Transferable Self-Supervised Instance Learning for Sleep Recognition

Aite Zhao, Yue Wang, and Jianbo Li

***Abstract*—Although the importance of sleep is increasingly recognized, the lack of general and transferable algorithms hinders scalable sleep assessment in healthy persons and those with sleep disorders. A deep understanding of the sleep posture, state, or stage is the premise of diagnosing and treating sleep diseases. At present, most existing methods draw support from supervised learning to monitor the whole sleep process. However, in the absence of sufficient labeled sleep data, it is difficult to guarantee the reliability of sleep recognition networks. To solve this problem, we propose a transferable self-supervised instance learning model for three sleep recognition tasks, i.e., sleep posture, state, and stage recognition. Firstly, a SleepGAN is designed to generate sleep data, and then, we combine enough self-supervised rotating sleep data and original data for non-parametric classification at the instance-level, finally, different sleep postures, states, or stages can be distinguished precisely. The proposed model can be applied to multimodal sleep data such as signals and images, and makeup for the inaccuracy caused by insufficient data, and can be transferred to sleep datasets of different sizes. The experimental results show that our algorithm for the physiological changes in the sleep process is superior to several state-of-the-art studies, which may be helpful to promote the intelligence of sleep assessment and monitoring.**

***Index Terms*—Sleep recognition, sleep diseases, multimodal data, SleepGAN, self-supervised learning, instance learning.**

I. Introduction

SLEEP is one of the major activities of daily living and occupies one-third of our life, which affects people's lives but remains mysterious in many ways [1], [2]. Sleep can be assessed using physical or physiological parameters, such as respiration rate, heart rate, temperature, and body movement [3], [4]. To preferably understand and evaluate the sleep process, we are committed to building a reliable sleep recognition model for sleep monitoring and auxiliary diagnosis.

As the key of sleep monitoring and treatment, sleep posture and stage in a sleep process are widely concerned and can be captured by multiple sensors. sleep posture is a prevalent issue among elderly and may cause pressure injuries (PI) if they have prolonged sleep in a single posture without moving. PI may result in constant pain, loss of mobility, depression, and even death. Studies have found that sleep issues are more prevalent within the residential care population [5], [6]. Sleep postures include the left recumbent position, right recumbent position and supine position, which can provide a dependable reference for the correct choice of the sleep posture. Clinical evidence shows that the recognition of sleep posture can be utilized as a diagnostic index of a series of chronic diseases and an auxiliary means for the treatment of diseases. Additionally, sleep stages can describe the depth of sleep, is also one of the most critical steps in effective diagnosis and treatment of sleep-related disorders. Sleep stages, i.e., the non-rapid eye movement sleep (NREMs) and rapid eye movement sleep (REMs), are recognized to locate deep and shallow sleep, also provide the basis for the treatment of sleep disorders [7], [8]. Moreover, we implement insomnia detection task, which is also the reference basis for assisting doctors in the diagnosis of insomnia [9]. Three sleep recognition tasks are shown in Fig. 1.

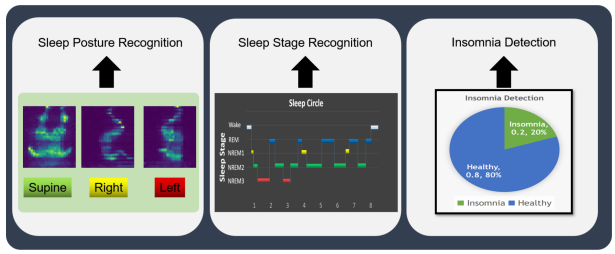


Fig. 1. Three tasks of sleep recognition, i.e., sleep posture recognition, sleep stage recognition, and insomnia detection. The process of sleep posture recognition is to use the image data collected by the pressure mattress for sleep monitoring to distinguish three different sleep postures, i.e., supine, left and right lying. Sleep stage recognition is to divide the subjects into different stages of sleep at night, including wake, REM, NREM 1, NREM 2 and NREM 3. Different colors in the figure represent different stages. Moreover, we also classify insomniacs and healthy subjects in order to detect insomniacs from input samples.

The data collection approaches of sleep stage and posture are also diverse, the most common method is to use the biological signals detected by polysomnography to describe the sleep state. However, due to the patients generally need to sleep in the laboratory, more electrodes are installed, which is easy to interfere with the sleep of patients, especially for mild patients, sometimes the examination fails due to poor sleep [10]. Recently, some non-invasive monitoring machines, such as wrist wearable devices, mobile phones, biological radar, etc., have made up for the defects of polysomnography. The fusion of these multimodal sleep data can make a more comprehensive sleep process.

Manuscript received 18 May 2021; revised 18 November 2021 and 10 April 2022; accepted 17 May 2022. Date of publication 23 May 2022; date of current version 20 October 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62106117, and in part by the Natural Science Foundation of Shandong Province under Grant ZR2021QF084. The Associate Editor coordinating the review of this manuscript and approving it for publication was Dr. S.-C. S. Cheung. *(Corresponding author: Aite Zhao.)*

The authors are with the College of Computer Science and Technology, Qingdao University, qingdao 266071, China (e-mail: zhaoaite@qdu.edu.cn; 2020025817@qdu.edu.cn; lijianbo@qdu.edu.cn).

Digital Object Identifier 10.1109/TMM.2022.3176751

In the field of sleep recognition, a large number of classification methods have been proposed to facilitate the automatic analysis of sleep process. Decision trees [11], [12], k-nearest neighbour (KNN) [13], [14], support vector machines (SVM) [15]–[17], convolutional neural network (CNN) [18]–[21], long short-term memory (LSTM) [22]–[24] are devoted to operating sleep biometrics, which provides an algorithmic basis for sleep assessment automation.

Although the above methods are applied to the single-modal sleep data respectively, it remains to be verified whether the above methods can be applied to multimodal sleep data (images and signals) at the same time. Besides, most of the previous studies use a certain number of labeled samples, only using supervised learning methods, which has weak recognition ability for new unlabelled data. Insufficient or unbalanced samples will also lead to the failure of these methods. Moreover, these methods rarely consider the differences and similarities between individuals or classes, which will lose some feature details.

In order to cover these shortcomings, we propose a transferable self-supervised instance learning model (SSIL). It is successfully applied to various types of datasets, expands the data capacity of multimodal sleep data, and improves data diversity. It also analyzes and processes the similarities and differences between samples and between classes. Compared with the previous supervised models, the proposed self-supervised model uses rotation data as labels for training, and can better model the relationship between different sleep samples.

The proposed model is divided into two layers, i.e., data generator and data analyzer. The data generator is referred as SleepGAN (sleep generative adversarial network) for sleep data generation, including a discriminator and a generator. The discriminator embeds a time-sensitive LSTM network, the generator employs a CNN-based network to generate sleep data and suffixes a joint loss function for collaborative training. The data analyzer includes the self-supervised instance learning algorithm to train the original data, the newly generated data, and their rotation self-supervised data and determines the final class based on the difference of the learned features.

The main contributions of this paper are summarized as follows:

- A sleep generative adversarial network (SleepGAN) is designed for sleep data generation.
- The proposed method uses the rotating image of sleep data as the label of self-supervised learning, which reduces the cost of labeling and ensures the diversity of data.
- In order to expand the data capacity and better describe the difference and similarity of sleep data at instance-level, self-supervised instance learning is proposed to learn a good feature representation, which captures the obvious similarity between instances, instead of classes, and merely asks features to be discriminative of individual instances.
- The proposed method can successfully apply to multimodal sleep data and accomplish three kinds of sleep recognition tasks, which outperforms other methods in literature.

In the rest of this article, Section II describes the related work; Section III describes our proposed approach for data generation and sleep recognition; Section IV shows the obtained experimental results and at last, we wrap up our study with the conclusion in Section V.

II. Related Work

In the literature, various aspects of sleep have been extensively explored. Based on the classification task of sleep recognition, many approaches have been proposed in the literature to identify sleep stages and postures to evaluate sleep quality and monitor abnormal changes in the sleep cycle.

A. Supervised Learning Methods on Sleep Recognition

As far as the sleep recognition method is concerned, the advantage of the supervised learning method is that it can analyze the sleep patterns of labeled data, calculate the differences between classes according to the classified information, and accurately predict the correct labels of test data.

1) Sleep Posture Recognition: For sleep posture recognition, some neural networks and learning classifiers are often used to correct and monitor subjects' sleep postures [25]–[30]. For example, a matching-based approach was proposed for sleep posture recognition to treat sleep pressure images as weighted 2D shapes and compute Euclidean distance for similarity measure [25]. Moein *et al.* employed a simple neural network to classify different sleep postures, which investigated the use of four hydraulic bed transducers placed underneath the mattress [26]. A deep multi-stream convolutional neural network was adopted for sleep posture classification using preprocessed depth images [27]. In addition, the traditional automatic learning methods 1D Fast Fourier transforms and 2D Gabor filter are also used for feature extraction in sleep posture recognition [28].

2) Sleep Stage Recognition: For sleep stage recognition, several deep models are trained to learn the sleep pattern [31]–[34], for example, a joint classification-and-prediction framework was proposed based on convolutional neural networks (CNNs) for automatic sleep staging by using polysomnography (PSG) signals, which leveraged the dependency among consecutive sleep epochs while surpassing the problems experienced with the common classification schemes [31]. In the literature [32], a rule-free refinement process based on a hidden Markov model (HMM) was proposed to optimize the sleep stage classification results automatically by using single-channel electroencephalography (EEG) records.

Various machine learning methods can also solve the problem of sleep stage scoring [35]–[37], for example, a k-means clustering-based feature weighting has been proposed and combined with the k-nearest neighbor and decision tree classifiers to classify the EEG sleep into six sleep stages including awake, non-rapid eye movement (NREM) stage 1, NREM stage 2, NREM stage 3, REM, and non-sleep (movement time), which can automatically score the sleep stages and help to sleep physicians on sleep stage scoring [35]. In addition, a fuzzy classifier and a genetic algorithm (GA) were designed for sleep scoring using a single EEG signal that detected differences in spectral features [36].

