A New Post-Processing Method using Latent Structure Influence Models for Channel Fusion in Automatic Sleep Staging

Sajjad Karimi, *Student Member, IEEE,* and Mohammad Bagher Shamsollahi, *Senior Member, IEEE*

***Abstract*— Human sleep stage dynamics can be adequately represented using Markov chain models, and the accuracy of sleep stage classification can be improved by considering these dynamics. The present study proposes a new post-processing method based on channel fusion using Latent Structure Influence Models (LSIMs). LSIMs can simultaneously model sequences of different channels to have a nonlinear dynamical fusion based on sleep stage dynamics. The proposed method develops and examines two channel-fusion algorithms: the standard LSIM fusion and the integrated LSIM fusion, in which the latter is more efficient and performs better. The proposed LSIM-based method simultaneously incorporates the nonlinear interactions between channels and the sleep stage dynamics. In the first step, existing sleep staging systems process every data channel independently and produce stage score sequences for each channel. These single-channel scores are then projected into belief space using the marginal one-slice parameter of all channels by LSIM fusion algorithms. The logarithms of marginal one-slice parameters are concatenated to obtain log-scale belief state space (LBSS) features in the standard LSIM fusion. In the integrated LSIM fusion, integrated LBSS (ILBSS) features are formed by combining the LBSS features of several LSIMs. Finally, a KNN classifier maps the LBSS and ILBSS features onto the sleep stages. By utilizing four recently developed sleep staging systems, the proposed method is applied to the publicly available SleepEDF-20 database that contains five AASM sleep stages (N1, N2, N3, REM, and W). Compared to single-channel (Fpz-Cz, Pz-Oz, and EOG) results, integrated LSIM fusion results have a statistically significant improvement of 1.5% in 2-channel fusion (Fpz-Cz and Pz-Oz) and 2.5% in 3-channel fusion (Fpz-Cz, Pz-Oz, and EOG). With an overall accuracy of 87.3% for 3-channel post-processing, the integrated LSIM fusion system offers one of the highest overall accuracy rates among existing studies.**

***Index Terms*— Latent Structure Influence Model, Automated Sleep Staging, Deep Learning, Wavelet Scattering Transform, Channel Fusion, EEG.**

I. Introduction

SLEEPING is as important as eating or breathing for humans, and it is an essential process to preserve human health. Several psychophysiological states occur in a healthy human brain during sleep, referred to as sleep stages [1]. Clinical evaluation of sleep usually involves scoring the sleep stages using overnight polysomnography (PSG) based on two well-known standard criteria, including Rechtschaffen-Kales (R&K) [2] and the American Academy of Sleep Medicine (AASM) [3]. AASM generalizes the R&K standard and recognizes five sleep stages, stage N1, stage N2, stage N3, stage rapid eye movements (REM), and wakefulness (W). Sleep scoring through visual inspection is excessively expensive and time-consuming, with variability in the agreement between experts [4]. Therefore, numerous works have aimed to automate this task using artificial intelligence (AI) systems in the past decades [5]–[9]. To date, various automated sleep stage scoring approaches have been developed to replicate expert labels with a restricted level of success. Improving sleep stage scoring relies mainly on preprocessing, feature generation, classifiers, and post-processing [10]–[12]. Following is an overview of existing approaches for preprocessing, features generation, and classifiers. After that, we discuss how Markov models can be used for post-processing as our contribution to the field of PSG-based sleep staging in the current paper.

Conventional feature-based machine learning and deep learning models are common approaches in automatic sleep staging systems. The performance of the conventional machine learning and deep learning systems is affected by the largeness of the data and the capability of the feature extraction method. Conventional machine learning models are better suited to small data sets than deep learning models. The performance of the machine learning and deep learning systems is affected by the largeness of the data and the capability of the feature extraction method. Machine learning models are better suited to small data sets than deep learning models. Conventional systems are developed based on classical feature generation and a subsequent classifier like support vector machine (SVM) or k-nearest neighbors (KNN) [13]. These systems depend on hand-engineered features that require prior sleep analysis knowledge. Empirical mode decomposition (EMD) and wavelet analysis are effective approaches to extracting sleep staging features [14]–[16]. They generally do not encode the dynamic of sleep stage transitions into their extracted features, which is important for identifying the stages of neighboring epochs. In contrast, deep learning systems automatically learn features from low-level signals or time-frequency images with a softmax layer as the classifier at the last stage [17]. Due to ability of deep learning systems to extract useful information from raw signal data, they have become popular for automatic sleep stage scoring [18]. Deep learning systems can be categorized according to the signal input types. One category processes 1-dimensional signals directly, while the other one uses 2-dimensional time-frequency images [17]. Early attempts at automatic sleep staging by deep learning have evolved rapidly in targeted modeling methodologies and effective network architectures. One-to-one and many-to-one methods are gradually being replaced by one-to-many (i.e., multitasking) and many-to-many (i.e., sequence-to-sequence) methods to better represent sleep data [6]. Following is a review of several deep learning studies on sleep staging.

