

Received September 1, 2020, accepted September 24, 2020, date of publication September 28, 2020, date of current version October 13, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3027289

# Transfer Learning Convolutional Neural Network for Sleep Stage Classification Using Two-Stage Data Fusion Framework

MEHDI ABDOLLAHPOUR<sup>1</sup>, TOHID YOUSEFI REZAAI<sup>1</sup>,  
ALI FARZAMNIA<sup>1</sup>, (Senior Member, IEEE),  
AND ISMAIL SAAD<sup>2</sup>, (Member, IEEE)

<sup>1</sup>Department of Biomedical Engineering, Faculty of Electrical and Computer Engineering, University of Tabriz, Tabriz 5166616471, Iran

<sup>2</sup>Faculty of Engineering, Universiti Malaysia Sabah, Kota Kinabalu 88400, Malaysia

Corresponding authors: Ali Farzamia (ali-farzamia@ieee.org) and Tohid Yousefi Rezaai (yousefi@tabrizu.ac.ir)

This work was supported by the Research and Innovation Management Center (PPPI) and the Faculty of Engineering, Universiti Malaysia Sabah (UMS).

**ABSTRACT** The most important part of sleep quality assessment is the classification of sleep stages, which helps to diagnose sleep-related disease. In the traditional sleep staging method, subjects have to spend a night in the sleep clinic for recording polysomnogram. Sleep expert classifies the sleep stages by monitoring the signals, which is time consuming and frustrating task and can be affected by human error. New studies propose fully automated techniques for classifying sleep stages that makes sleep scoring possible at home. Despite comprehensive studies have been presented in this field the classification results have not yet reached the gold standard due to the concentration on the use of a limited source of information such as single channel EEG. Therefore, this article introduces a new method for fusing two sources of information, including electroencephalogram (EEG) and electrooculogram (EOG), to achieve promising results in the classification of sleep stages. In the proposed method, extracted features from the EEG and EOG signals, are divided into two feature sets consisting of the EEG features and fused features of EEG and EOG. Then, each feature set transformed into a horizontal visibility graph (HVG). The images of the HVG are produced in a novel framework and classified by proposed transfer learning convolutional neural network for data fusion (TLCNN-DF). Employing transfer learning at the training stage of the model has accelerated the training process of the CNN and improved the performance of the model. The proposed algorithm is used to classify the Sleep-EDF and Sleep-EDFx benchmark datasets. The algorithm can classify the Sleep-EDF dataset with an accuracy of 93.58% and Cohen's kappa coefficient of 0.899. The results show proposed method can achieve superior performance compared to state-of-the-art studies on classification of sleep stages. Furthermore, it can attain reliable results as an alternative to conventional sleep staging.

**INDEX TERMS** Convolutional neural network, data fusion, horizontal visibility graph, sleep stage classification, transfer learning.

## I. INTRODUCTION

We spend one-third of our lives in sleep. Sleeping is an important process for the body to preserve its wellness and health. Any abnormalities in the sleep cycle may cause serious problems such as, extreme fatigue, lack of concentration or metabolic problems such as diabetes and obesity [1], [2]. Traditionally, physiological signals such as

EEG, EOG, electromyogram (EMG) and respiration signals are recorded in the sleep clinic or hospital for the scoring sleep stages during overnight sleep, using a method called polysomnography (PSG). The physician manually assigns a specific sleep stage to every 30-sec epoch of signals. American Academy of Sleep Medicine (AASM) [3] or Rechtschaffen and Kales (R&K) [4] rules are used to identify sleep stages. In general, sleep stages include Wake (W), Rapid Eye Movements (REM) and Non-Rapid Eye Movement (NREM) stages. The NREM stage also consists of N1 (light sleep),

The associate editor coordinating the review of this manuscript and approving it for publication was Siddhartha Bhattacharyya.

N2 and N3 (deep sleep) according to the AASM rule. AASM rule is a set of novel instructions, which describes physiological signal characteristics in different sleep stages. AASM rule is the modified version of R&K rule. In the AASM rule the sleep stages are changed from S1, S2, S3, S4, REM to N1, N2, N3 (merging S3 and S4), R and the transition between different sleep stages is also considered in sleep staging [5]. The output of sleep scoring is a sequence of sleep stages over a night that called hypnogram.

In manual sleep scoring, a sleep specialist is required to evaluate the PSG signals epoch by epoch, which is time consuming and complicated task and may be affected by human misjudgment. Besides, subjects may have difficulty falling asleep due to a change in the permanent sleeping place. Computer-aided methods like machine learning algorithms are used for sleep stage classification to solve described problems. Recently many studies have developed machine learning methods for sleep staging to achieve reliable and practical results [6]–[8].

Traditional machine learning methods consist of two main steps: 1. Extracting hand-crafted features according to the characteristics of the signals in different sleep stages. Variety of time domain, frequency domain and non-linear features, which derived from EEG and EOG signals have been used in sleep studies [9]–[11]. Decomposition of the EEG and EOG by wavelet transform (WT) or empiric mode decomposition (EMD) is also useful for extracting hand-crafted features in the sleep stage classification [12]–[14]. These features because of their high discriminant properties for different sleep stages are popular. 2. Using supervised learning algorithms such as support vector machine (SVM) [15], random forest (RF) [10], decision tree (DT) [16], k-nearest neighbor (KNN) [17] and deep neural networks (DNN) [18] to classify feature vectors that belong to different sleep stages.

Traditional classification methods are less flexible and except for a few trainable parameters, the model cannot be adjusted with input data during the training stage. In fact, these models can classify the data according to the structure they are designed for. Hence, they are constrained in achieving desired results. On the other hand, neural networks with up to millions of training parameters are very flexible and can be used for different type data. Neural networks can achieve reliable results through the training of complex and non-linear models for different classification problems [19]. Different types of neural networks have been developed for various machine learning applications. For instance, recurrent Neural network (RNN) classifies the current input by recalling computations of the previous input. It is well established that RNN, e.g. long short-term memory (LSTM) [20] is functional in sequential modeling tasks like sleep stage classification [6], [21]. Convolutional neural network (CNN) is developed for the processing of multidimensional data such as 2D images. Multiple convolutional and pooling layers are used in CNN to create a deep network to ensure recognition of high-level features in images. In recent years, CNNs achieved outstanding results in classification tasks

by training convolutional networks with large numbers of images, using nonlinear activation functions and new methods of regularization [22]. Besides, CNN can perform better than humans in the classification of geometric shapes and sketches [23]–[25]. Due to the complexity of sleep scoring problem and the need for a nonlinear model, several researches have been developed to use neural networks for the classification of sleep stages [26]–[28].

More useful and comprehensive information can be obtained by increasing the number of signals, which is used in sleep staging [29], [30]. Using data fusion methods, plays an important role in the sleep stage classification. Concatenation of extracted features from different sources is commonly used for fusing information of different signals. In manual sleep scoring, PSG signals are used by the sleep expert to achieve reliable results. Hence, methods that only use single-channel signals for classification are less accurate and not highly reliable compared to sleep expert results and not suitable for practical implementation. Data fusion approaches are used for fusing multiple signals to solve the problem of limited information when using a single channel signal [31]. The main purpose of data fusion in machine learning is to use multiple information sources for classification and ensure that important information from different sources does not overlap with each other and redundant and duplicate information is eliminated.

There are different levels of data fusion, including signal level, feature level and decision level. Many studies have employed data fusion techniques to obtain complementary information from multiple PSG signals [32], [33]. Zhang *et al.* [34] after extracting features with sparse variant of the deep belief network from EEG, EOG and EMG signals, used k-nearest neighbor (KNN), support vector machine (SVM) and hidden Markov model (HMM) classifiers for sleep staging. Then voting principle, based on classification entropy are used to fuse different classifiers results in decision level. Chambon *et al.* [35] used different PSG signals combinations for sleep stage classification. Through the training of the CNN with different numbers of EEG, EOG and EMG signals, showed that using more sensors improves performance of the model. Shi *et al.* [29] proposed two-stage multi-view learning algorithm using multi-channel EEG signals for sleep staging. Collaborative representation (CR) and joint sparse representation (JSR) methods used for extracting features from EEG signals. Extracted features from JSR and CR methods are integrated and fed to multiple kernel extreme learning machine for classification. Both feature level and classifier level fusion are used for sleep scoring.

Despite of the positive outcomes achieved by the multi-channel sleep staging methods, they still suffer from some limitation as described in the following. In most of these methods, selecting a large number of signals for classification added redundant information to the problem and increased the computational complexity without a substantial improvement in sleep staging results. Besides, each selected signals should add complementary information [36] to the problem

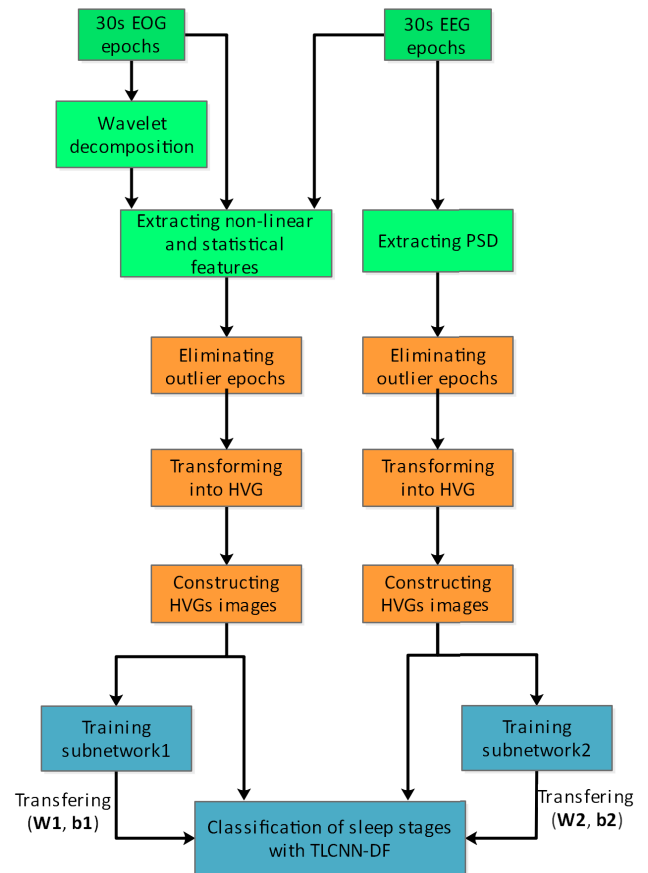
which is not possible with incorrect choice of signals. On the other hand, there are feature fusion methods based on feature concatenation which do not apply post-processing on the fused features. These methods due to the high number of features, impose computational cost on the classifier, increase training time of the model and prevent achieving optimal results [37], [38].