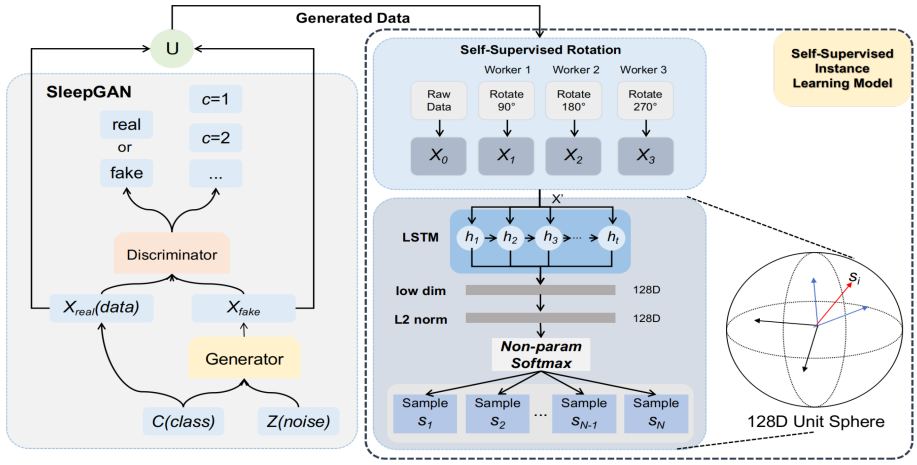


Fig. 2. The pipeline of our self-supervised instance learning approach. We design a SleepGAN to generate sleep images for data expansion of small datasets. The self-supervised rotations is used to enrich the characteristics of each sleep posture and to increase the data, which are fed into an instance learning model that encodes each image as a feature vector that is projected to a 128-dimensional space and L_2 normalized. The optimal feature embedding is learned via non-parametric discrimination, which tries to maximally scatter the features of training samples over the 128-dimensional unit sphere.

B. Unsupervised Learning Methods on Sleep Recognition

Although supervised learning can fully learn the salient features of labeled data, if there is not enough prior knowledge, it is difficult to label the sleep data manually, or the labeled data is too few, it will have a serious impact on the performance of the method. Unsupervised learning represents a more advanced learning mode, which can generate enough simulation training data and analyze different kinds of unlabeled new data through coding and decoding, the definition of the loss function, and objective function without human intervention. Self-supervised learning is a special case of unsupervised learning.

For instance, a self-supervised learning model was proposed for sleep recognition using multimodal sleep data [38], a new unsupervised complex-valued convolutional neural network was proposed for sleep stage classification using electroencephalogram [39]. Moreover, unsupervised learning algorithms and domain knowledge heuristics were combined to develop reliable sleep/wake state prediction models using unlabeled wrist actigraphy data [40]. Additionally, a sleep apnea (SA) detection model was proposed based on frequential stacked sparse autoencoder (FSSAE) and time-dependent cost-sensitive (TDCS) classification model [41].

Inspired by the mentioned studies, we proposed a hybrid self-supervised model to train the generated sleep data with no parameter discrimination, which can achieve sleep stage and posture recognition.

III. Transferable Self-Supervised Instance Learning for Sleep Recognition

The proposed self-supervised model consists of three parts, SleepGAN, self-supervised rotation and instance learning process. SleepGAN is mainly designed to make up for insufficient sample sleep data sets and enhance the learning ability of deep learning. Self-supervised rotation is used to expand the feature representation of data under different angles. Compared with the above two parts, instance learning uses the differences and distinctive features between individuals to describe the similarity of similar individuals and the differences of heterogeneous individuals. The three parts complement each other, and different parts are selected according to the complexity of the sleep dataset, which makes the experimental results exceed the recognition effect of several supervised learning algorithms and unsupervised learning algorithms. The framework of the entire model is shown in Fig. 2.

Formally, given a sleep sequence X, we divide it into K segments $X_k = \{X_1, X_2, \ldots, X_K\}$ of equal durations, $X_k \in \mathbb{R}^{D\times T}$, D is the feature dimension of each sample, and T is the sample number in one segment. The label sequence corresponding to the sequence data is $L_k = \{L_1, L_2, \ldots, L_K\}$.

A. Sleepgan

The generative adversarial network (GAN) [42] is very flexible and can train any kind of generator network. Most other frameworks require the generator network to have some specific functional forms, such as Gaussian discriminant analysis and naive Bayes [43], [44]. There is no need to design a model that follows any kind of factorization, any generators and discriminators will be useful. There is no need to use Markov chain to sample repeatedly and infer in the learning process, which avoids the difficult problem of approximate calculation of difficult probability.

A GAN [42] is mainly composed of two networks: the generator G and the discriminator D, trained in opposition to one another. The generative model G is to package a random noise z into a fake sample $X_{fake} = G(z)$, while the discriminant model D needs to judge whether the incoming sample (a training sample or a synthesized sample) is true or false, and output a probability distribution $P(S|X) = D(X)$. With the continuous improvement of discriminator's ability to identify samples, the

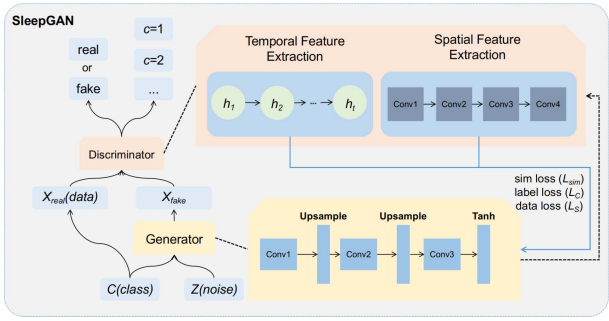


Fig. 3. The internal structure of the SleepGAN. SleepGAN is composed of a generator and a discriminator. The generator is an up-sampling deconvolutional neural network. The discriminator includes LSTM and CNN components that extract spatiotemporal features for classification.

forgery ability of generator G is also improving. The discriminator is trained to maximize the log-likelihood the first term, while the generator is trained to minimize the second term in 1.

$$L_S = E[logP(S = real|X_{real})] + E[logP(S = fake|X_{fake})] \quad (1)$$

A variant of GAN, i.e., the auxiliary classifier GAN (ACGAN) [45], every generated sample has a corresponding class label, $c \sim p_c$ in addition to the noise z. G uses both to generate images $X_{fake} = G(c, z)$. The discriminator gives both a probability distribution over sources and a probability distribution over the class labels, $P(S|X), P(C|X) = D(X)$. The objective function has two parts: the log-likelihood of the correct source, L_S (1), and the log-likelihood of the correct class, L_C (2).

$$L_C = E[logP(C = c|X_{real})] + E[logP(C = c|X_{fake})] \quad (2)$$

D is trained to maximize $L_C + L_S$ while G is trained to maximize $L_C - L_S$. ACGAN learn a representation for z that is independent of class label.

Under the consideration of the structure of ACGAN, the designed SleepGAN also consists of two parts, a generator G and a discriminator D. The discriminator is composed of an LSTM module and a CNN module, which uses the spatio-temporal features of sleep data as the main basis to distinguish its classification; the generator is a simple up-sampling deconvolutional module, due to the weak generation ability of LSTM. The data output by the generator is input into the discriminator in the form of sequence and compared with the original data, which has temporal and spatial features. The two models in the discriminator are specifically set for the output of the generator. The discriminator extracts the temporal and spatial features of the comparative data at the same time, which has an excellent effect. The structure is demonstrated in Fig. 3.

The objective function in CNN module is as follows:

$$\begin{cases} l_1 = \sigma(W_1 X_k + b_1) \\ l_2 = \sigma(W_2 l_1 + b_2) \\ \quad \cdots \\ l_D = \sigma(W_D l_{D-1} + b_D) \end{cases} . \quad (3)$$

where $l_1, l_2, \ldots, l_D$ are the output of each layer in CNN, W denotes the shared weight of neuron, σ is activation function. The l_D is obtained after the convolutional operation.

The objective function in LSTM module is as follows:

$$\begin{cases} f_t = \sigma(W_f \cdot [h_{t-1}, X_t]), \ i_t = \sigma(W_i \cdot [h_{t-1}, X_t]) \\ \tilde{c}_t = \sigma(W_c \cdot [h_{t-1}, X_t]), \ o_t = \sigma(W_o \cdot [h_{t-1}, X_t]) \\ c_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t \\ h_t = o_t \cdot \sigma_h(c_t) \end{cases} . \quad (4)$$

where X_t and h_{t-1} are the current and previous outputs in the LSTM node. σ is the activation function. f_t, i_t, and o_t denote the forget, input, and output gates at time step $t \in \{1, 2, 3 \ldots, T\}$. h_t denotes the last output of each node, the output h_T of the last LSTM node is selected as the final feature of this sequence.

Based on the outputs h_T and l_D of the two modules, the joint loss function is represented in 5.

$$\begin{aligned} \tilde{L}_C = ((&E[logP(C = c_{cnn}|X_{real})] \\ &+ E[logP(C = c_{cnn}|X_{fake})]) \\ &+ (E[logP(C = c_{lstm}|X_{real})] \\ &+ E[logP(C = c_{lstm}|X_{fake})]))/2 \end{aligned} \quad (5)$$

In order to avoid the discriminant inconsistency between LSTM and CNN, we also maximize the similarity of the output features of the CNN and LSTM modules:

$$L_{sim} = \frac{h_T \cdot l_D}{||h_T||||l_D||} \quad (6)$$

SleepGAN includes two loss functions, the loss function $\tilde{L}_C - L_S$ of the generator and the loss function $\tilde{L}_C + L_S + L_{sim}$ of the discriminator, which are maximized at the same time to optimize the effect of the hybrid model. After obtaining the data generated by SleepGAN, we incorporate it into the original data for self-supervised instance learning.

B. Self-Supervised Instance Learning Model

Manual data annotation is an essential step in supervised learning, which is time-consuming, laborious and noisy. Unlike supervised methods, unsupervised methods do not rely on human annotations and usually focus on pre-defined priors for a good representation of the data (such as smoothness, sparsity, and decomposition). The self-supervised method can be regarded as a special form of unsupervised learning method with a form of supervision, where the supervision is induced by self-supervised tasks rather than preset prior knowledge. In contrast to a completely unsupervised settings, self-supervised learning extracts information from the dataset itself to construct pseudo-labels. Our goal is to increase the multi-dimensional representation of the sleep data through the self-supervised rotation method, and combine the unsupervised instance learning method to calculate the salient features of each training instance, seamlessly connecting the two unsupervised methods.