In [19], the complex Morlet wavelet is used to extract time-frequency features. These features are given to a stacked sparse autoencoder to reduce the feature dimension. In [20], features are automatically learned by convolutional neural networks (CNNs) for classification without using prior domain knowledge. Another study utilizes CNNs to extract time-invariant features, and a bidirectional Long Short-Term Memory (LSTM) automatically learns transition rules among sleep stages from EEG epochs [7]. A hierarchical recurrent neural network named SeqSleepNet is proposed in [6] to perform a sequence-to-sequence classification that received a sequence of multiple epochs as input and classified them at once. Several studies use modern spectral transformations within conventional algorithms in sleep staging, including tunable Q-factor wavelet transform (TQWT), amplitude-modulated & frequency-modulated (AM-FM) components, and deep scattering transform [9], [14], [21], [22]. In [9], the feature extraction part of the algorithm is based on two advanced signal processing tools, including the scattering transform and diffusion

M. B. Shamsollahi and S. Karimi are with the Biomedical Signal and Image Processing Laboratory (BiSIPL), School of Electrical Engineering, Sharif University of Technology, Tehran (009821), Iran (e-mail: mbshams@sharif.edu). Codes will be available at https://github.com/sajjadkarimi91/lsim-sleep-fusion.

map. Study [9] performs sleep stage scoring by applying a multiview diffusion map (MDM) to fuse spectral information of two EEG channels. MDM has higher overall accuracy than the diffusion map of single channels.

Deep learning and conventional systems focus on generating and classifying features, but they do not directly consider sleep stage dynamics and channel interactions within the PSG recording. Researches on sleep stage dynamics suggest that a Markov chain model represents human sleep stages fairly well [23], and several sleep staging systems use hidden Markov models (HMMs) to analyze sleep stage transition patterns [5], [24], [25]. However, existing HMM-based studies only concatenate features of different PSG channels without considering interactions between them [5], [25], [26]. HMMs are used to consider sleep stage dynamics, and HMM parameters are learned by the supervised algorithm or the Baum-Welch algorithm [5]. For example, an HMM-based post-processing method has been developed to incorporate the temporal structure of sleep cycles into sleep stage scoring to reduce false positives in the classification [8]. This method utilizes a supervised HMM on a single-channel EEG.

Previous studies have examined the brain connectivity during sleep between patients with psycho-physiological insomnia (PPI) and controls using EEG channel interactions [27], [28]. According to their findings, channel communication and interactions during sleep may indicate a sleep disorder caused by PPI. While existing HMM-based sleep staging systems or post-processing methods incorporate sleep stage dynamics to improve performance, they still can not consider the PSG channel interactions. Suppose several interacting channels generate a multi-channel time series. In that case, an HMM with a single hidden variable is adequate, and latent structure influence models (LSIMs) are better than HMMs due to considering multiple interacting channels. LSIMs are a particular kind of coupled hidden Markov models (CHMMs) that consider the influence model's interactions with hidden Markov chains [29], [30]. We solved the inference and learning problems for LSIMs and indicated that LSIMs have superior performance compared to HMMs for processing multi-channel time series [30], [31]. While sequence-to-sequence (LSTM and RNN) deep learning methods consider sleep dynamics, they cannot handle PSG channel fusion effectively based on the current study results. The current study addresses these limitations of existing deep systems using LSIMs to propose a new post-processing method with a nonlinear fusion of channels considering sleep stage dynamics.

The current study focuses on a new post-processing method to improve the accuracy of existing sleep staging systems, and the contributions can be summarized as follows.