The motivations of this work stem from the limitations of previous studies which have not completely achieved reliable and robust results compared to clinical sleep staging. These issues include, use of the limited information source such as single channel signal, lack in utilization of potential information of extracted features and the use of the classifiers that can not fully fit on input data. Therefore, in the proposed method, we tried to resolve mentioned three issues through the use of two sources of information for classification, post-processing of the extracted features, and introducing a transfer learning method for training CNN.

In this paper, we propose a method for fusing EOG and EEG information for sleep stage classification. After decomposing EOG signal to sub-bands with wavelet transform (WT), the hand-crafted features are extracted from EEG, EOG and the wavelet coefficients of EOG. These features is transformed into a horizontal visibility graph (HVG) in two feature sets. Afterward, the HVGs are mapped to euclidean space with certain rules and graph images are produced. Finally, a novel CNN model, which pre-trained with transfer learning (TL) is used to classify the images of HVGs. The outline of the proposed algorithm is illustrated in Fig. 1.

The main contributions of this study are as follows.

- A novel feature mapping framework is used to transform fused features of EEG and EOG, and PSD of EEG into two sets of HVGs. Then, 2D images of graphs generated for classification. Transforming the features vectors into graphs, enables the use of connection information between feature for classification.
- A CNN structure with two pipelines in input for the fusion and classification of HVG images is proposed. Training of the model takes place in two stages using transfer learning. First, each pipeline which consist of a set of convolutional layers is individually pre-trained in two subnetworks and then transferred to the main model. Second, the main model trained for classifying the sleep stages. With the use of transfer learning for training CNN, overall classification results enhanced and training time of the model is reduced sharply.
- Recognition of the N1 sleep stage is difficult in sleep staging due to the small number of the stage N1 data samples and high correlation between features of the REM and N1 stages. By using proposed transfer learning method the recognition of the N1 and REM sleep stages has improved.
- Proposed sleep staging algorithm is used to classify publicly available Sleep-EDF [39], [40] dataset with PSG signals of 8 subjects and Sleep-EDFx dataset with PSG signals of 20 subjects. The evaluation results show



**FIGURE 1. Workflow of the proposed framework for the fusion and classification of sleep EEG and sleep EOG.**

significant improvement in the sleep stage classification compared to state-of-the-art studies.

The remainder of the study is organized as follows. In Section II, evaluation datasets is described and the framework of the proposed method is explained in detail. In Section III, the evaluation results of the proposed method are compared with other studies and the performance of the proposed method in various conditions is examined. In Section IV, the proposed algorithm is discussed according to the results. Section V draws a conclusion on this study.

## II. MATERIAL AND METHODS

### A. DATASETS

Sleep-EDF and Sleep-EDFx datasets were used in this study, which are available in the Public PhysioNet Database [40]. Sleep-EDF dataset contains recorded signals of 8 healthy males and females, 21 to 36 years old, who did not take any medication. The dataset consists of two group of subjects, 4 sleep cassette (SC) recording that collected in 1989 during 24 hours daily life at the home and 4 sleep telemetry (ST) recording that collected in 1994 during overnight sleep in the hospital from subjects who had mild difficulty falling asleep. Sleep-EDFx is expanded version of Sleep-EDF, which contains more PSG records, which also separated into two

groups of participants, SC that obtained from healthy Caucasians aged between 25 and 101 years during two subsequent day-night, to study the effects of age on sleep and ST obtained from 22 Caucasian healthy males and females for the analysis of the effects of temazepam on sleep. We used 10 ST and 10 SC recordings of Sleep-EDFx dataset. Both datasets include two EEG from Pz-Oz and Fpz-Cz channels, a horizontal EOG and a submental chin EMG. EEG and EOG signals have sampling rate of 100 Hz. Sleep stages are manually scored by well-trained technicians in according to R&K rule. The sleep stages are categorized into 8 stages consisting of: Wake, S1, S2, S3, S4, REM, unscored and movement time. We use the AASM criteria in this article, therefore S3 and S4 stages are combined to the N3 stage and unscored and movement time epochs are eliminated from hypnogram and associated signals. In previous studies that the Pz-Oz channel was used instead of the Fpz-Cz channel, results showed improved performance in sleep staging [13]. Thus, in current study an EEG signal from the Pz-Oz channel and a horizontal EOG are used for evaluation of proposed algorithm.

## B. FEATURE EXTRACTION

In the first step, the hand-crafted features, which containing the discriminant characteristics of signals for different sleep stages are briefly introduced. Due to the complexity of EEG and EOG signals different types of features such as temporal, spectral and non-linear features are extracted from signals. Each feature is derived from 30-sec epochs of signal. The extracted features are depicted in the Table 1 in detail.

### 1) POWER SPECTRAL DENSITY (PSD)

Since the frequency activity of the EEG varies with changes in sleep stages, it will be useful to measure the frequency content of the signal. According to the AASM rule, as sleep deepens, high-frequency content of the EEG decreases and the signal waves become slow. PSD is used as a popular hand-crafted feature in sleep studies [41], [42]. The Welch method [43] is used to estimate the power spectrum of a time series. First, the signal is divided into overlapping segments. Fast Fourier transform (FFT) is calculated for each segment and then used to estimate the periodograms. Finally, Welch's PSD estimation is obtained by averaging the periodogram of all segments.

For a time series ( $x$ ) that divided into  $K$  segments, periodogram can be calculated as,

$$P_k(f) = \frac{1}{W} |X_k(f)|^2 \quad (1)$$

$$W = \sum_{m=1}^M \omega^2[m] \quad (2)$$

where  $X_k(f)$  is FFT of  $k$ th segment of signal,  $\omega$  is window function and  $M$  is number of data points in each segment.

**TABLE 1. Extracted features from EEG and EOG for sleep staging.**

Feature Name	Feature Type	Source
PSD	Spectral	EEG
Complexity	Temporal	EEG, EOG
Kolmogorov Complexity	Non-linear	EEG, EOG, EOG sub-bands
Mean Absolute Amplitude	Temporal	EEG, EOG, EOG sub-bands
Mobility	Temporal	EEG, EOG, EOG sub-bands
Kurtosis	Temporal	EEG, EOG, EOG sub-bands
Standard Deviation	Temporal	EOG sub-bands

Eventually, PSD can be calculated as,

$$S_x(v) = \frac{1}{K} \sum_{k=1}^K P_k(v) \quad (3)$$

In this study 27 data samples of PSD from 1.5 Hz to 42.5 Hz are selected as the features.

### 2) DISCRETE WAVELET TRANSFORM (DWT)

The wavelet transform is used for time-frequency decomposition of signals. It represents a signal by the combination of delation and translation of the scaled basic function, which is a finite length signal known as mother wavelet. For implementation of DWT, a low-pass filter and a high-pass filter are applied to the signal at each level of decomposition. The signal is divided into two sub-bands, details and approximations, which include the high frequency and the low frequency information of the signal. EOG decomposition by DWT before extracting hand-crafted features has been shown to be useful in sleep staging [14]. The wavelet transform is applied to every 30-sec epoch of the EOG at 4 levels by the Daubechies mother wavelet, thereby decomposing the EOG into the set of the detail coefficients ( $D_1, D_2, D_3, D_4$ ) and the approximation coefficients ( $A_4$ ).

### 3) STANDARD DEVIATION

Standard deviation indicates dispersion of data around the mean. For a time series,  $x$ ,

$$std(x) = \sqrt{Var(x)} \quad (4)$$

$$Var(x) = \sigma^2 = \frac{1}{N} \sum_{n=1}^N (x_n - \bar{x})^2 \quad (5)$$

### 4) MEAN ABSOLUTE AMPLITUDE

The amplitude of the brain signal varies with the change in frequency activity of EEG. It also changes with the

appearance of different brain waves like delta, theta, alpha and beta, which are common at different sleep stages.

5) HJORTH PARAMETERS

Hjorth parameters are statistical features that related to the variance of the signal and its derivative. It consists of three parameters: activity, mobility and complexity. Mobility and complexity are used in the proposed method.

$$Mobility = \sqrt{\frac{Var(x')}{Var(x)}} \tag{6}$$

$$Complexity = \frac{Mobility(x')}{Mobility(x)} \tag{7}$$

6) KURTOSIS

Kurtosis is the measure of the data distribution. This feature calculates the deviation degree of the data from gaussian distribution.

$$Kurtosis = \frac{m_4}{m_2^2} \tag{8}$$

where  $m_n$  is  $n$ th central moment of the signal.

7) KOLMOGOROV COMPLEXITY

Andrey Kolmogorov has introduced an algorithmic approach for the quantitative definition of information called Kolmogorov complexity (KC) [44]. KC of a signal, is the length of the shortest computer binary program, which produces that signal as output [45]. KC measures the information, which related to the randomness of a signal.

C. FEATURE MAPPING

Before feature mapping, epochs with outlier features are removed using the z-score method then all features are normalized. The Z-score or standard score changes data distribution to the normal distribution by centering the data and dividing it by standard deviation. A threshold is then applied to the data to eliminate outliers. All the extracted features are divided into two feature sets. The first feature set contains the PSD sample points of EEG and the second feature set contains fused statistical and non-linear features of EEG, EOG and EOG sub-bands.