For multi-dimensional feature representation and data augmentation, e.g., rotation or cropping, which systematically enlarge the training dataset by explicitly generating more training samples, have been popularly used to improve the generalization performance of deep neural networks [46]. With the trained

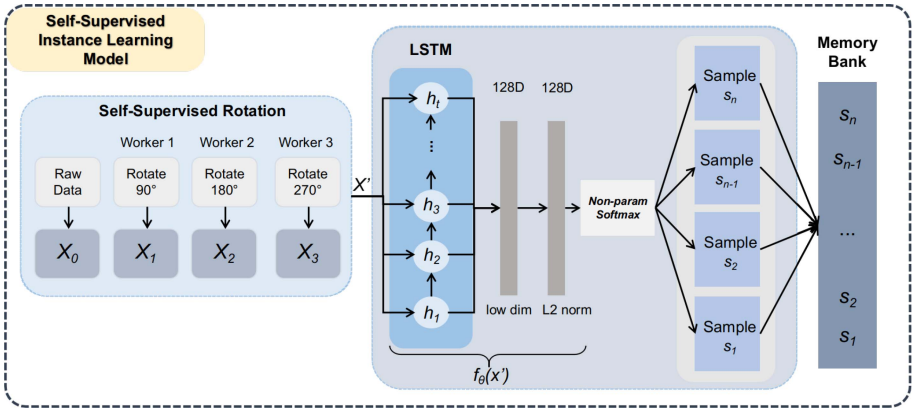


Fig. 4. Self-supervised instance learning model. After self-supervised transformation from different angles, the generated sleep data and pseudo-labels are fed into the feature extractor LSTM, and the learned 128-dimensional feature vector is stored in the memory bank through the non-parametric softmax discriminator, and is updated through continuous until the loss is minimal.

instance learning model, more obvious global features between different classes and individual local features can be obtained. The self-supervised instance learning model is demonstrated in Fig. 4.

Rotation Self-Supervision: By training the instance learning model, we can recognize four rotation degrees (0°, 90°, 180°, 270°) applied to its input image [47]. It successfully forces the instance learning model trained on it to learn semantic features useful for various visual perception tasks, such as object recognition, object detection and object segmentation. Let X be an input and the output matrix of rotation is X'.

Non-Parametric Instance Learning: An objective function $s = f_\theta(X')$ without supervision is the goal for instance learning. f_θ is an LSTM with parameters θ, mapping input data X' to feature s. This mapping would induces a metric over the data space, as $d_\theta(x,' y') = ||f_\theta(x') - f_\theta(y')||$ for instances x' and y', which should map visually similar data closer. The unsupervised feature learning approach is instance-level discrimination, which treats each sample instance as a distinct class of its own and train a classifier to distinguish between individual instance classes.

The instance-level classification is formulated using the softmax criterion. Suppose we have n samples $\{x'_1, x'_2, \ldots, x'_n\}$ in n classes and their features $s_1, s_2, \ldots, s_n$ with $s_i = f_\theta(x'_i)$. Under the conventional parametric softmax formulation, for instance x' with feature $s = f_\theta(x')$, the parametric softmax formulation with the weight vector w is modified to a non-parametric variant, we use s itself as the weight, and enforce $||s|| = 1$ via a L_2-normalization layer. Then the probability $P(i|s)$ becomes:

$$P(i|s) = \frac{exp(s_i^T s/\tau)}{\sum_{j=1}^{n} exp(s_j^T s/\tau)} \tag{7}$$

where τ is a temperature parameter that controls the concentration level of the distribution, and also necessary for tuning the concentration of s on our unit sphere. The joint probability $\prod_{i=1}^{n} P_\theta(i|f_\theta(x_i))$ is then maximized over the training set:

$$J(\theta) = \sum_{i=1}^{n} logP(i|f_\theta(x'_i)) \tag{8}$$

Instead of exhaustively computing s_j in 7 every time, a feature memory bank S is used for storing. Let $S = \{s_j\}$ be the memory bank and $f_i = f_\theta(x_i)$ be the feature of x_i. Then f_i is updated to S at the corresponding instance entry.

Noise-Contrastive Estimation: When the number of classes n is very large, for example, calculating the non-parametric softmax in the 7 is expensive. Popular techniques to reduce computation include hierarchical softmax [48], noise contrast estimation (NCE) [49] and negative sampling [50]. NCE is adapted to to solve the difficulty of calculating the similarity of all instances in the training set. The basic idea is to transform a multi-class classification problem into a set of binary classification problems, where the binary classification task is to distinguish between data samples and noise samples. Specifically, under our model, the probability that the feature representation s in the memory bank corresponds to the i-th example is:

$$\begin{cases} P(i|s) = \frac{exp(s^T f_i/\tau)}{Z_i} \\ Z_i = \sum_{j=1}^{n} exp(s_j^T f_i/\tau) \end{cases} \tag{9}$$

where Z_i is the normalizing constant. The noise distribution is $P_n = 1/n$. Following previous work, we assume that noise samples are m times more frequent than data samples. Then the posterior probability of sample i with feature s being from the data distribution is:

$$h(i,s) := P(D = 1|i,s) = \frac{P(i|s)}{P(i|s) + mP_n(i)}. \tag{10}$$

The training objective is to minimize the negative log-posterior distribution of data and noise samples:

$$\begin{aligned} J_{NCE}(\theta) = &-E_{P_d}\left[logh(i,s)\right] \\ &- m * E_{P_n}\left[log(1 - h(i,s'))\right]. \end{aligned} \tag{11}$$

where P_d denotes the actual data distribution. For P_d, s is the feature corresponding to x_i; whereas for P_n, s' is the feature from another sample. Both s and s' are sampled from the non-parametric memory bank S. Computing normalizing constant Z_i in 9 costs so much. We treat it as a constant and estimate its

value via Monte Carlo approximation:

$$Z \simeq Z_i \simeq nE_j \left[exp\left(s_j^T f_i/\tau\right)\right] = \frac{n}{m}\sum_{k=1}^{m} exp(s_{j_k}^T f_i/\tau), \tag{12}$$

$\{j_k\}$ is a random subset of indices. NCE reduces the computational complexity from O(n) to O(1) per sample, which yields competitive performance in our experiment.

Proximal Regularization: Instead of having many instances in one class, we only have one instance per class. During each training period, each class is only visited once, which will produce large oscillations due to random sampling fluctuations. The proximal optimization [51] method is employed to encourage the smoothness of the training dynamics. At current iteration t, the feature vector for data x_i is computed from the model $s_i^{(t)} = f_\theta(x_i)$. The memory bank of all the features are stored at previous iteration $S = \{s^{(t-1)}\}$. The loss function for a positive sample from P_d is:

$$-\log h\left(i, s_i^{(t-1)}\right) + \lambda||s_i^{(t)} - s_i^{(t-1)}||_2^2. \tag{13}$$

As training converges, the difference between iterations gradually decreases, i.e., $s_i^{(t)} - s_i^{(t-1)}$. The final objective function becomes:

$$J_{NCE}(\theta) = -E_{P_d}\left[logh\Big(i, s_i^{(t-1)} - \lambda||s_i^{(t)} - s_i^{(t-1)}||_2^2\Big]\right) - m * E_{P_n}\left[log(1 - h(i, s'^{(t-1)}))\right]. \tag{14}$$

IV. Experiment

The experiment was implemented on four different data sets, and the results of the experiment were evaluated according to the classification of supervised and unsupervised methods. The following is a list of all compared methods:

Supervised models:

- RF (random forest) is a classifier with multiple decision trees.
- DT (decision tree) refers to a decision support tool that uses a tree-like model of decisions and their possible consequences.
- KNN (k-nearest neighbor) is a classifier that finds a k nearest instance and votes to decide the class name of the new instance.
- GBDT (gradient boosting decision tree) is an iterative decision tree algorithm, which is composed of multiple decision trees. The results of all trees are accumulated to determine the final label.
- LSTM (long-short term memory) is a model with expandable nodes are suitable for temporal data.
- CNN (convolutional neural network) is a kind of feedforward neural network with deep structure and convolution calculation.
- Attention-based LSTM is an LSTM model with attention mechanism, which adjusts node output considering the attention weight.
- BiLSTM (bidirectional long-short term memory) is composed of a forward LSTM and a backward LSTM.
- GRU (gated recurrent unit) is a gating mechanism in recurrent neural networks, which is like an LSTM with a forget gate, but has fewer parameters than LSTM.
- GraphSleepNet [52] is an adaptive spatial-temporal graph convolutional networks for sleep stage classification.
- AttnSleep [53] is an attention-based deep learning approach for sleep stage classification.

Unsupervised models:

- K-means is an unsupervised clustering algorithm with iterative solution.
- GMM (Gaussian mixture model) [54] refers to the linear combination of multiple Gaussian distribution functions.
- HC (hierarchical clustering) [55] is an unsupervised algorithm that groups similar objects into groups called clusters.
- PCA (principal components analysis) [56] is an unsupervised learning dimension reduction algorithm.
- UEL (unsupervised embedding learning) [57] is a method proposed for classification task via invariant and spreading instance feature.
- PIC (parametric instance classification) [58] is a model designed for unsupervised visual feature learning.
- ContraWR (contrast with the world representation) [59] is a self-supervised model that uses global statistics from the dataset to distinguish signals associated with different sleep stages.
- CAE-M (convolutional auto-encoding memory network) [60] is an unsupervised deep anomaly detection for multi-sensor time-series signals.
- TS-TCC [61] is an unsupervised time-series representation learning framework via temporal and contextual contrasting.

A. Datasets

Our experiment was implemented on three sleep datasets, pressure map dataset (pressure map) [62], polysomnography (PSG) dataset [63], and sleep bioradiolocation dataset (bioradar) [64].

The pressure map data set was collected using two types of pressure-sensitive mattresses, including two corresponding independent experiments for several sleep positions.

Experiment I (e1): The pressure data was obtained from 13 subjects, 8 standard postures, and 9 additional states. A Vista Medical FSA SoftFlex 2048, 32×64 pressure mattress was used to collect the pressure data. The output posture image was in the range of [0,1000] with the sampling rate of 1 Hz. Each output file contained 120 image frames (approximately 2 minutes).

Experiment II (e2): Two kinds of 27×64 pressure mattresses (sponge mattress and air mattress) of Vista Medical BodiTrak BT3510 with a size of 27×64, were used to collect the pressure data of three standard postures for 8 subjects. Each output file contained an average output range of about 20 frames, with an output range of [0,500], and the sampling rate was 1 Hz.

The PSG Dataset contained the acceleration and heart rate output of 31 subjects monitored by the Apple watch, as well

as the labeled sleep data of the gold-standard polysomnography. They wore Apple watches to collect ambulatory activity patterns for a week and spent the night in a sleep lab. The sleep data and labels recorded by Apple watch were stored in separate files and marked with a random subject identifier. There were five sleep stages in the dataset: wake, NREM 1 (N1), NREM 2 (N2), NREM 3 (N3), REM (wake = 0, N1 = 1, N2 = 2, N3 = 3, REM = 5).