- A new post-processing method is proposed that considers nonlinear channel interactions and sleep stage dynamics concurrently using LSIMs.
- We present a new unsupervised approach to extract features in the belief state space using the inference algorithm of LSIMs called log-scale belief state space (LBSS) features. These new features improve class separability and enhance classification accuracy for existing sleep staging systems.
- A standard LSIM fusion and an integrated LSIM fusion are developed and examined based on LBSS features. Integrated LSIM fusion is an extension of standard fusion that considers several LSIMs with different hyperparameters, including hidden states and Gaussian numbers.
- We propose the integrated LBSS features assembled from multiple LSIMs with different states and Gaussian numbers. The advantage of integrated LBSS is that different dynamic sleep stages from different LSIMs are involved in the fusion.

The rest of this manuscript is structured as follows. In section II, we first review several state-of-the-art sleep staging systems. Then, we describe the proposed post-processing method to fuse spatiotemporal information of EEG & EOG channels in automated sleep staging. The proposed method is applied to a publicly available sleep dataset for several sleep staging systems. Section III presents its results compared to single channels, followed by conclusion in section IV.

II. Materials & Methods

This section describes how LSIMs are used to fuse channels and model the dynamics of sleep stages. Several state-of-the-art sleep staging systems are considered to determine the effectiveness of the proposed post-processing method. Each sleep staging system processes every PSG channel independently, and we use sleep stage scores or probabilities as the input of the proposed post-processing. Afterward, we describe how the proposed post-processing method analyzes the sequences of scores for multiple channels to improve the classification accuracy of each system. In the following sections, we briefly review several employed sleep staging systems, followed by a detailed description of the proposed LSIM-based fusion method.

A. Selected Deep Spectrum & Deep Learning Systems

The following presents four recent innovative sleep staging systems as baseline systems. In the current study, deep learning staging systems are applied independently to every channel. The logarithm of the last softmax layer values (logits) is used as input to the proposed post-processing method.

1) Diffusion Geometry of Deep Scattering Spectrum (DGDSS) [9]: This system has three main parts: feature generation, extraction of intrinsic sleep dynamics, and the learning step. Feature generation is performed by deep scattering transform [32] to extract spectral features from individual channels. Deep scattering transform is an enhanced spectral representation and is computed by cascaded multi-stage of wavelet convolutions and modulus operators [33]. The scattering features are the logarithm of renormalized coefficients, as recommended in [33]. Morlet mother wavelet is applied to 90 seconds interval that slides every 30 seconds. Setting other scattering parameters gives a sequence of 1278-dimensional feature vectors associated with expert labels on 30 seconds intervals.

DGDSS uses the diffusion map algorithm to reduce the dimension of scattering features. This part transforms the 1278-dimensional scatter features to 80-dimensional intrinsic sleep features (see Fig. 1). Intrinsic sleep features (80-dimensional) are input features of a standard kernel SVM with radial basis function as the learning step. The binary SVM is generalized to the 5-class SVM by the one-versus-one classification scheme. Outputs of the 5-class SVM include an estimated label and a 5-dimensional vector filled with negative hinge loss values (NHLVs) [34]. NHLVs are also known as SVM scores, and the sequences of these 5-dimensional vectors are collected in a multi-channel time-series for EEG channels. Fig. 1 displays DGDSS parts for a single-channel EEG recording. We refer the interested readers to [9] for further technical details. DGDSS is applied to every single channel, and then LSIM fusion algorithms work on the multi-channel time-series of NHLVs.

2) SeqSleepNet [6]: In this model, a hierarchical recurrent neural network is used to tackle the data classification problem, receiving a sequence of multiple epochs as input and determining their labels simultaneously. The goal of the computation is to identify a sequence of PSG epochs of length L represented by $(S_1, S_2, ..., S_L)$ that maximizes the conditional probability $p(y_1, y_2, ..., y_L|S_1, S_2, ..., S_L)$ [6].

On the epoch-processing level, the network incorporates a filter-bank layer designed to learn frequency-domain filters for preprocessing and an attention-based recurrent layer designed for short-term sequential modeling. Sequence processing involves placing a

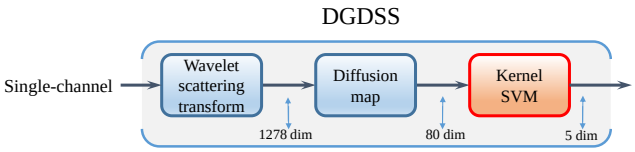

Fig. 1. Feature extraction and classification in DGDSS with 5-dimensional vectors of NHLVs as outputs (red blocks use train labels).

recurrent layer on top of the epoch-wise learned features for long-term modeling of sequential epochs. The output vectors of the top recurrent layer are then classified at every time step to generate the sequence of output labels.