1) HORIZONTAL VISIBILITY GRAPH (HVG)

Recently, transforming physiological signals into graphs such as correlation graphs and visibility graphs has been considered for machine learning applications [46], [47]. Visibility algorithms are usually used to transform a time series or a set of data points to a graph [48]. A graph consists of nodes and edges. Each node corresponding to a weight and each edge indicates the relation between two nodes, which connected to them. Each feature in the feature set is depicted as a node in the corresponding HVG and the equation 5 that introduced by Lacasa [49] is used to determine the connection between nodes. The nodes,  $i$  and  $j$  are connected if:

$$i < n < j, \quad x_i, x_j > x_n \tag{9}$$

where  $x_n$  is  $n$ th feature in feature vector.

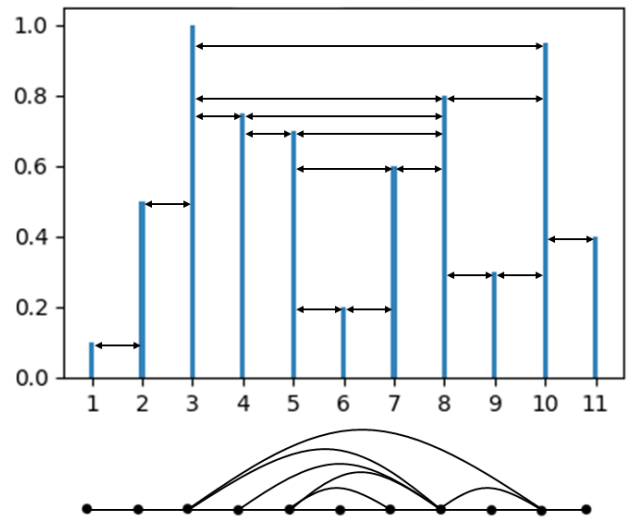


FIGURE 2. The transformation of the feature set (0.10, 0.50, 1.00, 0.75, 0.70, 0.20, 0.60, 0.80, 0.95, 0.40) into HVG.

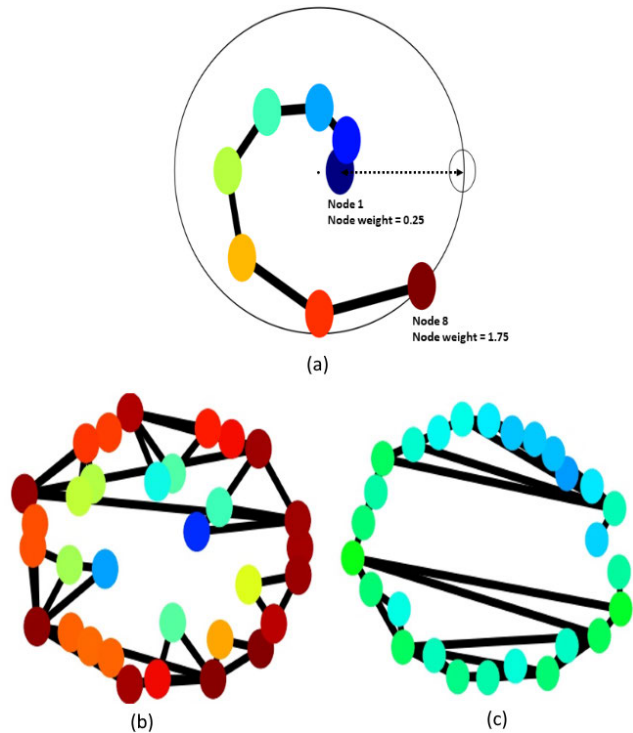
An example of the HVG is shown in Fig. 2. Furthermore, the geometric procedure for establishing connections between nodes, based on the node values, is demonstrated in the bar chart.

2) CONSTRUCTION OF HVG IMAGE

Graph kernels are commonly used for classifying graphs. Despite the achievement of acceptable results using graph kernels, they are subject to several limitations which discussed in the following. First, the classification of graphs using graph kernels is based on the computing similarity between each possible pair of graphs in the dataset, which has a high computational complexity. As a result, as the size of the dataset increases, the computational cost and training time of the model increases at a much higher rate [50]. Second, graph kernels compare two graphs based on the substructures (trees, paths, etc) of the graphs. Therefore, local features of the graphs are used as a criterion for the classification of graph and the global features of the graphs are ignored [51].

These issues have led us to use CNN for graph classification. On the other hand, graphs are defined in topological space so that, they defined by nodes and connections and they do not have specific shape. Therefore, we introduced a framework for drawing graphs in 2D euclidean space with a specific structure so that, the changes in graphs for different classes can be recognized by the CNN. In constructing graph images two key characteristics of the CNNs are considered. 1- Local receptive field of CNN layers guaranteed extracting local features of image. Therefore, CNN is not sensitive to the arrangement of nodes. 2- The relative positions of extracted local feature are important for classification. Therefore, the main structure of drawing should be preserved for all of graphs. [52]

In introduced framework a circular graph is drawn for each HVG and the following two parameters of circular graph change as the weights of the nodes change.



**FIGURE 3.** An example of constructed image from the HVGs. (a) HVG of feature vector (0.25, 0.46, 0.68, 0.89, 1.10, 1.32, 1.53, 1.75) (b) Transformed features of EEG, EOG and wavelet decomposed EOG sub-bands. (c) Transformed PSD features of EEG.

- Node position: All of the nodes are placed on circumference of a circle, each node moves away from the center of the circle on the radius by increasing its weight or approaching the center as weight of the node decreases.
- Node color: The color of the nodes changes with respect to the weight of the nodes in the RGB spectrum.

For the implementation of feature mapping framework, networkx [53] library of Python is used. An example of obtained image from a sample HVG with 8 nodes is illustrated in Fig. 3a. As can be seen, in the corresponding graph, the weights of the nodes are increased from minimum to maximum value. The dashed arrow shows the possible positions of node 1 for different weights, and color of the nodes also can be changed by changing the weight of the nodes. Furthermore, HVGs that obtained from a 30-sec epoch of EEG and EOG are shown in Fig. 3. This framework can also be used for feature level data fusion. As shown in Fig. 3b, the generated graph is the result of concatenating non-linear and statistical features of EEG and EOG signals. Fig. 3c shows obtained graph from PSD features of EEG signal.

#### D. TRANSFER LEARNING CONVOLUTIONAL NEURAL NETWORK FOR DATA FUSION (TLCNN-DF)

**Network architecture:** Two images constructed for classification of a 30-sec epoch of sleep signals. Therefore, the structure of the CNN consists of two separate pipelines for each image. One image for EEG features and another image for fused EEG and EOG features. We first designed two

**TABLE 2.** Subnetworks architecture for the classification of the two image sets.

	Type	Size	Numbers	Dropout
Subnetwork1	Conv1	$(2 \times 2)$	32	-
	Max1	$(3 \times 3)$	-	0.5
	Conv2	$(2 \times 2)$	64	-
	Max2	$(2 \times 2)$	-	-
	Conv3	$(2 \times 2)$	128	0.5
	Dense1	-	128	0.4
Subnetwork2	Dense2	-	5	-
	Conv1	$(2 \times 2)$	64	-
	Max1	$(3 \times 3)$	-	0.5
	Conv2	$(2 \times 2)$	128	-
	Max2	$(2 \times 2)$	-	0.5
	Dense1	-	128	0.4
Dense2	-	5	-	

subnetworks for classifying different image sets and then combined them to develop the main CNN. The proposed convolutional neural network for data fusion (CNN-DF) receives 2 images with size of  $(64 \times 64 \times 3)$  at the input at the same time to recognize corresponding sleep stage. Common architecture pattern of CNN layers, which have recently become popular are used for network design [54]. In such a network, after passing the image through the input layer, a set of convolutional filters with a specific size and trainable coefficients is applied to the image. Convolutional filters slide across the entire image, forming feature maps to feed the next layer. The convolutional layers, which are used in the proposed architecture, followed by the rectified linear unit (Relu) activation function. The pooling layer applied to feature maps in order to reduce their spatial size and training parameters. Thus, by using pooling layers the computational cost and training time of the CNN decreases.

Two simple subnetworks are designed by repeating mentioned layer pattern with different size. Besides, softmax activation function is used as CNN classification stage. Table 2 shows the details of the parameters of the subnetworks, which designed to classify two image sets individually. For constructing TLCNN-DF, in both subnetworks the outputs of the pooling layers, before the fully connected (FC) dense layers, are reshaped and concatenated into the 1-dimensional vector and connected to FC layer in main network. TLCNN-DF is shown in Fig. 4.