The sleep bioradiolocation dataset contained 32 non-contact sleep monitoring records. The bioradar was developed in the Remote Sensing Laboratory of Bauman Moscow State Technical University. All subjects had no sleep-disordered breathing and sleep-related movement. Among them, four patients were diagnosed with insomnia and their records were marked. The output of the bioradar was a continuous wave of a quadrature receiver. The radar adopted 3.6GHz - 4.0GHz stepped frequency modulation, the maximum transmitting power is 3mW, and the sampling rate was 50Hz.

B. Experimental Settings

The experiment was completed on four sleep datasets, and appropriate settings were arranged according to the characteristics of each data set. The device uses GeForce RTX 2080, 31.1 GiB of memory and Intel Xeon(R) W-2133 CPU. The settings are described in accordance with the dataset.

For the four sleep datasets, we randomly selected 80% for training, and 20% for testing, with a data capacity of 400 to 29000. The final testing time on each dataset was 302ms (pressure map I), 131ms (pressure map II), 259ms (PSG), and 130ms (bioradar).

Pressure Map Dataset I (e1): The dataset contained 20024 samples, we first convert the image to gray image and convert each value into 0 to 1 for training. The input dimension and time step size in LSTM were 64 and 32, which matches the size of original images, with a hidden output of 128, and a learning rate of 0.001. The number of negative samples for NCE was set to 512, the temperature parameter for the softmax classifier was 0.2, and the momentum for non-parametric updates was 0.5.

Pressure Map Dataset II (e2): The dataset contained 462 samples, due to the small number of images, SleepGAN was selected to be enabled. First of all, we use image enhancement to increase the contrast of the image to solve the problem of low resolution, which was also conducive to enhance the generation ability of SleepGAN. In the generator of SleepGAN, the dimensionality of the latent space was 100, the input, output, kernel size, and stride size of the three convolution layers are (128, 128, 3, 1), (128, 64, 3, 1), (64, 1, 3, 1) respectively, with 2 up-sampling layers. Finally, 18000 images were generated with SleepGAN, 6000 for each class, which fully met the training requirements of the SSIL. The input dimension and time step size in the LSTM module in SSIL were 64 and 27, which matches the size of original images, with a hidden output of 128, and a learning rate of 0.001, other settings were the same as e1.

PSG Dataset: The dataset contained 25418 samples, due to the sparse features of each sample and the limitation of deep learning, our proposed model did not achieve good results on this dataset. The input dimension of LSTM in SSIL is 4 with the step size of 2.

Sleep Bioradiolocation Dataset: The dataset contained 4650 samples, which was enough to train directly with the SSIL model, instead of using SleepGAN to generate the simulated sleep data first. After the rotation self-supervised process, the input dimension and time step size of the LSTM module were set to 32 and 10, with the hidden layer dimension of 128, and the learning rate of 0.001. The number of negative samples for NCE was 512, the temperature parameter for the softmax classifier was 0.1, and the momentum for non-parametric updates was 0.5.

TABLE I
PERFORMANCE OF CLASSIFICATION COMPARED WITH STATE-OF-THE-ART MODELS ON PRESSURE MAP DATASET I

Classifier	DT	GBDT	LR	RF	KNN	-	-
Supine	99.21%	99.58%	99.44%	99.67%	99.91%	-	-
Right	98.47%	97.82%	98.47%	98.58%	98.58%	-	-
Left	98.73%	97.57%	98.52%	98.20%	98.52%	-	-
Supervised Models	CNN	BiLSTM	LSTM(Attn)	LSTM	GRU	GraphSleepNet[57]	AttnSleep[58]
Supine	99.09%	99.10%	98.64%	99.57%	99.34%	99.17%	99.28%
Right	98.87%	98.87%	98.92%	99.14%	97.93%	98.98%	98.87%
Left	99.18%	98.88%	96.91%	98.46%	98.43%	98.73%	98.43%
Unsupervised Models	Kmeans	GMM[59]	HC[60]	PCA[61]	UEL [62]	PIC[63]	-
Supine	83.09%	76.71%	13.74%	96.72%	98.69%	97.60%	-
Right	47.35%	52.99%	48.26%	42.07%	97.16%	97.74%	-
Left	48.96%	6.4%	42.00%	83.13%	97.63%	97.29%	-
Unsupervised Models	ContraWR[64]	CAE-M[65]	TS-TCC[66]	SSIL	-	-	-
Supine	98.75%	99.08%	97.39%	***99.54%***	-	-	-
Right	98.02%	97.93%	96.93%	***98.24%***	-	-	-
Left	98.48%	97.94%	96.21%	***98.83%***	-	-	-

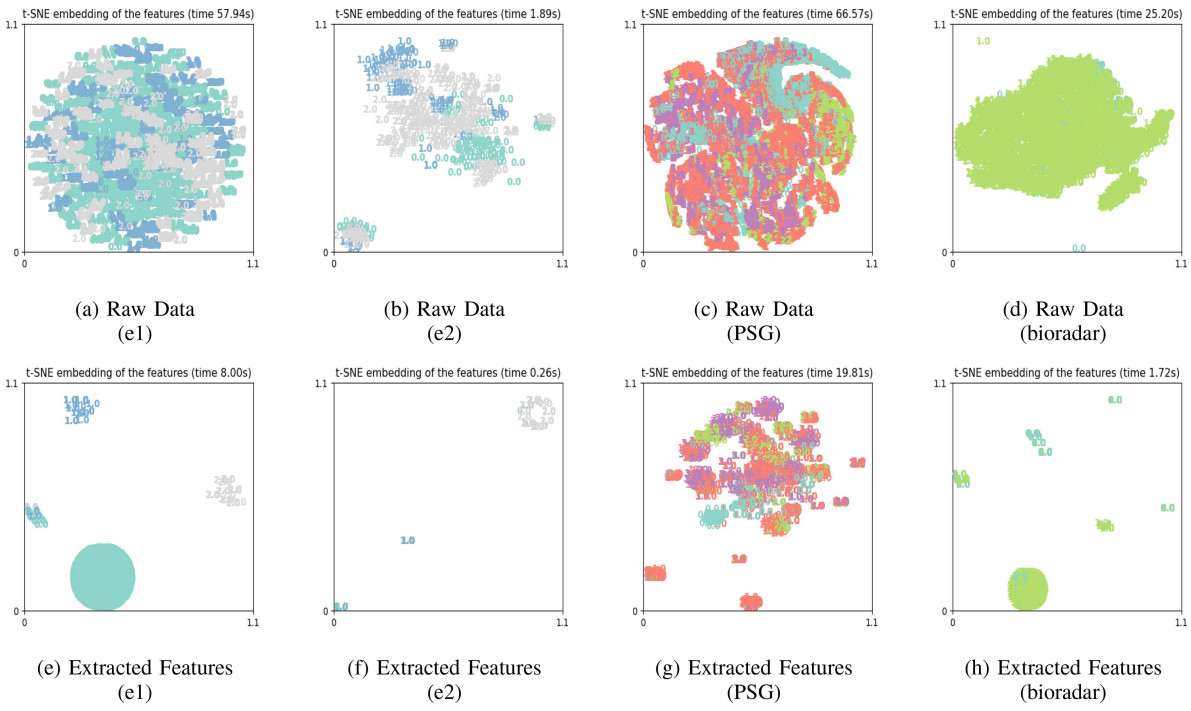
C. Results of Sleep Posture Recognition

In this experiment, three sleep postures, i.e., supine, left, and right lateral positions are recognized to help the subjects adjust the incorrect postures. Among the three sleeping positions, the supine position accounts for the largest proportion, and the number of left and right positions is similar. The input data is the images collected by the two pressure mattresses, which contain rich spatial and temporal information, so all the test methods have achieved satisfactory results.

Results on Pressure Map Dataset I (e1): The sleep data is derived from Vista Medical FSA SoftFlex 2048 with each image has 2048 (64*32) dimensional features.

As demonstrated in Fig. 6, the SSIL has a classification accuracy rate of over 98% for each class and a recognition probability of over 99% for the supine position, which is suitable for sleep posture detection and recognition. Additionally, due to the large difference in the features of the left and right lateral positions, it will hardly be misclassified, but the difference from the supine position is small, and it is more likely to be misclassified as the supine position.

Furthermore, the classification results of traditional classifiers, supervised deep models and unsupervised models for the three sleep postures are shown in Table I. Generally, the unsupervised algorithm is not as effective as the supervised algorithm in learning features of sleep postures. Because the unsupervised algorithm does not train the classification label in advance, which brings great difficulties to the classification. In this dataset, the best performance of supervised algorithm is KNN, which is better than other traditional classifiers. The novel deep models, i.e., GraphSleepNet [52] and AttnSleep [53] have achieved more than 99% accuracy of the testing data, which apply the graph



(a) Raw Data (e1) (b) Raw Data (e2) (c) Raw Data (PSG) (d) Raw Data (bioradar)

(e) Extracted Features (e1) (f) Extracted Features (e2) (g) Extracted Features (PSG) (h) Extracted Features (bioradar)

Fig. 5. Dimension reduction comparison between extracted features and original data by using t-SNE.

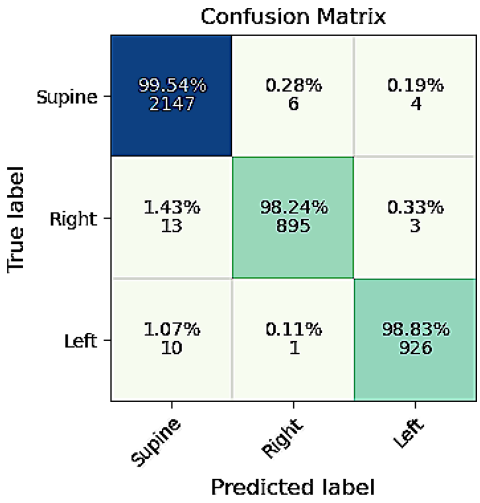


Fig. 6. The confusion matrix of the classification results on the e1 dataset.