3) TinySleepNet [35]: Compared to the existing deep models, this one consists of fewer parameters to train, requiring a smaller amount of training data and computational resources. TinySleepNet is a novel technique to effectively train the model end-to-end based on raw single-channel EEG. The model becomes more robust to shifts along the time axis through data augmentation and prevents it from remembering the sleep stages sequences [35]. TinySleepNet is an improved version of DeepSleepNet that uses raw single-channel EEGs to score sleep stages automatically. Compared to DeepSleepNet [7], TinySleepNet focuses primarily on an efficient model architecture that significantly reduces the number of parameters and computing resources necessary for the EEG analysis. Also, it proposes an innovative training method that uses data augmentation to generate different sets of training data for different training epochs to avoid overfitting.

CNNs are the first part of the network close to the input signals. There are four convolutional layers as well as two max-pooling and two dropout layers in the CNNs. In contrast to DeepSleepNet, TinySleepNet uses only one CNN branch instead of two branches. The second part of the network is a unidirectional RNN consisting of a single LSTM layer, followed by a dropout layer.

4) XSleepNet [17]: Using multi-view signals (i.e., raw signals and time-frequency images) can make sleep staging tricky and challenging to understand. With XSleepNet, a sequence-to-sequence sleep staging model can be learned from raw signals and time-frequency images to create a joint representation of sleep stages. Since different views may generalize or overfit at different rates, XSleepNet is trained such that the learning pace on each view is adapted based on their generalization/overfitting behavior [17]. As a multi-view model, XSleepNet is composed of two network streams: one for the raw signal and the other for the time-frequency image.

The principle behind this model is robustness (to the amount of training data) and complementarity (i.e., how the two input views complement one another) in the network while maintaining flexibility to learn from multiple views effectively.

B. LSIMs

LSIMs are suitable tools for modeling multi-channel time-series dependent on time and space, and they are capable of detecting nonlinearity and nonstationarity, making them well suited for multi-channel brain signals. LSIMs use the influence model to consider coupled Markov chains and have an interpretable and linear-wise structure in state-space [30]. Like HMMs, inference and learning problems are essential in using LSIMs in practice. Following are some definitions and explanations about notations of LSIMs [30], and then there is an overview of existing solutions for inference and learning problems [30], [31].

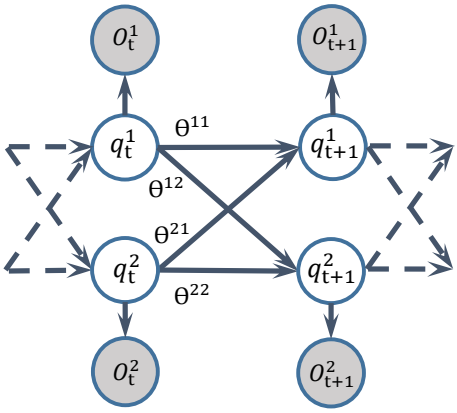

Fig. 2. Representation of graphical model for a 2-channel LSIM.

Assuming that there exists a C-channel LSIM and its associated multi-channel time-series, we denote $S^c = \{S_1^c, S_2^c, ..., S_{M(c)}^c\}$ to be state space of channel c in the LSIM, and $q_t^c \in S^c$ and $o_t^c \in \mathbb{R}^{L(c)}$ are state and observation of channel c at time t, respectively ($L(c)$ is dimension). There is also a simplifying definition as $v_t^c(m) = \{q_t^c = S_m^c\}$. Initial state probabilities of each channel are denoted by $\pi_m^c = P(q_1^c = S_m^c)$. In addition, $a_{m,n}^{c,\xi} = P(q_t^\xi = S_n^\xi | q_{t-1}^c = S_m^c)$, and $\theta^{c,\xi}$ is coupling weight from channel c to channel ξ. The sets of all initial probabilities, transition matrices and coupling weights are denoted by π, A and Θ respectively. The emission probabilities of the observation given its hidden state are written as $b_m^c(o_t^c) = f(o_t^c | q_t^c = S_m^c)$ where o_t^c may be either discrete or continuous. In current study, observations are continuous amplitude, and emission probabilities $b_m^c(o_t^c)$ belong to Gaussian Mixture Model (GMM) families with $D(c)$ components. Sets of all mixing weights, means and covariance matrices are also denoted by ω , μ and Σ. Thus, LSIM parameters are indicated by $\lambda = \{\pi, A, \Theta, \omega, \mu, \Sigma\}$. The influence model factorizes transition matrices of coupled Markov chains as follows