**Training CNN:** The key concept in training feed-forward CNNs is to use back-propagation error to adjust the model weights. For this purpose, gradient descent algorithm is used

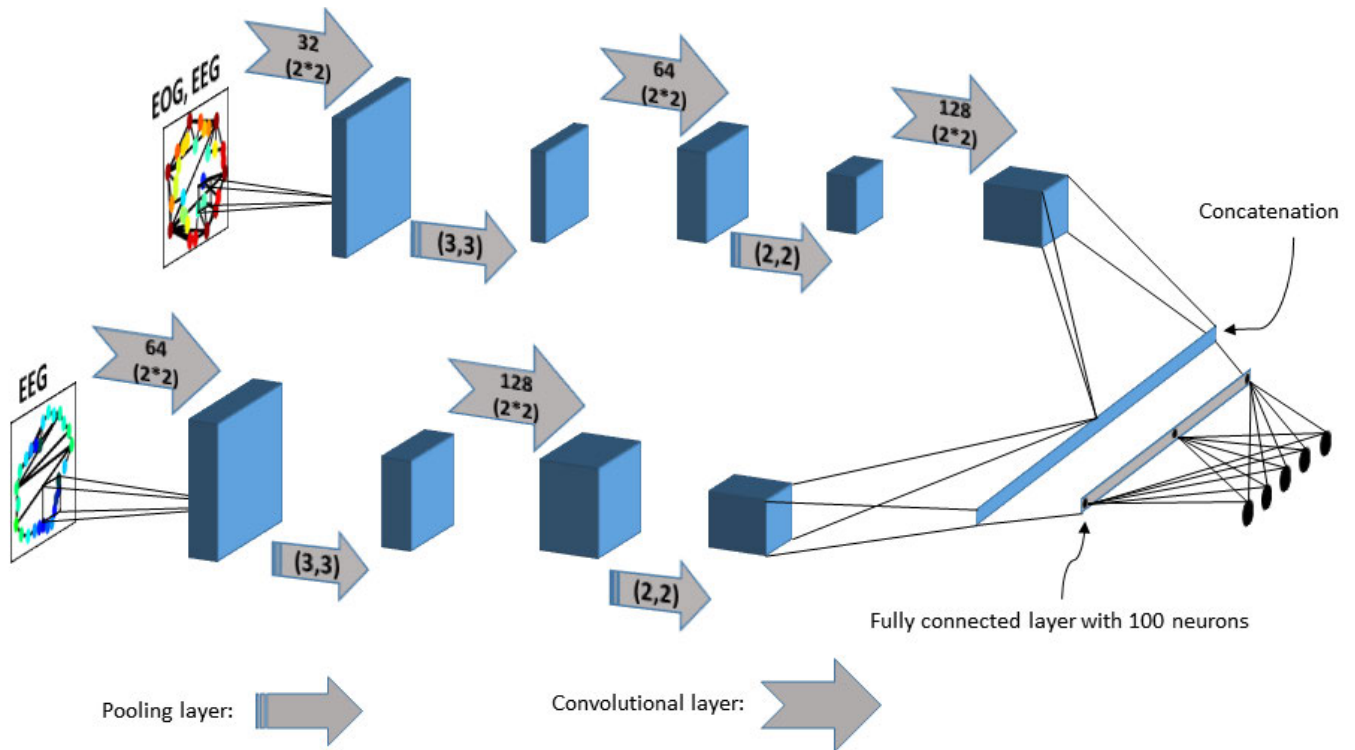


FIGURE 4. Architecture of transfer learning convolutional neural network for data fusion (TLCNN-DF).

to minimize the model error (loss function) by updating weights of the model at each iteration [55].

**Transfer learning:** In machine learning methods, a part of the information from a model that is trained before for a specific task is transferred to a new model with related task to improve the performance of new model [56]. In current study the proposed transfer learning method for training the CNN consists of two stages as described below:

- **Stage1:** Subnetwork 1 and subnetwork 2 as described in table 2 are individual CNNs that can be trained by images with size of  $(64 \times 64 \times 3)$ . Each subnetwork is trained individually by graph images that are constructed in the feature mapping step. Therefore, the images that obtained from the fused EEG and EOG features are used to train subnetwork 1, and the images that obtained from the EEG features are used to train subnetwork 2.

- **Stage2:** Each neuron in the neural network is characterized by its trainable weight and bias  $(W, b)$ . In the CNN, the FC and convolutional layers are composed from neurons. Therefore, model’s training are accomplished through updating weights and biases of the FC and convolutional layers. The weights and biases of the convolutional layers, which are trained in the stage 1, are transferred to the corresponding convolutional layers in the TLCNN-DF. Then, the training capability of the convolutional layers is switched off and two images sets are fed to the TLCNN-DF. At this stage, only FC dense layers can be trained. In other words, we extracted the trained convolution layers from the subnetworks and merged these layers together in the main network.

TABLE 3. Measures of performance assessment.

Sensitivity	Specificity	Precision	F1-Score
$\frac{TP}{TP + FN}$	$\frac{TN}{TN + FP}$	$\frac{TP}{TP + FP}$	$\frac{2TP}{2TP + FP + FN}$

T: True, F: False, N: Negative, P: Positive

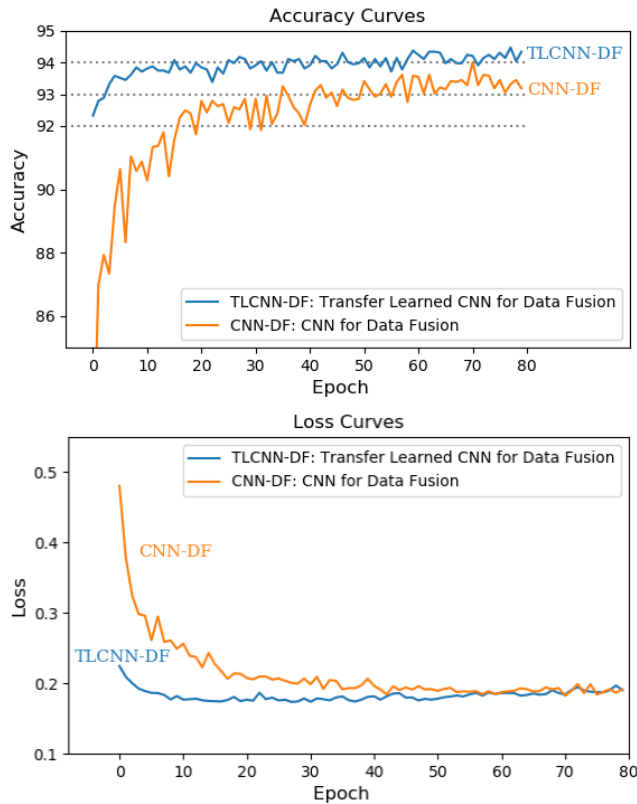
The algorithm was implemented in Python3.6. Furthermore, Tensorflow and Keras libraries were used for designing and evaluating TLCNN-DF. All of the experiments, which are explained in the following, were performed on the 2.4 GHz quad-core CPU.

The results of the experiments are evaluated in terms of accuracy and Cohen’s kappa coefficient [57] in overall assessment, and sensitivity, specificity, precision and F1-Score for per-stage assessment. Table 3 describes the evaluation criteria in detail.

### III. EXPERIMENTS

#### A. EXPERIMENT 1: PERFORMANCE COMPARISON BETWEEN TLCNN-DF AND CNN-DF

As mentioned before, training process is completed in two stages. In the first stage, the weights of the convolutional layers are trained in the subnetworks and transferred to the main model. In the second stage, the training of the main



**FIGURE 5.** Test accuracy and test loss curves of the TLCNN-DF and CNN-DF during training of the models.

model begins. In this experiment, we are going to demonstrate the advantages of transfer learning by evaluating outcomes of the model that trained by two different learning modes: conventional learning (CNN-DF) and transfer learning (TLCNN-DF). In transfer learning mode, the subnetworks are trained in 30 epochs, afterwards in stage 2 the main model is trained in 80 epochs. In conventional learning method the model is trained without transfer learning and the entire model trained in 80 epochs. Performance of the models are compared in terms of overall accuracy and Cohen's kappa also per-stage sensitivity, specificity, F1-score and precision. Furthermore, accuracy and cost changes of the models are illustrated during the training process. For fair comparison, training data samples are exactly the same in the two modes and models are evaluated on the same test data.

Fig. 5 shows the test accuracy and the test cost of the models during the training process. As can be seen, TLCNN-DF has reached an acceptable accuracy (higher than 92%) at the first epoch. Besides, by increasing epochs the accuracy of the model converges to the final value with smaller tolerance in comparison to CNN-DF. Consequently, the results of TLCNN-DF are reproducible and consistent compared to CNN-DF. As shown in Fig. 5 loss of TLCNN-DF is reduced at a higher rate. It can be said that, in less than 20 training epochs, model is fully trained and loss function is minimized.

For the analysis of the outputs of two classifiers, sensitivity, F1-score, specificity and precision for 5 sleep stages are shown in Table 4. As we can observe, the highest sensitivities for sleep stages are obtained for Wake: 99.1%, N1: 60.9%, N3: 85.6% and REM: 92.2% by TLCNN-DF and N2: 94.2% by CNN-DF. TLCNN-DF is classified the sleep stages with an overall accuracy and Cohen's kappa of 94.34% and 0.912 respectively, which is a significant improvement compared to CNN-DF with overall accuracy and Cohen's kappa of 93.19% and 0.894. For more detailed comparison of the performance of the two classifiers see Table 4.

## B. EXPERIMENT 2: TRAINING TIME COMPARISON BETWEEN TLCNN-DF AND CNN-DF

The purpose of this experiment is to analyze the effect of the transfer learning on the training time of the model. We are also investigating that how the classification results change for different number of training epochs in subnetworks. In this experiment, the full training time of the models and the accuracy changes during the main model's training process are considered.

Fig. 6 shows main model's accuracy curves for the various number of training epochs in subnetworks. The scenario is such that the subnetworks are trained in 1, 5, 10, 15, 20, 25 and 30 epochs for different cases. next, the main network is trained in 50 epochs and the test accuracy is monitored. As can be seen, the accuracy curves for different epochs of pre-training convolutional layers do not differ significantly from each other. Accuracy curves in all of the charts fluctuate between 93.5% to 94.5%. In fact, it can be said that in the first time the training data has passed through the subnetworks the convolutional layers are fully trained. Therefore, feeding the training data once into the subnetworks is sufficient for pre-training of the convolutional layers.