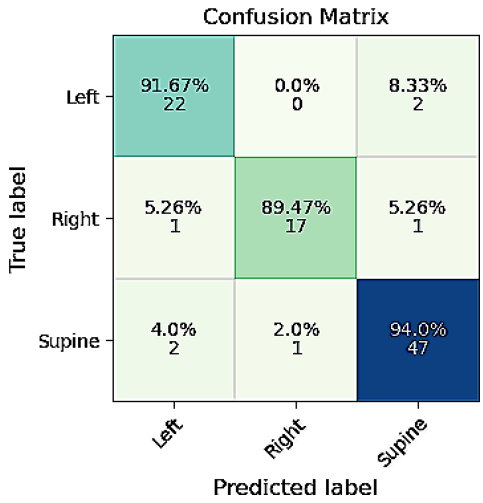


Fig. 7. The confusion matrix of the classification results on the e2 dataset.

model and attention model into the training process. Among unsupervised algorithms, UEL [57] and PIC [58] have outstanding effect since they measure posture information by embedding an instance based softmax embedding method and a parametric instance learning approach, which processes each sample as an instance, and captures abundant global and local information. The proposed model is more complex for extract spatiotemporal and self-supervised information, which greatly improves the classification performance. We also compare three SOTA unsupervised methods for sleep recognition published in 2021 [59]–[61], the results on dataset e1 are very stable, slightly inferior to our proposed method. These latest methods show the advantages of unsupervised methods for sleep recogniton, we will continue to optimize SSIL model through these methods.

Results on Pressure Map Dataset II (e2): Similar to pressure dataset e1, dataset e2 is also collected by a pressure sensor of Vista Medical BodiTrak BT3510 brand with 1728 (64*27) dimensions of each image. Due to the small number of samples, we first use SleepGAN to generate enough simulation images, and 80% of the original dataset for training, and still use 20% of the original dataset for testing, which greatly improves the effect.

As illustrated in Fig. 7, we can see the results of classifying 93 testing samples are all around 90%, indicating the advantages of using generated data. Moreover, we use the testing data of

TABLE II
PERFORMANCE OF CLASSIFICATION COMPARED WITH STATE-OF-THE-ART MODELS ON PRESSURE MAP DATASET II

Classifier	DT	GBDT	LR	RF	KNN	-	-
Supine	77.36%	92.45%	92.45%	88.68%	90.57%	-	-
Right	63.64%	68.18%	90.91%	77.27%	81.82%	-	-
Left	83.33%	55.56%	83.33%	83.33%	88.89%	-	-
Supervised Models	CNN	BiLSTM	LSTM(Attn)	LSTM	GRU	GraphSleepNet[57]	AttnSleep [58]
Supine	89.47%	81.82%	84.48%	75.47%	87.50%	88.86%	87.39%
Right	72.22%	89.47%	83.33%	68.42%	63.64%	85.93%	86.30%
Left	94.44%	84.62%	76.47%	80.95%	65.22%	89.96%	89.18%
Unsupervised Models	Kmeans	GMM[59]	HC[60]	PCA[61]	UEL [62]	PIC[63]	-
Supine	100.00%	43.64%	12.50%	85.71%	89.37%	90.79%	-
Right	80.59%	11.11%	4.76%	52.94%	88.41%	88.69%	-
Left	0.00%	85.00%	93.75%	30.77%	91.59%	90.63%	-
Unsupervised Models	ContraWR[64]	CAE-M[65]	TS-TCC[66]	SSIL	-	-	-
Supine	91.15%	90.01%	90.81%	***91.67%***	-	-	-
Right	89.01%	88.19%	90.91%	***89.47%***	-	-	-
Left	85.92%	84.92%	83.33%	***94.00%***	-	-	-

the synthetic dataset (with the capacity of 18462, 462 for real samples) for classification, and the accuracy rate reaches about 99%, which proves the high performance of SSIL classification and the high generation ability of SleepGAN.

Moreover, the classification results of traditional classifiers, supervised deep models and unsupervised models for the three sleep postures are demonstrated in Table II. The classification results of the three postures are illustrated in the light of traditional classifiers, supervised models and unsupervised models. Generally, the supervised model is better than the unsupervised model. For the supine position, GBDT and LR have the best effect, and the results exceed the deep model due to the lack of training samples; for the right lying position, LR has the highest accuracy; for the left lying position, CNN and SSIL have the best effect. Although the performance of UEL [57] and PIC [58] is not outstanding in the three posture classification, they are stable and slightly better than the average performance of the compared supervised algorithms. Although SSIL has some disadvantages compared with the supervised model, its performance is stable, and has a gap with other unsupervised models. Compared with the results of ContraWR [59], CAE-M [60] and TS-TCC [61], the performance of TS-TCC [61] for detecting supine and right lying posture ranks the first, and the right lying recognition result is superior than ours, which illustrates that it is sensitive to this dataset.

Additionally, we also discuss the generated images and loss changes in SleepGAN. In Fig. 8, the comparison of the original image, enhanced image, and generated image shows that the general position of the image generated by SleepGAN is similar to that of the original image, and the simulation degree of supine position is high. Although the side-lying image generated by SleepGAN has a serious loss of detail, it does not affect the model's differentiation of these classes. Experiments show that SleepGAN has the potential for generation of fine sleep pictures. The three losses in training epoch is plotted in Fig. 10. The three losses show a decreasing trend and eventually become stable, which verifies the availability of SleepGAN.

Moreover, we split SleepGAN to generate three different split models, i.e., remove the loss of L_{sim}, remove the temporal discriminator, and remove the spatial discriminator. It can be seen from Fig. 9 that the positions of the three generated images are basically correct compared with the original graph, but no matter which module is missing, the details of the generated image will be lost, e.g., the information of human limbs is lost in the generated image of the supine position.

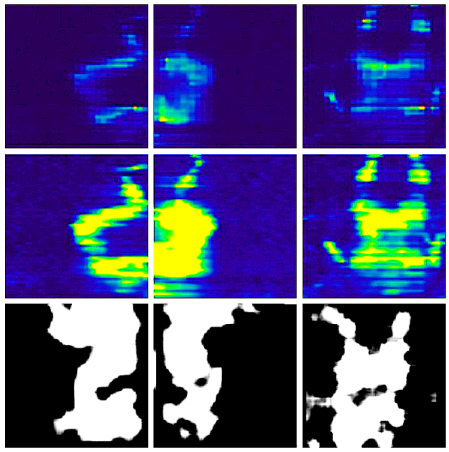


Fig. 8. Images generated by SleepGAN. The first row is the original image of three sleep positions, the second row is the enhanced image, and the third row is the generated image. It can be seen that the image of the supine position is better, while the details of the left and right positions are poor, but it does not affect the overall position.

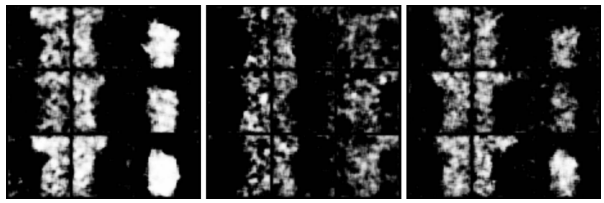


Fig. 9. Images generated by the split SleepGAN. We use the matrix of 3*3 to represent the image generated in the training process, showing three cases from left to right: remove the loss of L_{sim}, remove the temporal discriminator, and remove the spatial discriminator.

D. Results of Sleep Stage Recognition

In this section, four-dimensional features of the sleep sequence produced by the provided pre-processing tool, which are cosine acceleration, counting feature, heart rate and time stamp. We divided the five sleep stages into four classes, wake, N1&N2, N3, and REM, respectively.

Firstly, we use the confusion matrix to represent the classification of the model. As shown in Fig. 11, N1 and N2 have the largest number of samples in sleep stage, which is much higher than other stages, reflecting the imbalance of data sets, which will lead to over fitting of deep model. Although the recognition accuracy of SSIL in the other three sleep stages is relatively average, there is still a bias towards the data.

Secondly, the classification results of traditional classifiers, supervised deep models and unsupervised models for the four sleep stages are exhibited in Table III. Due to the sparse feature of each sleep stage and the imbalance of the data, all models do not perform well. We can see that the recognition rate of most models for N1 and N2 sleep stage is higher than other stages, and this imbalance will make the model over fit. Among the three models, RF, LSTM and kmeans are the most accurate models for wake recognition. The performance of SSIL is balanced in

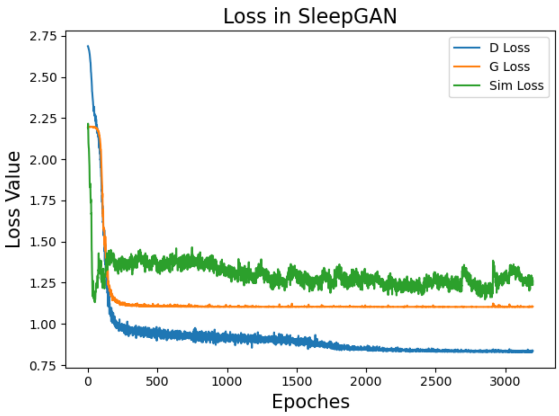


Fig. 10. The loss in SleepGAN. D loss, G loss, and Sim loss represent the discriminator loss, generator loss and similarity loss in the SleepGAN. G loss and D loss converge more smoothly, while Sim loss fluctuates between 1.25 and 1.5, which indicates that there is still a gap between the images generated by the two feature extractors, i.e., temporal and spatial feature extractors, in the discriminator of the SleepGAN. We set Sim loss to make the results of the two feature extractors as the same as possible to avoid the inconsistency of the generated images.

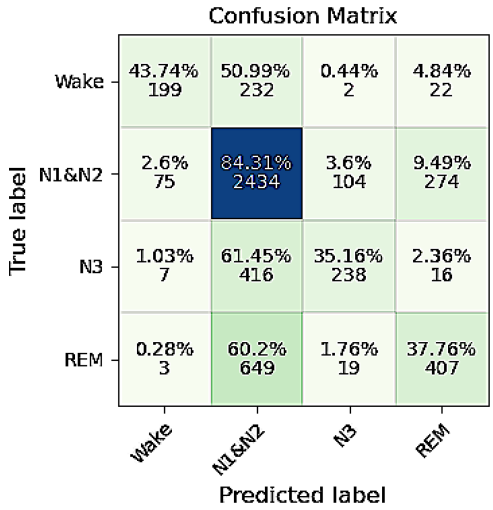


Fig. 11. The confusion matrix of the classification results on the PSG dataset.