$$P(q_t^\xi | q_{t-1}^1, ..., q_{t-1}^C) = \sum_{c=1}^{C} \theta^{c,\xi} P(q_t^\xi | q_{t-1}^c),$$
$$\theta^{c,\xi} \geq 0, \ \sum_{c=1}^{C} \theta^{c,\xi} = 1. \tag{1}$$

Set of observations at time t is denoted by $o_t = \{o_t^1, o_t^2, ..., o_t^C\}$ and $o_{t_s:t_p} = \{o_{t_s}, o_{t_s+1}, ..., o_{t_p}\}$ indicates an interval of observations. Fig. 2 shows the structure of a 2-channel LSIM. In this study, observations of LSIMs are the sequences of NHLVs or logits. As pointed out before, DGDSS or deep models provides a vector of NHLVs or logits for each 30 seconds interval per channel, and this vector is viewed as a feature vector for the proposed fusion algorithm.Arranging these vectors along the time horizon leads to a 5-dimensional time series per channel with an equal duration of expert labels for each subject. We denote the groups of time series at all channels and subjects by $o_{1:T}$, where T is the total number of epochs across all subjects.

Inference and learning are two important LSIM problems that must be addressed to make them useful. The inference is defined as computing the probability distribution over hidden states given an interval of observations. The belief state space is the space of probability distributions for hidden states. The learning problem consists of selecting optimal parameters for a multi-channel time series observation to maximize an appropriate criterion. There is a

comprehensive framework with many formulas and recursive relationships for indicating the solution of inference and learning problems in LSIMs. The following is a summary of those solutions, and the equations may be challenging to follow.

The inference is performed by the marginal forward, one-slice, and backward parameters that are respectively defined as follows [30]

$$\begin{aligned} \alpha^{\xi}_{t|t-1}(m) &= P(v^{\xi}_t(m)|o_{1:t-1}) \\ \alpha^{\xi}_{t|T}(m) &= P(v^{\xi}_t(m)|o_{1:T}) \\ \beta^{\xi}_t(m) = \frac{\alpha^{\xi}_{t|T}(m)}{\alpha^{\xi}_{t|t-1}(m)} &= \tilde{b}^{\xi}_m(o^{\xi}_t) \times \frac{f(o_{t+1:T}|v^{\xi}_t(m), o_{1:t})}{f(o_{t+1:T}|o_{1:t})}. \end{aligned} \tag{2}$$

where,

$$\tilde{b}^{\xi}_m(o^{\xi}_t) = \frac{b^{\xi}_m(o^{\xi}_t)}{f(o^{\xi}_t|o_{1:t-1})}. \tag{3}$$

These parameters are calculated recursively by an approximate inference algorithm that considers the following main equations [30]

$$\begin{aligned} \alpha^{\xi}_{t|t-1}(m) &= \sum_{c=1}^{C} \theta^{c,\xi} \sum_{n_c=1}^{M(c)} a^{c,\xi}_{m,n_c} \alpha^{c}_{t-1|t-2}(n_c) \tilde{b}^{c}_{n_c}(o^{c}_{t-1}), \\ \beta^{\xi}_t(m) &= \tilde{b}^{\xi}_m(o^{\xi}_t) \sum_{w=1}^{C} \hat{d}^{\xi,w}_t \sum_{n_w=1}^{M(w)} \rho^{\xi,w}_{t+1}(m, n_w) \beta^{w}_{t+1}(n_w). \end{aligned} \tag{4}$$

where,

$$\begin{aligned} \rho^{\xi,w}_{t+1}(m, n_w) &= \alpha^{w}_{t+1|t}(n_w) + \\ \theta^{\xi,w}[a^{\xi,w}_{m,n_w} &- \sum_{m_\xi=1}^{M(\xi)} a^{\xi,w}_{m_\xi,n_w} \alpha^{\xi}_{t|t-1}(m) b^{\xi}_m(o^{\xi}_t)] \end{aligned} \tag{5}$$