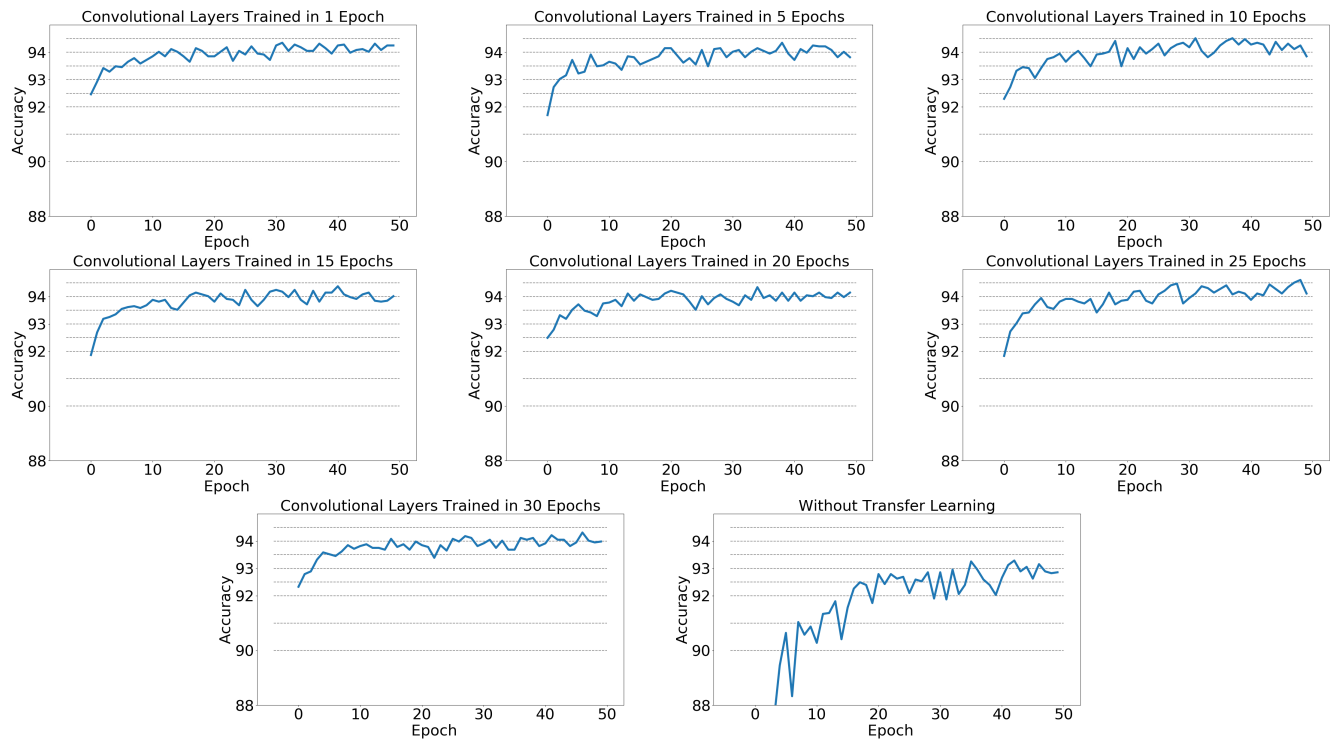
The total training time of the TLCNN-DF in the different cases, which described earlier is shown in Table 5. 12092 30-sec epochs of the Sleep-EDF dataset are used for training the model. As mentioned in Table 5, the CNN-DF is trained faster without transfer learning. However, there is a noteworthy point in case 7, two subnetworks have been trained for 60 epochs (30 epochs each) in total, which means we had 60 more training epochs in case 7 compared to case 8, and yet it can be seen that the training time of the models is close to each other, 302 minutes for case 7 and 283 minutes for case 8. The point is also more evident in case 1, by using transfer learning, the training time of the model is almost one-third of the training time of conventional training method and performance of the model has improved significantly as shown in the Fig. 6. On the other hand, in the test stage, the TLCNN-DF is classified two input images of HVG in just 8 milliseconds.

Since the CNN is trained just once for the classification tasks and training time of the network is not important in sleep staging, we have trained each subnetwork in 30 epochs to create a safe margin for CNN performance. Even though



**TABLE 4. Performance of the proposed CNN without transfer learning (CNN-DF) and with transfer learning (TLCNN-DF).**

	Wake	N1	N2	N3	REM	Wake	N1	N2	N3	REM
	CNN-DF					TLCNN-DF				
Sensitivity	98.7	46.0	94.2	84.8	87.3	99.1	60.9	93.3	85.6	92.2
Specificity	98.7	99.4	95.7	99.3	98.1	98.4	99.0	97.4	99.0	98.6
F1-Score	98.8	57.3	90.8	88.2	85.9	98.9	65.7	92.7	87.7	90.5
Precision	98.8	75.7	87.7	91.7	84.6	98.6	71.4	92.2	90.0	88.9
	accuracy: 93.19, Cohen's kappa: 0.894					Accuracy: 94.34, Cohen's kappa: 0.912				



**FIGURE 6. Test accuracy of TLCNN-DF for different pre-training epochs of convolutional layers in subnetworks.**

training subnetworks in a single epoch has yielded promising results.

**C. EXPERIMENT 3: PERFORMANCE EVALUATION OF THE PROPOSED ALGORITHM WITH BENCHMARK DATASETS**

In small datasets like Sleep-EDF, selection of the test and train data influences the results of the classification. Nevertheless, it appears necessary to use Sleep-EDF dataset in order to compare the results of this paper with other studies. To overcome bias of sample selection issue, the dataset is randomly divided into 5-fold train and test data and fed to TLCNN-DF. Therefore, the final results are obtained by averaging the model results in five runs. Consequently, obtained results will be reliable, robust and reproducible. At each run, at the first stage the weights of the convolutional layers are

trained in subnetworks in 30 epochs and transferred to the main model (CNN-DF), then in the second stage CNN-DF is trained for 50 epochs. In order to analyze the generalizability of the proposed method it is necessary to evaluate its performance on different datasets. Sleep-EDFx is a larger dataset with 305 hours of overnight sleep signals. The results were obtained in single run using the Sleep-EDFx dataset.

Table 6 and Table 7 show the normalized confusion matrices for the Sleep-EDF and Sleep-EDFx datasets, respectively. Per-stage numbers of 30-sec epochs, which used as test data in algorithm are, Wake: 1603, N1: 117, N2: 723, N3: 254 and REM: 327 for Sleep-EDF dataset and Wake: 4037, N1: 223, N2: 1704, N3: 654 and REM: 714 for Sleep-EDFx dataset. The diagonal elements of confusion matrix which highlighted in bold, indicate sensitivity of each sleep stage. Besides,

**TABLE 5. Training time of the proposed CNN in different cases.**

Case	TLCNN-DF							CNN-DF
	1	2	3	4	5	6	7	8
Epoch number for training subnetworks	1	5	10	15	20	25	30	-
Training time (minutes)	118	144	175	207	239	270	302	283

**TABLE 6. Classification results and normalized confusion matrix of TLCNN-DF on Sleep-EDF dataset.**

		Wake	N1	N2	N3	REM
Expert scored	Wake	<b>0.989</b>	0.007	0.001	0.001	0.002
	N1	0.103	<b>0.564</b>	0.154	0.009	0.170
	N2	0.007	0.011	<b>0.923</b>	0.037	0.022
	N3	0.012	0.000	0.134	<b>0.854</b>	0.000
	REM	0.009	0.037	0.055	0.000	<b>0.899</b>
Specificity(%)	98.4	98.9	96.9	98.9	98.6	
F1 Score(%)	98.8	61.7	91.2	86.8	89.1	
Precision(%)	98.6	68.0	90.3	88.2	88.3	
Overall accuracy (%): 93.58, Cohen's kappa: 0.899						

the specificity, precision and F1-score for each sleep stage in Sleep-EDF and Sleep-EDFx dataset and the overall accuracy and Cohen's kappa coefficient are calculated and reported in Table 6 and Table 7.

As can be seen, with the exception of the N1 stage, the sensitivities and F1-scores are higher than 85.4% and 86.8% for the Sleep-EDF dataset and higher than 85.6% and 85.7% for the Sleep-EDFx dataset, respectively. Sensitivity of stage N1 is lower than other stages and difficult to classify due to the small number of epochs for training the model and the poor distinguishing characteristics of the signal at N1 stage. The 30-sec epochs in N1 stage are 4% of the total epoch in Sleep-EDF dataset and 3.3% of total epochs in Sleep-EDFx dataset. Hence, it is difficult for the model to recognize the N1 sleep stage compared to other sleep stages. Overall accuracies and Cohen's kappa coefficients for Sleep-EDF and Sleep-EDFx are 93.58%, 0.899 and 93.16%, 0.889 in 5 class sleep staging, respectively. The results show precise and robust performance of the algorithm for various datasets. Besides, obtained Cohen's kappa coefficients show almost perfect agreement [58] between clinical sleep staging and proposed algorithm's sleep staging.

Table 8 compares the performance of our proposed method for sleep stage classification using the Sleep-EDF dataset with recent state-of-the-art works in terms of overall accuracy and Cohen's kappa. The highest accuracy and Cohen's kappa are highlighted in bold, which show the performance

**TABLE 7. Classification results and normalized confusion matrix of TLCNN-DF on Sleep-EDFx dataset.**

		Wake	N1	N2	N3	REM
Expert scored	Wake	<b>0.989</b>	0.005	0.003	0.000	0.003
	N1	0.148	<b>0.364</b>	0.224	0.004	0.260
	N2	0.006	0.008	<b>0.923</b>	0.034	0.029
	N3	0.005	0.001	0.138	<b>0.856</b>	0.000
	REM	0.017	0.007	0.100	0.001	<b>0.875</b>
Specificity(%)	98.2	99.4	96.0	99.0	98.2	
F1 Score(%)	98.7	47	89.8	87.8	85.7	
Precision(%)	98.5	66.9	87.6	90.2	84.1	
Overall accuracy (%): 93.16, Cohen's kappa: 0.889						

of our proposed method. Similar results are observed in the TLCNN-DF performance for classification of Sleep-EDFx, as shown in Table 9. TLCNN-DF are yielded more promising results compared to state-of-the-art works in the classification of sleep stages on Sleep-EDF and Sleep-EDFx datasets.