TABLE III
PERFORMANCE OF CLASSIFICATION COMPARED WITH ADVANCED MODELS ON PSG DATASET (WAKE/N1&N2/N3/REM)

Classifier	DT	GBDT	LR	RF	KNN	-	-
Wake	60.23%	36.32%	32.18%	63.91%	55.17%	-	-
N1&N2	73.28%	96.82%	94.46%	82.14%	77.47%	-	-
N3	53.92%	3.46%	5.27%	50.75%	46.99%	-	-
REM	61.86%	10.82%	3.70%	57.26%	49.95%	-	-
Supervised Models	CNN	BiLSTM	LSTM(Attn)	LSTM	GRU	GraphSleepNet[57]	AttnSleep [58]
Wake	35.05%	27.38%	39.06%	56.53%	49.76%	55.37%	56.38%
N1&N2	93.49%	95.20%	87.94%	89.26%	92.20%	88.62%	89.49%
N3	35.71%	0.00%	29.19%	14.04%	4.85%	28.59%	24.75%
REM	0.00%	10.49%	24.89%	10.37%	9.03%	35.15%	30.57%
Unsupervised Models	Kmeans	GMM[59]	HC[60]	PCA[61]	UEL [62]	PIC[63]	-
Wake	48.81%	22.34%	16.40%	2.98%	42.37%	40.33%	-
N1&N2	73.20%	13.22%	9.75%	31.78%	83.92%	80.60%	-
N3	30.11%	66.30%	4.97%	6.96%	31.72%	32.63%	-
REM	9.18%	28.69%	78.24%	74.96%	35.29%	30.41%	-
Unsupervised Models	ContraWR[64]	CAE-M[65]	TS-TCC[66]	SSIL	-	-	-
Wake	36.69%	38.43%	35.51%	***43.74%***	-	-	-
N1&N2	95.73%	84.25%	97.93%	***84.31%***	-	-	-
N3	5.76%	29.19%	4.72%	***35.16%***	-	-	-
REM	11.14%	29.89%	10.82%	***37.76%***	-	-	-

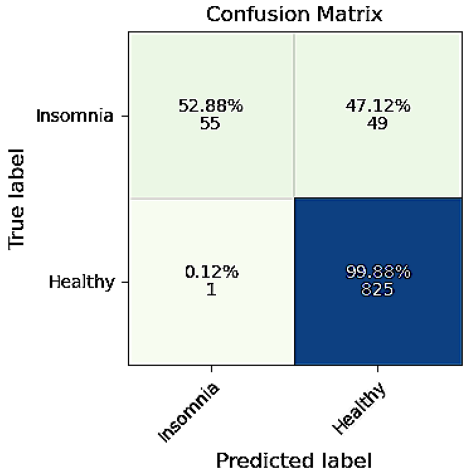


Fig. 12. The confusion matrix of the classification results on the bioradar dataset.

TABLE IV
PERFORMANCE OF CLASSIFICATION OF INSOMNIA PATIENTS AND HEALTHY SUBJECTS COMPARED WITH STATE-OF-THE-ART MODELS ON SLEEP BIORADIOLOCATION DATASET

Classifier	DT	GBDT	LR	RF	KNN	-	-
Healthy	91.68%	100.00%	79.38%	97.14%	95.78%	-	-
Insomnia	48.00%	1.60%	20.00%	49.60%	60.80%	-	-
Supervised Models	CNN	BiLSTM	LSTM(Attn)	LSTM	GRU	GraphSleepNet[57]	AttnSleep [58]
Healthy	100.00%	93.21%	96.33%	94.33%	94.01%	95.23%	95.82%
Insomnia	0.00%	51.90%	58.04%	60.17%	48.21%	58.16%	59.23%
Unsupervised Models	Kmeans	GMM[59]	HC[60]	PCA[61]	UEL [62]	PIC[63]	-
Healthy	12.80%	36.23%	36.76%	4.48%	97.45%	96.95%	-
Insomnia	92.55%	54.84%	48.25%	87.44%	55.17%	51.57%	-
Unsupervised Models	ContraWR[64]	CAE-M[65]	TS-TCC[66]	SSIL	-	-	-
Healthy	95.57%	96.23%	97.95%	***99.88%***	-	-	-
Insomnia	50.39%	49.05%	50.22%	***52.88%***	-	-	-

the unsupervised model, the recognition rate of N1 and N2 is the highest, and the results of the other three stages are balanced. ContraWR [59], CAE-M [60] and TS-TCC [61] have great deviation in the recognition results of N1 and N2, which proves the influence of unbalanced data on unsupervised learning algorithm. Therefore, these methods have not achieved considerable results.

E. Results of Insomnia Detection

In this experiment, we successfully detected and distinguished the sleep difference between healthy subjects and patients with insomnia, and compared the recognition accuracy of traditional classifiers and deep models. Experimental results show that SSIL can obtain a higher recognition rate, but due to the unevenness of the data, it still cannot overcome the phenomenon of over-fitting.

Fig. 12 shows the recognition performance of the split parts of the proposed model. We can see that the recognition rate of healthy subjects is significantly higher than that of insomnia patients, which is caused by the lack of data of insomnia patients. In this dataset, the effect of removing the significant enhancement layer from the model is the worst, which shows that it has the greatest impact on the model and the strongest ability to deal with unbalanced data.

TABLE V
PERFORMANCE OF CLASSIFICATION COMPARED WITH ADVANCED MODELS

Pressure-e1[67]		Pressure-e2 [67]		PSG-4class [68]		Bio-radar [69]	
Supervised Models-Classifiers							
DT	98.93%	DT	75.27%	DT	67.16%	DT	85.81%
GBDT	98.70%	GBDT	79.57%	GBDT	60.78%	GBDT	86.77%
LR	99.00%	LR	90.32%	LR	57.78%	LR	71.40%
RF	99.08%	RF	84.95%	RF	71.08%	RF	90.75%
KNN	99.28%	KNN	88.17%	KNN	65.61%	KNN	91.08%
Supervised Models-Deep Networks							
CNN	98.98%	CNN	87.10%	CNN	59.01%	CNN	86.49%
BiLSTM	99.25%	BiLSTM	84.95%	BiLSTM	58.68%	BiLSTM	88.47%
LSTM(Attn)	99.03%	LSTM(Attn)	83.87%	LSTM(Attn)	62.41%	LSTM(Attn)	91.72%
LSTM	99.20%	LSTM	75.27%	LSTM	58.39%	LSTM	90.00%
GRU	98.80%	GRU	76.34%	GRU	58.54%	GRU	88.49%
GraphSleepNet[57]	99.02%	GraphSleepNet[57]	88.25%	GraphSleepNet[57]	62.25%	GraphSleepNet[57]	90.87%
AttnSleep [58]	99.07%	AttnSleep [58]	87.62%	AttnSleep [58]	60.28%	AttnSleep [58]	91.94%
Unsupervised Models							
Kmeans	66.52%	Kmeans	37.63%	Kmeans	51.23%	Kmeans	81.83%
GMM[59]	54.15%	GMM[59]	46.23%	GMM[59]	24.34%	GMM[59]	38.71%
HC[60]	34.67%	HC[60]	37.00%	HC[60]	25.29%	HC[60]	42.50%
PCA[61]	80.82%	PCA[61]	72.04%	PCA[61]	36.04%	PCA[61]	75.48%
UEL[62]	98.03%	UEL [62]	88.12%	UEL [62]	61.33%	UEL [62]	91.19%
PIC[63]	97.56%	PIC[63]	89.37%	PIC[63]	59.42%	PIC[63]	90.38%
ContraWR[64]	98.26%	ContraWR[64]	91.19%	ContraWR[64]	60.28%	ContraWR[64]	91.30%
CAE-M[65]	98.28%	CAE-M[65]	90.91%	CAE-M[65]	62.44%	CAE-M[65]	90.61%
TS-TCC[66]	97.63%	TS-TCC[66]	91.39%	TS-TCC[66]	60.06%	TS-TCC[66]	92.82%
SSIL	***99.08%***	SSIL	***92.47%***	SSIL	***64.31%***	SSIL	***94.62%***

TABLE VI
PERFORMANCE OF SPLIT COMPONENTS IN SSIL

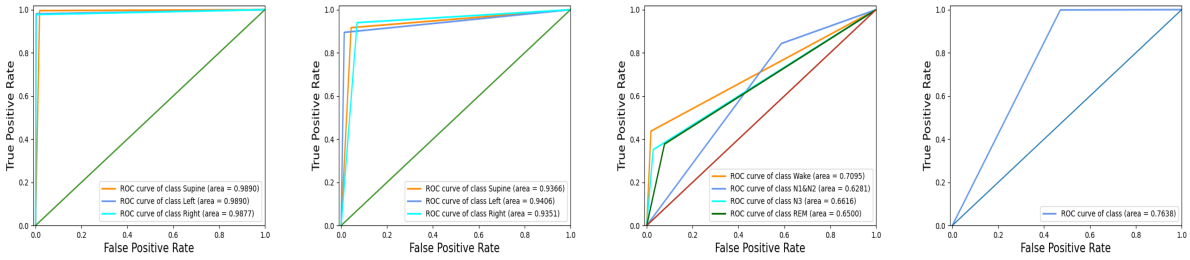
Dataset	Self-Supervised Rotation	Instance Learning (Param Softmax)	Instance Learning (Non-Param Softmax)	SSIL
e1	98.99%	97.79%	98.98%	***99.08%***
e2	90.19%	84.37%	86.46%	***92.47%***
PSG	59.38%	60.27%	62.18%	***64.31%***
Bioradar	90.05%	90.18%	92.26%	***94.62%***

In Table IV, the classification results of traditional classifiers, supervised deep models and unsupervised models for the insomnia detection are shown in details. Similar to the dataset of sleep stage, bioradar dataset also has the phenomenon of data imbalance. The number of samples from healthy subjects accounts for more than 70%. Therefore, the CNN model has a serious over fitting phenomenon. All samples are identified as healthy subjects. KNN and LSTM model are excellent and stable, and the detection probability of insomnia samples is more than 60%. There is still a big margin for improvement in our SSIL to deal with the imbalance of data in this dataset. ContraWR [59], CAE-M [60] and TS-TCC [61] have great deviation in the recognition results of healthy controls, and the recognition rate of TS-TCC [61] is slightly higher than that of the other two models.

F. Performance of the Entire Model

1) Performance of the Split Components of SSIL: In this section, as an ablation experiment, we evaluated the performance of the split components in SSIL, including self-supervised rotation, instance learning with param softmax, instance learning with non-param softmax, and the whole model. As illustrated in Table VI, instance learning with non-param softmax slightly outperforms that with param softmax. In datasets e2 and bio-radar, the self-supervised rotation module shows its data amplification capability and is superior than other components.