LSIM parameters λ are learned by applying a recently developed re-estimation algorithm to $o_{1:T}$ [31]. This algorithm requires another parameter called the two-slice parameter. The two-slice parameter $\Gamma^{c,\xi}_t(n_c, n_\xi)$ is the joint distribution of two successive hidden states from different or same channels given all observations [31]. This parameter is defined by

$$\Gamma^{c,\xi}_t(n_c, n_\xi) = \tilde{b}^{c}_{n_c}(o^{c}_t) \alpha^{c}_{t-1|t-2}(n_c) \theta^{c,\xi} a^{c,\xi}_{n_c,n_\xi} \beta^{\xi}_t(n_\xi). \tag{6}$$

Parameters of the influence model are re-estimated using the two-slice parameter through [31]

$$\begin{aligned} \theta^{c,\xi} &= \frac{\sum_{n_\xi=1}^{M(\xi)} \sum_{n_c=1}^{M(c)} \sum_{t=1}^{T} \Gamma^{c,\xi}_t(n_c, n_\xi)}{\sum_{s=1}^{C} \sum_{n_\xi=1}^{M(\xi)} \sum_{n_s=1}^{M(s)} \sum_{t=1}^{T} \Gamma^{s,c}_t(n_s, n_\xi)}, \\ a^{c,\xi}_{n_c,n_\xi} &= \frac{\sum_{t=1}^{T} \Gamma^{c,\xi}_t(n_c, n_\xi)}{\sum_{n=1}^{M(\xi)} \sum_{t=1}^{T} \Gamma^{c,\xi}_t(n_c, n)}. \end{aligned} \tag{7}$$

C. Proposed Post-processing Framework

Nonlinear and nonstationary EEG signals provide information about spatial and temporal patterns of brain electrical activity. Studies on functional connectivity suggest that brain regions have temporal dependencies. Therefore, LSIMs are suitable candidates for analyzing multi-channel brain signals. LSIMs can be used to analyze temporal dynamics and spatial interactions of PSG channels in sleep staging.

In LSIMs, coupling weights $\theta^{c,\xi}$ determine the structure of the nonlinear interaction between channels, and transition probabilities $a^{c,\xi}_{n_c,n_\xi}$ indicate dynamical transitions between hidden states. Thus, the influence model provides a nonlinear dynamical fusion of channels. Our study proposes a new post-processing method that maps the score sequences of channels to belief state space using LSIMs.

Two fusion algorithms are proposed to enhance the accuracy of single channels processed by its baseline sleep staging system. The raw PSG signals are divided into 30-second epochs in automated sleep staging, and a baseline sleep staging system assigns a 5-dimensional stage score to each epoch. It is important to note that each PSG channel is processed by its trained baseline system. In the first fusion algorithm called standard LSIM fusion, an LSIM is considered to model the sequences of scores (NHLVs or logits) of PSG channels, and LSIM channels are the same as PSG channels (see Fig. 3). Therefore, each channel of LSIM is composed of 5-dimensional observations, and there are two and three channels in the selected sleep dataset of the current study. LSIM parameters are learned unsupervised for modeling score sequences of different channels using a baseline sleep staging system. The LSIM learning algorithm adjusts the parameters to maximize the observation likelihood. Therefore, the parameters are optimized to consider the sleep dynamic between epochs and channel interaction. Two hyperparameters in the LSIM learning algorithm, including state numbers $M(c)$ and Gaussian numbers $D(c)$, must be set before learning. LSIMs have an inherent difficulty in parameter learning since there are multiple channels with different states and Gaussian numbers. The grid-search size increases exponentially with the number of channels to select the optimal hyperparameters. We assume all channels have the same state and Gaussian numbers to simplify and reduce the hyperparameter space.