Further comparison is made based on number of features that used in different algorithms which is given in Table 8 and Table 9. As can be seen, using comprehensive and more feature is an effective factor in achieving superior classification performance. On the other hand, by increasing the number of features, computational complexity for classifier increases. In the proposed method by extracting 59 features from EEG and EOG signals, the algorithm achieved higher accuracy compared to Hassan *et al.* [13] and Jiang *et al.* [59] studies which used higher number of features.

#### D. EXPERIMENT 4: PERFORMANCE COMPARISON BETWEEN THE PROPOSED ALGORITHM AND CONVENTIONAL CLASSIFICATION ALGORITHMS

In the classic data fusion methods, feature vectors from different information sources are integrated into a feature vector. Likewise, all the extracted features from the EEG and EOG signals in this study, are combined into one feature vector. The most commonly used classifiers in sleep studies including support vector machine (SVM), decision tree (DT), logistic regression (LR), linear discriminant analysis (LDA), k-nearest neighbor (KNN) and random forest (RF) are used

**TABLE 8. Performance comparison of proposed algorithm with state-of-the-art studies in 5-class sleep staging on Sleep-EDF dataset in terms of accuracy and Cohen's kappa. The number of utilized features in different studies is indicated in the last column.**

Study	Acc.(%)	kappa	No. of features
Hassan et al., 2015 [60]	86.5	0.787	10
Hsu et al., 2013 [61]	87.2	0.82	36
Zhu et al., 2014 [15]	88.9	0.83	9
Hassan et al., 2016 [12]	90.1	0.844	26
Hassan et al., 2016 [62]	90.7	0.855	16
Rahman et al., 2018 [14]	91.02	-	30
Sharma et al., 2017 [63]	91.1	0.861	39
Zhou et al., 2020 [64]	91.2	-	109
Yildirim et al., 2019 [65]	91.22	-	-
Hassan et al., 2016 [13]	91.5	0.865	97
Jiang et al., 2019 [59]	92.7	0.884	100
Proposed algorithm	<b>93.58</b>	<b>0.899</b>	59

**TABLE 9. Performance comparison between proposed algorithm and state-of-the-art studies in 5-class sleep staging on Sleep-EDFx dataset in terms of accuracy and Cohen's kappa. The number of utilized features in different studies is indicated in the last column.**

Study	Accuracy (%)	Cohen's kappa	No. of features
Zhou et al., 2020 [64]	82.4	-	109
Yildirim et al., 2019 [65]	90.98	-	-
Alickovic et al., 2018 [66]	91.1	0.879	41
Rahman et al., 2018 [14]	92.6	-	30
Jiang et al., 2019 [59]	92.6	0.882	100
Proposed algorithm	<b>93.16</b>	<b>0.889</b>	59

for the classification of integrated feature vectors. 4 Scenarios are considered for sleep staging in different number of classes, including 2-class (Wake and sleep), 3-class (Wake, REM and NREM), 4-class (Wake, REM, N1, N2+N3) and 5-class (Wake, REM, N1, N2, N3) cases. For a fair comparison between classifiers, the train data and the test data for evaluating classifiers are exactly the same. Hence the results are not influenced by using different training and testing epochs of signals for different classifiers.

Classification accuracy of different classifiers in different modes are shown in the Table 10. In the two-class mode (sleep and Wake), due to the fundamental differences in the characteristics of sleep and Wake stages and the simplicity of the

**TABLE 10. Performance of various classifiers on different number of sleep stages in term of accuracy.**

Classifier Hyper-parameter	2-Class	3-Class	4-Class	5-Class
SVM Kernel: RBF	97.35	91.73	87.8	87.46
DT Max depth: not limited	97.75	93.55	89.68	89.13
LR Solver: newton-cg	98.18	93.51	90.51	89.88
LDA Solver: least squares	98.01	93.05	90.04	90.37
KNN No. of trees = 10	98.67	95.86	93.78	93.05
RF No. of neighbors = 10	98.57	95.37	93.58	93.08
TLCNN-DF	<b>98.77</b>	<b>96.79</b>	<b>95.1</b>	<b>94.14</b>

classification problem, the classifiers have been able to obtain similar results, so that the lowest accuracy is 97.35% by SVM. As the number of classes increases and the classification problem becomes more complex, the performance of the other classifiers starts to decline compared to TLCNN-DF. In conventional classification methods, non-informative features have a significant effect on classification result. However, in the proposed method the entire structure of HVGs matters in image classification with CNN not only the features themselves. The overall classification accuracy obtained by TLCNN-DF for the different modes of classification is as follows: 2-class: 98.77, 3-class: 96.77, 4-class: 95.1 and 5-class: 94.14. Results show improvement in the classification accuracy and the robustness of the method for classifying different number of classes. Furthermore, the proposed algorithm is outperformed other conventional data fusion and classification methods.

#### IV. DISCUSSION AND FUTURE WORK

The main subjects that we have tried to challenge and improve in this study can be described as follows. Choosing hand-crafted features for extracting discriminant characteristics of EOG and EEG. Considering the numerous data fusion methods in different fields, this article proposes a novel method for feature level data fusion using HVGs. The popular architecture for fusing outputs of convolutional layers are also employed in our model's constructure [67], [68]. For instance GoogLeNet [69] used concatenation to fuse outputs of convolutional layers. Finally, in order to enhance described CNN-based fusion method, a novel transfer learning algorithm is proposed for the pre-training of convolutional layers of the model.

For feature extraction, we have tried to reduce the complexity of the problem by selecting simple features. The frequency domain features of the signal are the most useful hand-crafted features, which are used for sleep staging. Considering that

**TABLE 11. True and false recognition of stage N1 epochs by TLCNN-DF and CNN-DF.**

		Wake	N1	N2	N3	REM
N1	TLCNN-DF	12 10.4%	70 60.9%	17 14.8%	0 00.0%	16 13.9%
	CNN-DF	13 11.3%	53 46.0%	24 20.9%	0 00.0%	25 21.8%

EEG and EOG are active in different frequency bands for Wake, N2 and N3 stages. These sleep stages can be well separated by using the frequency characteristics of signal. Unlike most studies that extract specific frequency bands power, such as Delta (0 to 4 Hz), Theta (4 to 8 Hz) and Beta (15 to 32 Hz), in this work entire power spectrum used as a feature set. In this way, frequency information of signal is obtained with better resolution. Also, the EOG was decomposed by the wavelet transform into different frequency bands, which makes frequency information of EOG more prominent for feature extraction. On the other hand, N1 and REM stages have similar frequency behavior on EOG and EEG, therefore statistical and nonlinear features have been extracted from the signals to identify these sleep stages.

A well-known problem in sleep staging is the poor recognition of the N1 stage by algorithms. Also, in the current study, it was observed that the sensitivity of N1 stage is lower than other stages, and it is usually misclassified with other sleep stages, especially Wake, N2 and REM. As mentioned earlier, this could be due to the small number of data samples for N1 sleep stage for training the model. In fact, in the case of unbalanced data sets, the trained classifier is biased and as a result, the classifier cannot have promising performance. Therefore, the class with small number of training samples misclassified as other classes with more training samples [70]. On the other hand, choosing the appropriate classifier can also affect unbalance dataset problem. Table 11 shows the classified 30-sec test epochs of stage N1 for the TLCNN-DF and CNN-DF. The rows of the table are expert scored N1 epochs and the epochs on the columns are scored by classifiers. As mentioned in table 11, CNN-DF is able to correctly identify 46% of total N1 epochs, also N1 is incorrectly classified to Wake, N2 and REM by 11.3%, 20.9% and 21.8%, respectively. By using transfer learning for training the model, TLCNN-DF is correctly recognized 60% of N1 stage, and the false recognition is reduced to 10.4%, 14.8% and 13.9% for Wake, N2 and REM stages, respectively.

As shown in the Experiment 1, transfer learning has also improved overall accuracy, Cohen's kappa coefficients and per-stage classification performance in addition to improving detection of the stage N1. On the other hand the training time of the model is also reduced. The key idea of using transfer learning in this algorithm stems from the disadvantages of simultaneous training of the feature extraction part (convolutional layers) and the classification part (FC layers)

of CNNs. As is clear, the optimization algorithm of CNN tries to improve classification results by updating the weights of all layers at the same time. Due to the fact that the number of these weights is too many, the change in the weight of the convolutional layers and the FC layers at the same time may lead to sub-optimal result. As shown in Fig. 5 and Fig. 6 the accuracy and loss for CNN-DF has high tolerance in compare to TLCNN-DF. Besides, the deeper the model gets, the more complex and non-linear it becomes, and it takes more time to train. Thus, by separating the training process of the feature extraction and classification parts of the CNN, we can achieve an enhancement in the classification performance of the model.

Choosing sleep signals for automatic sleep stage classification has always been a challenge. According to sleep studies, the positive outcomes achieved in sleep staging with single channel EEG or EOG, even though the expert scored sleep stages still have a significant difference with algorithm scored sleep stages [13], [14], [65]. This has led us to try to propose a reliable method for fusing information of EEG and EOG for automatic sleep stage classification. The proposed fusion method was implemented in two stages, feature level fusion and CNN-based fusion. The feature level fusion is based on graph theory. All the features are transformed to HVGs in two feature sets. One feature set includes PSD features of EEG and the other feature set includes integrated EEG and EOG features. Constructing HVG from feature sets caused to comprehensive relation information between features, appears in the form of connections between graph nodes. Therefore, the classification is also affected by the relations between the features. With the introduced framework, it is possible to visualize HVGs in Euclidean space as well as the possibility of using CNN to classify HVGs.

In the following, we draw attention to the limitations of the proposed method and highlight the potential future work in this field. The feature mapping framework receives feature sets as input and after transforming feature sets to HVGs, constructs HVG images as output. As the number of nodes in each image increases, the complexity of the image increases and the need for deeper CNN with more complexity can be felt for classification. For future research, it is important to determine the maximum number of features in a feature set, which can be illustrated in an image, which does not have a negative impact on the performance of the CNN and the graph nodes do not overlap with neighboring nodes.