The results of all the models compared in this paper are shown in Table V. We can see that the unsupervised learning method is obviously worse than the supervised learning method. Due to the lack of classification label information, the learned features will not be representative. On the pressure-e1 dataset, the traditional classifier and the deep model have the similar effect, which can produce interpretable sleep posture features. On the contrary, unsupervised learning method is poor. UEL [57] and PIC [58] regard training samples as instances and fuse various complex computing methods for obtaining a result close to the supervised algorithms, which is significantly superior than other unsupervised methods. Our SSIL exceeds these unsupervised methods and achieves the same good result as supervised method. On the pressure-e2 dataset, unlike the pressure-e1 with sufficient training samples, the recognition rate of deep model is reduced. With the comparison, GraphSleepNet [52] and AttnSleep [53] perform relatively stable, and the classification results of ContraWR [59], CAE-M [60] and TS-TCC [61] are higher than that of other unsupervised models. Meanwhile, SSIL generates sleep images through SleepGAN, which solves the problem of insufficient samples and produces high accuracy. On the PSG dataset, the recognition rate of supervised model is about 50%–60%. The unsupervised model have weak classification ability, while CAE-M [60] performs the best among them. SSIL is better than most supervised models, while RF performs the best that the tree structure can analyze sparse features more carefully. On the bioradar dataset, KNN, AttnSleep [53] and TS-TCC [61] stand out



(a) e1 (b) e2 (c) PSG (d) Bioradar

Fig. 13. The ROC curve of SSIL on the four sleep datasets. AUC value is used to evaluate the classification effect and stability of each class.

from all the models, and SSIL achieves the highest classification results as the supervised model.

Specifically, we draw the ROC curve for comparing the classification results of each sleep state in the datasets. It can be seen in Fig. 13, the AUC value of the three sleep postures, i.e., supine, left lying position, and right lying position, fluctuate about 98%, indicating the tendency of perfect classification in the e1 dataset. In the e2 dataset, obviously, the recognition result of the SSIL is the best, but it still does not reach the result of e1. Due to the problem of uneven data in PSG dataset, the large amount of N1 and N2 data leads to abnormal ROC curve. Among other classes, the class "Wake" has the highest AUC, which illustrates that this classification is significantly different from other classifications. The Bioradar dataset has only two classes, so there is only one ROC curve. As with the PSG dataset, the sample size of insomnia patients is too small, which directly affects the AUC value.

2) Feature Separability Comparison: Furthermore, we also show that the output of SSIL is separable compared with the original data. We use a dimensionality reduction visualization method t-SNE (t-distributed Stochastic Neighbor Embedding) [65], which can convert the similarity between data points into probability, as shown in Fig. 5. The first and second rows represent the t-SNE outputs of the original data and the SSIL.

Specifically, the results on e1, e2, and bioradar datasets are significant. The sample points belonging to the same classes are clustered together. The clusters have a large interval and clear boundary. The samples with different colors overlapped together are misclassified samples, which indicates that the features of samples are similar and difficult to distinguish. For the dataset PSG, we can see that the sample points are scattered, and it is difficult to distinguish subjects in the same class. It also shows that our method on this dataset has limitations and is also related to the sparsity of features.

3) Robustness Verification: For verifying the robustness of the model, three kinds of noise, i.e., Gaussian noise, salt noise and Poisson noise have been added to the data (Table VII). The noisy data are input into our model for training, the performance is slightly inferior to our original data. The performance degradation range is within 2%, which verifies the robustness of the proposed model. Compared with the three noise, it can be seen that the Gaussian noise has a great impact on the model and data, but does not affect the average recognition level of the model.

4) Preliminary Work: Through the above evaluation of the results of the sleep recognition tasks, we can see that our proposed method goes beyond a large number of supervised and unsupervised models. The previous work confirmed that the experimental results of our proposed SSRM [38] (composed of an upstream self-supervised pre-training task and a downstream recognition) in performing specific sleep recognition tasks exceeded the SSIL proposed in this paper. For example, the recognition rate reached 71.01% on the PSG dataset, reflecting the advantages of the self-supervised learning mechanism. The similarity between these two models is that they are both self-supervised models, and the architecture is divided into pre-training and downstream recognition tasks, but the settings when implementing sleep recognition experiments are different, which is one of the reasons for the slightly larger difference in results. Compared with most deep learning models, our proposed SSIL still reflects its stability and efficiency.

TABLE VII
PERFORMANCE OF SSIL AFTER ADDING NOISE

	Noise			
Dataset	Gaussian noise	Salt noise and	Poisson noise	Non-noise
e1	98.33%	98.98%	98.69%	***99.08%***
e2	91.80%	92.03%	92.14%	***92.47%***
PSG	62.96%	63.79%	63.97%	***64.31%***
Bioradar	93.83%	94.20%	94.37%	***94.62%***

V. CONCLUSION

This paper presents a novel transferable self-supervised method by effectively exploiting multi-modal sleep data for stage and posture recognition. Our key idea is to jointly utilize two developed unsupervised learning mechanisms, i.e., the SleepGAN, and the self-supervised instance learning model, to generate sleep data and learn better feature representations. Our proposed self-supervised method can achieve this goal by learning global and local feature representations, i.e., salient features of classes and individuals, which shows effective for sleep stage and posture classification. Experimental results on four public datasets demonstrate that our method outperforms other unsupervised methods and several supervised methods, and achieves comparable performance to the supervised baseline. We also show the generated sleep data from the SleepGAN.

Although the proposed model has obtained considerable recognition results in addressing these three tasks, there are also various limitations in dealing with multimodal data. When processing the image data output from the pressure mattress, the limitation of the traditional deep learning framework is that it

cannot learn the significant features of a small number of samples. When processing PSG data, because each original sample has few feature dimensions, we propose to combine multiple original samples into one training sample and classify them by using the features of time series data. When processing biological radar data, the number of samples of healthy subjects is much larger than that of insomnia patients, resulting in over fitting phenomenon. Although the proposed model cannot completely solve the problem of unbalanced sample number, it is also better than other popular models. In summary, the data of different modalities has a variety of performance characteristics, so it is necessary to design targeted models to overcome the corresponding difficulties.

For the future work, we will focus on using a variety of sleep data to quantify the sleep status of patients. By locating the stage of sleep disorders and analyzing bad sleep posture, we can predict the changes of sleep status and the probability of insomnia in the future. These data and methods will also be applied to the prediction and prevention of depression after being verified and tested. Furthermore, semi-supervised learning [66] or weakly supervised learning [67] methods have gradually matured in the field of video segmentation and object detection, and achieved promising detection results. We will also take their advantages to optimize our sleep recognition model. For instance, weakly supervised model is adopted to label sleep data for location, segmentation and recognition; Or semi-supervised model is designed to detect the saliency of sleep data before recognition. It will also become an innovative research direction in the future.

REFERENCES

[1] A. Sasidharan, S. Sulekha, and B. Kutty, "Current understanding on the neurobiology of sleep and wakefulness," *Int. J. Clin. Exp. Physiol.*, vol. 1, no. 1, pp. 3–9, 2014.

[2] J. Yang, J. M. Keller, M. Popescu, and M. Skubic, "Sleep stage recognition using respiration signal," in *Proc. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2016, pp. 1–1.

[3] A. Roebuck *et al.*, "A review of signals used in sleep analysis," *Physiol. Meas.*, vol. 35, no. 1, pp. 1–57, 2014.

[4] C. Yang, G. Cheung, V. Stankovic, K. Chan, and N. Ono, "Sleep apnea detection via depth video and audio feature learning," *IEEE Trans. Multimedia*, vol. 19, no. 4, pp. 822–835, Apr. 2017.

[5] S. Mansfield, K. Obraczka, and S. Roy, "Pressure injury prevention: A survey," *IEEE Rev. Biomed. Eng.*, vol. 13, pp. 352–368, 2020.

[6] K. Tang, A. Kumar, M. Nadeem, and I. Maaz, "CNN-based smart sleep posture recognition system," *IoT*, vol. 2, pp. 119–139, 2021.

[7] 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.

[8] P. Ghasemzadeh, H. Kalbkhani, S. Sartipi, and M. G. Shayesteh, "Classification of sleep stages based on lstar model," *Appl. Soft Comput.*, vol. 75, pp. 523–536, 2019.

[9] H. Khan, M. Heyat, D. Lai, F. Akhtar, and F. Alkahtani, "Progress in detection of insomnia sleep disorder: A comprehensive review," *Curr. Drug Targets*, vol. 22, no. 6, pp. 672–684, 2021.

[10] E. Dafna, A. Tarasiuk, and Y. Zigel, "Sleep-quality assessment from full night audio recordings of sleep apnea patients," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2012, pp. 3660–3663.

[11] M. Hanaoka, M. Kobayashi, and H. Yamazaki, "Automated sleep stage scoring by decision tree learning," in *Proc. 23rd Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2001, vol. 2, pp. 1751–1754.

[12] S. Y. Chang *et al.*, "An ultra-low-power dual-mode automatic sleep staging processor using neural-network-based decision tree," *IEEE Trans. Circuits Syst. I: Regular Papers*, vol. 66, no. 9, pp. 3504–3516, Sep. 2019.

[13] X. Xu, F. Lin, A. Wang, C. Song, and Y. Hu, "On-bed sleep posture recognition based on body-earth mover's distance," in *Proc. IEEE Biomed. Circuits Syst. Conf.*, 2015, pp. 1–1.

[14] R. Boostani, F. Karimzadeh, and M. Nami, "A comparative review on sleep stage classification methods in patients and healthy individuals," *Comput. Methods Programs Biomed.*, vol. 140, pp. 77–91, 2017.

[15] D. S. Sisodia, K. Sachdeva, and A. Anuragi, "Sleep order detection model using support vector machines and features extracted from brain ecg signals," in *Proc. Int. Conf. Inventive Comput. Inform.*, 2017, pp. 1011–1015.

[16] L. Mulaffer, M. Shahin, M. Glos, T. Penzel, and B. Ahmed, "Comparing two insomnia detection models of clinical diagnosis techniques," in *Proc. 39th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2017, pp. 3749–3752.

[17] L. Walsh, S. McLoone, J. Ronda, J. F. Duffy, and C. A. Czeisler, "Non-contact pressure-based sleep/wake discrimination," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 8, pp. 1750–1760, Aug. 2017.

[18] Y.-Y. Li and Y.-J. Lei, "Sleep posture classification with multi-stream cnn using vertical distance map," in *Proc. Int. Workshop Adv. Image Technol.*, 2018, pp. 1–1.

[19] X. Hu *et al.*, "Non-invasive sleeping posture recognition and body movement detection based on RFID," in *Proc. IEEE Int. Conf. Internet Things IEEE Green Comput. Commun. IEEE Cyber, Phys. Social Comput., IEEE Smart Data*, 2018, pp. 1817–1820.

[20] C. Torres, J. C. Fried, K. Rose, and B. S. Manjunath, "A multiview multimodal system for monitoring patient sleep," *IEEE Trans. Multimedia*, vol. 20, no. 11, pp. 3057–3068, Nov. 2018.