We propose a novel feature extraction approach in the belief state space based on its ability to generate a nonlinear dynamical fusion of score sequences after learning LSIM parameters. The inference algorithm performs this feature extraction by computing probability distributions over hidden states for a given observation. Marginal forward and one-slice parameters are a representation of belief state space. As demonstrated in (2), the marginal one-slice parameter provides a better representation than the marginal forward parameter because it considers past, present, and future observations, whereas the marginal forward parameter only considers past observations. A detailed formula of the marginal one-slice parameter from its definition is as follows [30]

$$\begin{aligned} \alpha^{\xi}_{t|T}(m) &= \alpha^{\xi}_{t|t-1}(m) \beta^{\xi}_t(m) \\ &= \left(\tilde{b}^{\xi}_m(o^{\xi}_t) \sum_{c=1}^{C} \theta^{c,\xi} \sum_{n_c=1}^{M(c)} a^{c,\xi}_{m,n_c} \alpha^{c}_{t-1|t-2}(n_c) \tilde{b}^{c}_{n_c}(o^{c}_{t-1}) \right) \times \\ &\quad \left(\sum_{w=1}^{C} \hat{d}^{\xi,w}_t \sum_{n_w=1}^{M(w)} \rho^{\xi,w}_{t+1}(m, n_w) \beta^{w}_{t+1}(n_w) \right). \end{aligned} \tag{8}$$

The marginal one-slice parameter contains the effect of both marginal forward and backward parameters, and it transfers channel observations into belief state space. The first part of (8) shows the dynamical effect of past and present stage scores on the marginal one-slice parameter, and $\theta^{c,\xi}$ also shows the nonlinear interaction from past stage scores of channel c. The second part of (8) indicates the dynamical modification of future stage scores in the present marginal one-slice parameter, and $\hat{d}^{\xi,w}_t$ shows the resulting nonlinear interaction from the future stage scores of channel w. In addition, the LSIM learning algorithm is unsupervised and does not require true stage labels. Thus, the marginal one-slice parameter can be viewed as an unsupervised nonlinear dynamical fusion that projects NHLVs or logits to the belief state space. We define the log-scale belief state

space (LBSS) feature vector of LSIM for epoch t in sequence $o_{1:T}$ as follows

$$LBSS_t = \log\Big(\big[\alpha^1_{t|T}(1), \alpha^1_{t|T}(2), ..., \alpha^1_{t|T}(M(1)), \\ \alpha^2_{t|T}(1), \alpha^2_{t|T}(2), ..., \alpha^C_{t|T}(M(C)-1), \alpha^C_{t|T}(M(C))\big]\Big). \quad (9)$$

Because each observation sample (stage scores) is linked to its hidden state (see Fig. 2), the number of LBSS features is the same as the number of epochs. The dimension of LBSS features is equal to the sum of all channel hidden states $\sum_{c=1}^{C} M(c)$. For example, if there is a 2-channel LSIM with 10 hidden states per channel, the dimension of LBSS features is 20. Fig. 3 shows a block diagram of the new post-processing method, with LBSS features as the LSIM inference block outputs. In the final step, a KNN with Euclidean distance is applied to LBSS features.

Standard LSIM fusion has three hyperparameters, including hidden state numbers, Gaussian numbers, and the K-parameter of KNN. We need to determine the optimal LSIM structure and K-parameter based on the training data. Thus, the training set is divided into a smaller train set and a validation set. Each K-parameter has several validation accuracy values based on different LSIMs, and the average validation accuracy is calculated by averaging them all together. The K-parameter is determined using a simple grid search to maximize the average validation accuracy. After finding the optimum K-parameter, hidden states and Gaussian numbers are selected as the best arguments for validation accuracy with the optimum K-parameter. Then, these optimum hyperparameters are applied to the test dataset.

Standard LSIM fusion selects only one optimum LSIM to analyze train and test datasets. LSIMs with different hidden states and Gaussian numbers can cover a broader range of dynamics and interactions. Therefore, we presented another approach that concatenates the LBSS features of multiple LSIMs, referred to as integrated LSIM fusion. Fig. 4 shows the integrated LSIM fusion framework with N different LSIMs. This fusion makes integrated LBSS (ILBSS) features from multiple N LSIMs as follows

$$ILBSS_t = [LBSS^1_t, LBSS^2_t, ..., LBSS^N_t], \quad (10)$$

where $LBSS^k_t$ denotes the LBSS features of LSIM number k. Integrated fusion offers two main advantages compared to the standard version. The first advantage is that because multiple LSIMs are used, their LBSS features cover different dynamics. Second, using multiple LSIMs instead of single selected LSIMs will reduce the biases and increase the generality. Integrated LSIM fusion has only the K-parameter of KNN as a hyperparameter. Due to the integration of all LSIMs within ILBSS features, average validation accuracy is not required in this case. This hyperparameter is determined using a simple grid search on the validation accuracy.