Despite improvements in stage N1 recognition using transfer learning, the obtained results are still not in perfect agreement with expert scoring results. Therefore, it is necessary to increase the sensitivity rate of the N1 stage by extracting the proper features, which indicate the unique characteristics of stage N1.

The process of mapping graphs into the Euclidean space and generating its images has been done in a simple way in our method. However, there are many different ways to draw a graph. Changes in methods for drawing graphs can affect the classifier performance. For instance, using grayscale or

binary images significantly decrease the training time of the model. However, its effect on the performance of the CNN can be assayed. In addition, the use of other algorithms to determine the graph connections instead of visibility algorithms or drawing graphs in other shapes instead of circular shape can be examined in future studies. Also, graph neural network (GNN) are a new type of neural networks, which have recently gained attention. GNNs take the graphs directly as input. Therefore, the GNN can be used for classification of HVGs without any requirement for mapping graphs into euclidean space.

So far sleep studies have used a specific type of database for the training and testing the classification algorithms, so that they will have difficulty in classifying datasets with different recording conditions. Therefore, providing a general and comprehensive algorithm to classify new data, accurately with one-time training is still an important challenge in automatic sleep staging. On the other hand, our proposed method has been tested on healthy subjects, so there may be limitations in classifying the sleep stages of subjects with sleep disorders. As a future work, we can evaluate the performance of the TLCNN-DF in subjects with sleep problems.

## V. CONCLUSION

In this paper, a novel algorithm is presented for sleep stage classification using EEG and EOG. This article introduces a new data fusion framework with transforming features into HVG, so that after extracting the hand-crafted features from the EEG and EOG, features are mapped into two image sets related to EEG features and fused EEG, EOG features and then, fed to TLCNN-DF with two specific pipelines for each image set. The TLCNN-DF is trained in two stages using transfer learning, in first stage convolutional layers are pre-trained in two separate subnetworks with two image sets and transferred to the main network. Then at the second stage, only the FC layers are trained. The experimental results indicated that the use of transfer learning in proposed CNN, increases the training speed and performance of the network. In addition, by comparing the TLCNN-DF with other state-of-the-art studies on classifying Sleep-EDF and Sleep-EDFx datasets, TLCNN-DF is able to improve the sleep staging in terms of accuracy and Cohen's kappa. Due to the obtained Cohen's kappa coefficient there is perfect agreement between the sleep expert and the proposed algorithm, which shows the proposed algorithm has the potential for practical use.

## REFERENCES

- [1] S. H. Sheldon, R. Ferber, and M. H. Kryger, *Principles and Practice of Pediatric Sleep Medicine*. Amsterdam, The Netherlands: Elsevier, 2005.
- [2] T. Lee-Chiong, *Sleep Medicine: Essentials and Review*. Oxford, U.K.: Oxford Univ. Press, 2008.
- [3] C. Iber and C. Iber, *The AASM Manual for the Scoring of Sleep and Associated Events: Rules, Terminology and Technical Specifications*, vol. 1. Westchester, IL, USA: American Academy of Sleep Medicine, 2007.
- [4] A. Kales and A. Rechtschaffen, *A Manual of Standardized Terminology, Techniques and Scoring System for Sleep Stages of Human Subjects*. Washington, DC, USA: United States Government Printing Office, 1968.

- [5] L. Novelli, R. Ferri, and O. Bruni, "Sleep classification according to AASM and rechtschaffen and kales: Effects on sleep scoring parameters of children and adolescents," *J. Sleep Res.*, vol. 19, no. 1p2, pp. 238–247, Mar. 2010.
- [6] C. Sun, J. Fan, C. Chen, W. Li, and W. Chen, "A two-stage neural network for sleep stage classification based on feature learning, sequence learning, and data augmentation," *IEEE Access*, vol. 7, pp. 109386–109397, 2019.
- [7] H. Phan, F. Andreotti, N. Cooray, O. Y. Chen, and M. De Vos, "Joint classification and prediction CNN framework for automatic sleep stage classification," *IEEE Trans. Biomed. Eng.*, vol. 66, no. 5, pp. 1285–1296, May 2019.
- [8] H. W. Agnew, W. B. Webb, and R. L. Williams, "The first night effect: An Eeg study of sleep," *Psychophysiology*, vol. 2, no. 3, pp. 263–266, Jan. 1966.
- [9] H. Dong, A. Supratak, W. Pan, C. Wu, P. M. Matthews, and Y. Guo, "Mixed neural network approach for temporal sleep stage classification," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 2, pp. 324–333, Feb. 2018.
- [10] P. Memar and F. Farajji, "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.
- [11] X. Li, L. Cui, S. Tao, J. Chen, X. Zhang, and G.-Q. Zhang, "HyCLASSS: A hybrid classifier for automatic sleep stage scoring," *IEEE J. Biomed. Health Informat.*, vol. 22, no. 2, pp. 375–385, Mar. 2018.
- [12] A. R. Hassan and M. I. Hassan Bhuiyan, "Automatic sleep scoring using statistical features in the EMD domain and ensemble methods," *Biocybernetics Biomed. Eng.*, vol. 36, no. 1, pp. 248–255, 2016.
- [13] A. R. Hassan and M. I. H. Bhuiyan, "A decision support system for automatic sleep staging from EEG signals using tunable Q-factor wavelet transform and spectral features," *J. Neurosci. Methods*, vol. 271, pp. 107–118, Sep. 2016.
- [14] M. M. Rahman, M. I. H. Bhuiyan, and A. R. Hassan, "Sleep stage classification using single-channel EOG," *Comput. Biol. Med.*, vol. 102, pp. 211–220, Nov. 2018.
- [15] G. Zhu, Y. Li, and P. Wen, "Analysis and classification of sleep stages based on difference visibility graphs from a single-channel EEG signal," *IEEE J. Biomed. Health Informat.*, vol. 18, no. 6, pp. 1813–1821, Nov. 2014.
- [16] Y. Cheng, "A rule-based automatic sleep staging method," Master Paper, Dept. Comput. Sci. Inf. Eng., Nat. Cheng Kung Univ., Tainan, Taiwan, 2009.
- [17] H. Phan, Q. Do, T.-L. Do, and D.-L. Vu, "Metric learning for automatic sleep stage classification," in *Proc. 35th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2013, pp. 5025–5028.
- [18] R. K. Tripathy and U. Rajendra Acharya, "Use of features from RR-time series and EEG signals for automated classification of sleep stages in deep neural network framework," *Biocybernetics Biomed. Eng.*, vol. 38, no. 4, pp. 890–902, 2018.
- [19] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [20] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [21] H. Phan, F. Andreotti, N. Cooray, O. Y. Chen, and M. D. Vos, "Automatic sleep stage classification using single-channel EEG: Learning sequential features with attention-based recurrent neural networks," in *Proc. 40th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2018, pp. 1452–1455.
- [22] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.
- [23] M. Liang and X. Hu, "Recurrent convolutional neural network for object recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3367–3375.
- [24] K. Murata, M. Mito, D. Eguchi, Y. Mori, and M. Toyonaga, "A single filter CNN performance for basic shape classification," in *Proc. 9th Int. Conf. Awareness Sci. Technol. (iCAST)*, Sep. 2018, pp. 139–143.
- [25] Q. Yu, Y. Yang, F. Liu, Y.-Z. Song, T. Xiang, and T. M. Hospedales, "Sketch-a-net: A deep neural network that beats humans," *Int. J. Comput. Vis.*, vol. 122, no. 3, pp. 411–425, May 2017.
- [26] 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.
- [27] H. Park, K. Park, and D.-U. Jeong, "Hybrid neural-network and rule-based expert system for automatic sleep stage scoring," in *Proc. 22nd Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, vol. 2, Jul. 2000, pp. 1316–1319.