[21] A. Zhao, J. Dong, J. Li, L. Qi, and H. Zhou, "Associated spatio-temporal capsule network for gait recognition," *IEEE Trans. Multimedia*, vol. PP, pp. 846–860, 2022.

[22] M. Radha, P. Fonseca, A. Moreau, M. Ross, and R. M. Aarts, "Sleep stage classification from heart-rate variability using long short-term memory neural networks," *Sci. Rep.*, vol. 9, no. 1, 2019, Art. no. 14149.

[23] T. Ergen and S. S. Kozat, "Online training of lstm networks in distributed systems for variable length data sequences," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 10, pp. 5159–5165, Oct. 2018.

[24] B. Erik, G. Ulf, and G.-M. Gary, "Recurrent deep neural networks for real-time sleep stage classification from single channel EEG," *Front. Comput. Neurosci.*, 2018.

[25] X. Xu *et al.*, "Body-earth mover's distance: A matching-based approach for sleep posture recognition," *IEEE Trans. Biomed. Circuits Syst.*, vol. 10, no. 5, pp. 1023–1035, Oct. 2016.

[26] M. Enayati, M. Skubic, J. M. Keller, M. Popescu, and N. Z. Farahani, "Sleep posture classification using bed sensor data and neural networks," in *Proc. 40th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2018, pp. 461–465.

[27] Y. Li, Y. Lei, L. C. Chen, and Y. Hung, "Sleep posture classification with multi-stream cnn using vertical distance map," in *Proc. Int. Workshop Adv. Image Technol.*, 2018, pp. 1–4.

[28] N. Mohsin, X. Liu, and S. Payandeh, "Signal processing techniques for natural sleep posture estimation using depth data," in *Proc. IEEE 7th Annu. Inf. Technol., Electron. Mobile Commun. Conf.*, 2016, pp. 1–8.

[29] S. M. Mohammadi *et al.*, "Sleep posture classification using a convolutional neural network," in *Proc. 40th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2018, pp. 1–4.

[30] S. Ostadabbas, M. B. Pouyan, M. Nourani, and N. Kehtarnavaz, "In-bed posture classification and limb identification," in *Proc. IEEE Biomed. Circuits Syst. Conf.*, 2014, pp. 133–136.

[31] H. Phan, F. Andreotti, N. Cooray, O. Y. Chén, and M. D. 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.

[32] D. JIANG, Y. nan 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, 2019.

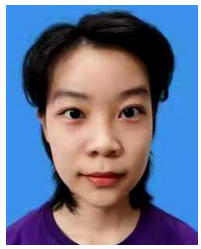
[33] Y. Zhang *et al.*, "Sleep stage classification using bidirectional LSTM in wearable multi-sensor systems," in *Proc. IEEE Int. Conf. Comput. Commun. Workshops*, 2019, pp. 443–448.

[34] Z. Chen, M. Wu, W. Cui, C. Liu, and X. Li, "An attention based CNN-LSTM approach for sleep-wake detection with heterogeneous sensors," *IEEE J. Biomed. Health Inform.*, vol. 25, no. 9, pp. 3270–3277, Sep. 2020.

[35] S. Güneş, K. Polat, and şebnem Yosunkaya, "Efficient sleep stage recognition system based on EEG signal using k-means clustering based feature weighting," *Expert Syst. Appl.*, vol. 37, no. 12, pp. 7922–7928, 2010.

[36] H. G. Jo, J. Y. Park, C. K. Lee, S. K. An, and S. K. Yoo, "Genetic fuzzy classifier for sleep stage identification," *Comput. Biol. Med.*, vol. 40, no. 7, pp. 629–634, 2010.
[37] N. Banluesombatkul *et al.*, "MetaSleepLearner: A pilot study on fast adaptation of bio-signals-based sleep stage classifier to new individual subject using meta-learning," *IEEE J. Biomed. Health Inform.*, vol. 25, no. 6, pp. 1949–1963, Jun. 2020.
[38] A. Zhao, J. Dong, and H. Zhou, "Self-supervised learning from multi-sensor data for sleep recognition," *IEEE Access*, vol. 8, pp. 93907–93921, 2020.
[39] J. Zhang and Y. Wu, "Complex-valued unsupervised convolutional neural networks for sleep stage classification," *Comput. Methods Programs Biomed.*, vol. 164, pp. 181–191, 2018.
[40] Y. El-Manzalawy, O. Buxton, and V. Honavar, "Sleep/wake state prediction and sleep parameter estimation using unsupervised classification via clustering," in *Proc. IEEE Int. Conf. Bioinf. Biomed.*, 2017, pp. 718–723.
[41] K. Feng, H. Qin, S. Wu, W. Pan, and G. Liu, "A sleep apnea detection method based on unsupervised feature learning and single-lead electrocardiogram," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–12, 2020.
[42] I. J. Goodfellow *et al.*, "Generative adversarial networks," *Adv. Neural Inf. Process. Syst.*, vol. 3, pp. 2672–2680, 2014.
[43] C. Fang *et al.*, "Gaussian discriminant analysis for optimal delineation of mild cognitive impairment in Alzheimer's disease," *Int. J. Neural Syst.*, 2018.
[44] J. H. Xue and D. M. Titterington, "Comment on "on discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes," *Neural Process. Lett.*, vol. 28, no. 3, pp. 169–187, 2008.
[45] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier GANs," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 2642–2651.
[46] P. Goyal, D. Mahajan, A. Gupta, and I. Misra, "Scaling and benchmarking self-supervised visual representation learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 6391–6400.
[47] H. Lee, S. J. Hwang, and J. Shin, "Rethinking data augmentation: Self-supervision and self-distillation," in *Proc. Int. Conf. Learn. Representations*, 2019, pp. 1–11.
[48] F. Morin and Y. Bengio, "Hierarchical probabilistic neural network language model," in *Proc. Int. Workshop Artif. Intell. Statist.*, 2005, pp. 246–252.
[49] M. Gutmann and A. Hyvärinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 297–304.
[50] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.
[51] N. Parikh and S. Boyd, "Proximal algorithms," *Found. Trends Optim.*, vol. 1, no. 3, pp. 127–239, 2014.
[52] Z. Jia *et al.*, "GraphSleepNet: Adaptive spatial-temporal graph convolutional networks for sleep stage classification," in *Proc. 29th Int. Joint Conf., Artif. Intell. Org.*, 2020, pp. 1324–1330. [Online]. Available: https://doi.org/10.24963/ijcai.2020/184
[53] E. Eldele *et al.*, "An attention-based deep learning approach for sleep stage classification with single-channel EEG," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 809–818, 2021.
[54] C. R. Patti, T. Penzel, and D. Cvetkovic, "Automated sleep spindle detection using iir filters and a Gaussian mixture model," in *Proc. 37th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2015.
[55] Mélanie *et al.*, "Hierarchical clustering of brain activity during human nonrapid eye movement sleep," *Proc. Nat. Acad. Sci.*, vol. 109, no. 15, pp. 5856–5861, 2012.
[56] M. K. Scullin, T. L. Harrison, S. A. Factor, and D. L. Bliwise, "A neurodegenerative disease sleep questionnaire: Principal component analysis in Parkinson's disease," *J. Neurological Sci.*, vol. 336, no. 1–2, pp. 243–246, 2014.
[57] M. Ye, X. Zhang, P. C. Yuen, and S.-F. Chang, "Unsupervised embedding learning via invariant and spreading instance feature," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2019.
[58] Y. Cao *et al.*, "Parametric instance classification for unsupervised visual feature learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020.
[59] C. Yang, D. Xiao, M. B. Westover, and J. Sun, "Self-supervised EEG representation learning for automatic sleep staging," 2021, *arXiv:2110.15278*.
[60] Y. Zhang, Y. Chen, J. Wang, and Z. Pan, "Unsupervised deep anomaly detection for multi-sensor time-series signals," *IEEE Trans. Knowl. Data Eng.*, pp. 1–14, 2021, doi: 10.1109/TKDE.2021.3102110.
[61] E. Eldele *et al.*, "Time-series representation learning via temporal and contextual contrasting," 2021, *arXiv:2106.14112*.
[62] M. B. Pouyan, J. Birjandtalab, M. Heydarzadeh, M. Nourani, and S. Ostadabbas, "A pressure map dataset for posture and subject analytics," in *Proc. IEEE EMBS Int. Conf. Biomed. Health Inform.*, 2017, pp. 65–68.
[63] O., Y. Huang, D. Forger, and C. Goldstein, "Sleep stage prediction with raw acceleration and photoplethysmography heart rate data derived from a consumer wearable device," *SLEEP*, vol. 42, no. 12, 2019, Art. no. zsz180.
[64] A. Tataraidze *et al.*, "Bioradiolocation-based sleep stage classification," in *Proc. 38th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2016, pp. 2839–2842.
[65] L. V. D. Maaten and H. Geoffrey, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 2605, pp. 2579–2605, 2008.
[66] D. Zhang, H. Tian, and J. Han, "Few-cost salient object detection with adversarial-paced learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 12236–12247.
[67] D. Zhang, J. Han, L. Yang, and D. Xu, "SPFTN: A joint learning framework for localizing and segmenting objects in weakly labeled videos," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 475–489, Feb. 2020.

Aite Zhao received the bachelor's degree in software engineering from the Qingdao University of Technology, Qingdao, China, in 2013, and the Ph.D. degree from the College of Information Science and Engineering, Ocean University of China, Qingdao, China, in June 2020. She is currently a Visiting Ph.D. Researcher with the School of Informatics, University of Leicester, Leicester, U.K. She is also a Lecturer with the College of Computer Science and Technology, Qingdao University, Qingdao, China. Her research interests include computer vision, pattern recognition, machine learning, data analysis, and robotics.



Yue Wang received the bachelor's degree in computer science and technology from Harbin Normal University, Harbin, China, in June 2020. She is currently working toward the master's degree with Qingdao University, Qingdao, China, majoring in computer technology. Her research interests include computer vision, pattern recognition, machine learning, and traffic flow prediction.

Jianbo Li received the bachelor's and master's degrees from the Department of Computer Science, Qingdao University, Qingdao, China, in 2002 and 2005, respectively, and the Ph.D. degree from the Department of Computer Science, University of Science and Technology of China, Hefei, China, in 2009. He is currently a Professor and the Dean of the College of Computer Science and Technology, Qingdao University, Qingdao, China. His research interests include urban computing, mobile social networks, and machine learning.