[41] T. Song, W. Zheng, P. Song, and Z. Cui, "EEG emotion recognition using dynamical graph convolutional neural networks," *IEEE Trans. Affect. Comput.*, vol. 11, no. 3, pp. 532–541, Jul./Sep. 2020.

[42] P. Zhong, D. Wang, and C. Miao, "EEG-based emotion recognition using regularized graph neural networks," *IEEE Trans. Affect. Comput.*, early access, May 11, 2020, doi: 10.1109/TAFFC.2020.2994159.

[43] I. C. Covert *et al.*, "Temporal graph convolutional networks for automatic seizure detection," in *Proc. Mach. Learn. Healthcare Conf.*, 2019, pp. 160–180.

[44] Z. Jia *et al.*, "GraphSleepNet: Adaptive spatial-temporal graph convolutional networks for sleep stage classification," in *Proc. 29th Int. Joint Conf. Artif. Intell. (IJCAI)*, Jul. 2020, pp. 1324–1330.

[45] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 922–929.

[46] A. Hyvärinen, "New approximations of differential entropy for independent component analysis and projection pursuit," in *Proc. Adv. Neural Inf. Process. Syst.*, 1997, pp. 1–7.

[47] Z. Jia *et al.*, "Multi-view spatial-temporal graph convolutional networks with domain generalization for sleep stage classification," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 1977–1986, 2021.

[48] K. Xu, C. Li, Y. Tian, T. Sonobe, K.-I. Kawarabayashi, and S. Jegelka, "Representation learning on graphs with jumping knowledge networks," 2018, *arXiv:1806.03536*.

[49] M. Simonovsky and N. Komodakis, "Dynamic edge-conditioned filters in convolutional neural networks on graphs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3693–3702.

[50] S. Khalighi, T. Sousa, J. M. Santos, and U. Nunes, "ISRUC-sleep: A comprehensive public dataset for sleep researchers," *Comput. Methods Programs Biomed.*, vol. 124, pp. 180–192, Feb. 2016.