- [28] B. Yang and H. Liu, "Automatic identification of insomnia based on single-channel EEG labelled with sleep stage annotations," *IEEE Access*, vol. 8, pp. 104281–104291, 2020.
- [29] J. Shi, X. Liu, Y. Li, Q. Zhang, Y. Li, and S. Ying, "Multi-channel EEG-based sleep stage classification with joint collaborative representation and multiple kernel learning," *J. Neurosci. Methods*, vol. 254, pp. 94–101, Oct. 2015.
- [30] F. Saccomandi, L. Priano, A. Mauro, R. Nerino, and C. Guiot, "Automatic detection of transient EEG events during sleep can be improved using a multi-channel approach," *Clin. Neurophysiol.*, vol. 119, no. 4, pp. 959–967, Apr. 2008.
- [31] D. L. Hall and J. Llinas, "An introduction to multisensor data fusion," *Proc. IEEE*, vol. 85, no. 1, pp. 6–23, Jan. 1997.
- [32] M. Abdollahpour, T. Y. Rezaei, A. Farzamina, and S. Meshgini, "Sleep stage classification using dempster-shafer theory for classifier fusion," in *Proc. IEEE Int. Conf. Artif. Intell. Eng. Technol. (IICAIET)*, Nov. 2018, pp. 1–4.
- [33] C. Chen, X. Liu, A. Ugon, X. Zhang, A. Amara, P. Garda, J.-G. Ganascia, C. Philippe, and A. Pinna, "Symbolic fusion: A novel decision support algorithm for sleep staging application," in *Proc. 5th EAI Int. Conf. Wireless Mobile Commun. Healthcare, Transforming Healthcare Through Innov. Mobile Wireless Technol.*, 2015, pp. 19–22.
- [34] J. Zhang, Y. Wu, J. Bai, and F. Chen, "Automatic sleep stage classification based on sparse deep belief net and combination of multiple classifiers," *Trans. Inst. Meas. Control*, vol. 38, no. 4, pp. 435–451, Apr. 2016.
- [35] S. Chambon, M. N. Galtier, P. J. Arnal, G. Wainrib, and A. Gramfort, "A deep learning architecture for temporal sleep stage classification using multivariate and multimodal time series," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 4, pp. 758–769, Apr. 2018.
- [36] H. F. Durrant-Whyte, "Sensor models and multisensor integration," in *Autonomous Robot Vehicles*. New York, NY, USA: Springer, 1990, pp. 73–89.
- [37] C. Chu, A.-L. Hsu, K.-H. Chou, P. Bandettini, and C. Lin, "Does feature selection improve classification accuracy? Impact of sample size and feature selection on classification using anatomical magnetic resonance images," *NeuroImage*, vol. 60, no. 1, pp. 59–70, Mar. 2012.
- [38] O. Kwon and J. M. Sim, "Effects of data set features on the performances of classification algorithms," *Expert Syst. Appl.*, vol. 40, no. 5, pp. 1847–1857, Apr. 2013.
- [39] B. Kemp, A. H. Zwinderman, B. Tuk, H. A. C. Kamphuisen, and J. J. L. Obery, "Analysis of a sleep-dependent neuronal feedback loop: The slow-wave microcontinuity of the EEG," *IEEE Trans. Biomed. Eng.*, vol. 47, no. 9, pp. 1185–1194, Sep. 2000.
- [40] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, Jun. 2000.
- [41] S.-T. Pan, C.-E. Kuo, J.-H. Zeng, and S.-F. Liang, "A transition-constrained discrete hidden Markov model for automatic sleep staging," *Biomed. Eng. OnLine*, vol. 11, no. 1, p. 52, 2012.
- [42] D. Y. Kang, P. N. DeYoung, A. Malhotra, R. L. Owens, and T. P. Coleman, "A state space and density estimation framework for sleep staging in obstructive sleep apnea," *IEEE Trans. Biomed. Eng.*, vol. 65, no. 6, pp. 1201–1212, Jun. 2018.
- [43] P. Welch, "The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms," *IEEE Trans. Audio Electroacoustics*, vol. AE-15, no. 2, pp. 70–73, Jun. 1967.
- [44] A. N. Kolmogorov, "Three approaches to the quantitative definition of information," *Int. J. Comput. Math.*, vol. 2, nos. 1–4, pp. 157–168, 1968.
- [45] F. Kaspar and H. Schuster, "Easily calculable measure for the complexity of spatiotemporal patterns," *Phys. Rev. A, Gen. Phys.*, vol. 36, no. 2, p. 842, 1987.
- [46] Q. Cai, Z. Gao, J. An, S. Gao, and C. Grebogi, "A graph-temporal fused dual-input convolutional neural network for detecting sleep stages from EEG signals," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, early access, Aug. 5, 2020, doi: 10.1109/TCSII.2020.3014514.
- [47] S. Abdulla, M. Diykh, R. L. Laft, K. Saleh, and R. C. Deo, "Sleep EEG signal analysis based on correlation graph similarity coupled with an ensemble extreme machine learning algorithm," *Expert Syst. Appl.*, vol. 138, Dec. 2019, Art. no. 112790.
- [48] B. Luque, L. Lacasa, F. Ballesteros, and J. Luque, "Horizontal visibility graphs: Exact results for random time series," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 80, no. 4, Oct. 2009, Art. no. 046103.
- [49] L. Lacasa, B. Luque, F. Ballesteros, J. Luque, and J. C. Nuño, "From time series to complex networks: The visibility graph," *Proc. Nat. Acad. Sci. USA*, vol. 105, no. 13, pp. 4972–4975, Apr. 2008.
- [50] A. J.-P. Tixier, G. Nikolentzos, P. Meladianos, and M. Vazirgiannis, "Graph classification with 2D convolutional neural networks," in *Proc. Int. Conf. Artif. Neural Netw. Munich, Germany: Springer*, 2019, pp. 578–593.
- [51] G. Nikolentzos, P. Meladianos, and M. Vazirgiannis, "Matching node embeddings for graph similarity," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 2429–2435.
- [52] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time series," in *The Handbook of Brain Theory and Neural Networks*, vol. 3361, no. 10. Cambridge, MA, USA: MIT Press, 1995, p. 1995.
- [53] A. Hagberg, D. A. Schult, and P. J. Swart, "Exploring network structure dynamics and function using NetworkX," in *Proc. 7th Python Sci. Conf. (SciPy)*, Pasadena, CA, USA, Aug. 2008, pp. 11–15.
- [54] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
- [55] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
- [56] L. Torrey and J. Shavlik, "Transfer learning," in *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*. Hershey, PA, USA: IGI Global, 2010, pp. 242–264.
- [57] J. Cohen, "A coefficient of agreement for nominal scales," *Educ. Psychol. Meas.*, vol. 20, no. 1, pp. 37–46, Apr. 1960.
- [58] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 33, pp. 159–174, Mar. 1977.
- [59] D. Jiang, Y.-N. Lu, Y. Ma, and Y. Wang, "Robust sleep stage classification with single-channel EEG signals using multimodal decomposition and HMM-based refinement," *Expert Syst. Appl.*, vol. 121, pp. 188–203, May 2019.
- [60] A. R. Hassan, S. K. Bashar, and M. I. H. Bhuiyan, "On the classification of sleep states by means of statistical and spectral features from single channel electroencephalogram," in *Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI)*, Aug. 2015, pp. 2238–2243.
- [61] Y.-L. Hsu, Y.-T. Yang, J.-S. Wang, and C.-Y. Hsu, "Automatic sleep stage recurrent neural classifier using energy features of EEG signals," *Neurocomputing*, vol. 104, pp. 105–114, Mar. 2013.
- [62] A. R. Hassan and M. I. H. Bhuiyan, "Computer-aided sleep staging using complete ensemble empirical mode decomposition with adaptive noise and bootstrap aggregating," *Biomed. Signal Process. Control*, vol. 24, pp. 1–10, Feb. 2016.
- [63] 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.
- [64] J. Zhou, G. Wang, J. Liu, D. Wu, W. Xu, Z. Wang, J. Ye, M. Xia, Y. Hu, and Y. Tian, "Automatic sleep stage classification with single channel EEG signal based on two-layer stacked ensemble model," *IEEE Access*, vol. 8, pp. 57283–57297, 2020.
- [65] O. Yildirim, U. Baloglu, and U. Acharya, "A deep learning model for automated sleep stages classification using PSG signals," *Int. J. Environ. Res. Public Health*, vol. 16, no. 4, p. 599, Feb. 2019.
- [66] E. Alickovic and A. Subasi, "Ensemble SVM method for automatic sleep stage classification," *IEEE Trans. Instrum. Meas.*, vol. 67, no. 6, pp. 1258–1265, Jun. 2018.
- [67] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Noguees, J. Yao, D. Mollura, and R. M. Summers, "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1285–1298, May 2016.
- [68] L. D. Nguyen, D. Lin, Z. Lin, and J. Cao, "Deep CNNs for microscopic image classification by exploiting transfer learning and feature concatenation," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, 2018, pp. 1–5.
- [69] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.
- [70] S. Vucetic and Z. Obradovic, "Classification on data with biased class distribution," in *Proc. Eur. Conf. Mach. Learn.* Berlin, Germany: Springer, 2001, pp. 527–538.



**MEHDI ABDOLLAHOUPUR** received the B.S. degree in electrical engineering and the M.S. degree in biomedical engineering (bioelectric) from the University of Tabriz, Tabriz, Iran. His research interests include deep learning, signal processing, data fusion, and graph theory.



**ALI FARZAMNIA** (Senior Member, IEEE) received the B.Eng. degree in electrical engineering (telecommunication engineering) from Islamic Azad University, Urmia, Iran, in 2005, the M.Sc. degree in electrical engineering (telecommunication engineering) from the University of Tabriz, in 2008, and the Ph.D. degree in electrical engineering (telecommunication engineering) from Universiti Teknologi Malaysia (UTM), in 2014. He has been appointed as a Senior Lecturer (Assistant Professor) with the Electrical and Electronic Engineering Program, Faculty of Engineering, Universiti Malaysia Sabah (UMS), since 2014. His research interests include wireless communication, signal processing, network coding, information theory, and bio-medical signal processing. He is also a Chartered Engineer (CEng), U.K., and a member of IET.



**ISMAIL SAAD** (Member, IEEE) received the B.Eng. degree from Universiti Putra Malaysia (UPM), in 1999, the master's degree from the University of Southampton, in 2001, and the Ph.D. degree from Universiti Teknologi Malaysia (UTM), in 2009. He has been an Associate Professor with the Electrical and Electronics Engineering Program, Faculty of Engineering, Universiti Malaysia Sabah (UMS), since 2013. He has published more than 100 Scopus index papers as primary and corresponding authors. His research interests include micro and nanoelectronics device and materials, digital integrated circuit system design, and nanofiber membrane for energy applications. He has been a Graduate Member of the Board of Engineers Malaysia (BEM) and the Institute of Engineers Malaysia (IEM), since 2010. During his Ph.D. Convocation Ceremony, he was awarded as the Chancellor and the Best Student Award. He is passionate about research and received awards in Malaysia Technology Expo (MTE), Bio-Innovation, and UMS Research and Innovation (PEREKA) Research Competition from 2011 to 2016. He has secured over a million research grants with numerous dedicated collaborative research partners. In administration experience, he proves an active involvement with holding a position as the Head of the Artificial Intelligence Research Unit, the Program Head of Electrical and Electronics Engineering, the Deputy Dean of Academic and Internationalization, and currently as the Dean with the Faculty of Engineering, Universiti Malaysia Sabah (UMS). He was appointed as the First Chair of the IEEE Sabah Subsection in 2017.



**TOHID YOUSEFI REZAII** received the B.Sc., M.Sc., and Ph.D. degrees in electrical engineering (telecommunication) from the University of Tabriz, Tabriz, Iran, in 2006, 2008, and 2012, respectively. He is currently with the Faculty of Electrical and Computer Engineering, University of Tabriz. His current research interests include biomedical signal processing, data compression, compressed sensing, statistical signal processing, pattern recognition statistical learning, and adaptive filters.

• • •