III. Performance Assessment

A. Experimental setup and Statistics

To evaluate the performance of the proposed post-processing method, we consider a commonly considered benchmark databases SleepEDF-20, from the public repository Physionet [36]. This is the Sleep Cassette (SC*) subset of the Sleep-EDF Expanded dataset. The subset SC* contains PSG recordings from 20 subjects without sleep-related medication aged 25-34 (28.7±2.9). Two consecutive day-night PSG recordings were collected for each subject, except for subject 13. The PSG recordings contain two scalp-EEG signals from the Fpz-Cz and Pz-Cz channels, one EOG (horizontal), one EMG, and one oro-nasal respiration signal. EEG and EOG were collected at the same sampling rate of 100 Hz. According to the R&K standard, sleep experts manually classified these recordings into one of eight classes (W, N1, N2, N3, N4, REM, MOVEMENT, UNKNOWN). The Fpz-Cz, Pz-Cz, and EOG channels are used to evaluate models without further processing. In addition, the N3 and N4 stages are merged into one stage N3, in accordance with the AASM standard. Since we are interested in sleep periods, we only consider 30 minutes of such periods just before and after sleep. Because MOVEMENT and UNKNOWN stages are not the sleep stages, we excluded them.

There are long wakefulness periods at the start and the end of recordings when a subject is not sleeping. To have a reasonable comparison with existing studies, we consider 30 minutes of such periods before and after the sleep periods. Some recent studies considered 30 minutes before and after the lump-off-on, which causes more wakefulness stages. Due to this, there are some deviations in our reproduced results from the original papers.

There are two general cross-validations (CV) schemes for automatic sleep stage scoring in literature, including leave-one-subject-out CV (LOSOCV) and non-leave-one-subject-out CV (non-LOSOCV) [7]. LOSOCV is close to the real-world scenario, and the training set and the validation set contain different subjects. This scheme considers the inter-individual variability, and the computed accuracy is acceptable for new arrival subjects from a given annotated database [9]. By contrast, training and testing sets are dependent on the non-LOSOCV scheme, and the effect of this dependency might exaggerate the performance. To have a thorough evaluation of the proposed algorithm and consider inter-individual variability, this study focuses on AASM scoring with leave-one-subject-out cross-validation [6], [7], [9].

All performance measurements used in the current study are computed through the unnormalized confusion matrix $M \in R^{5\times5}$. For $1 \le p; q \le 5$, the entry M_{pq} represents the number of expert-assigned p-class epochs, which were predicted to the q-class. The F1-score ($F1_p$) of the p-th class, is computed based on precision (PR_p) and recall (RE_p) through

$$PR_p = \frac{M_{pp}}{\sum_p M_{pq}},\ RE_p = \frac{M_{pp}}{\sum_q M_{pq}},\ F1_p = \frac{2PR_pRE_p}{PR_p + RE_p}. \quad (11)$$

The overall accuracy (ACC) and macro F1-score ($MF1$) are computed respectively through

$$ACC = \frac{\sum_p M_{pp}}{\sum_{p,q} M_{pq}},\ MF1 = \frac{1}{5}\sum_p F1_p. \quad (12)$$

In this study, the overall accuracy and macro F1-score are used as performance criteria [9]. These provide a comprehensive comparison with other studies in this field.

B. Results

This section reports the results of applying the proposed LSIM fusion to the Sleep-EDF SC* database using LOSOCV (20-fold CV). The four automated sleep staging systems are applied independently to three channels of the PSG dataset, namely Fpz-Cz, Pz-Oz, and EOG. These systems include DGDSS, TinySleepNet, SeqSleepNet, and XSleepNet. The total number of epochs in this dataset is 42308 for all systems.

In TABLE I, the first three columns indicate the overall accuracy and macro F1-score for different channels and sleep staging systems. As expected from the literature, the Fpz-Cz has the best results compared to Pz-Oz and EOG for this dataset. In agreement with the literature, XSleepNet also performs well on single channels. TinySleepNet, SeqSleepNet, and XSleepNet achieved slightly lower

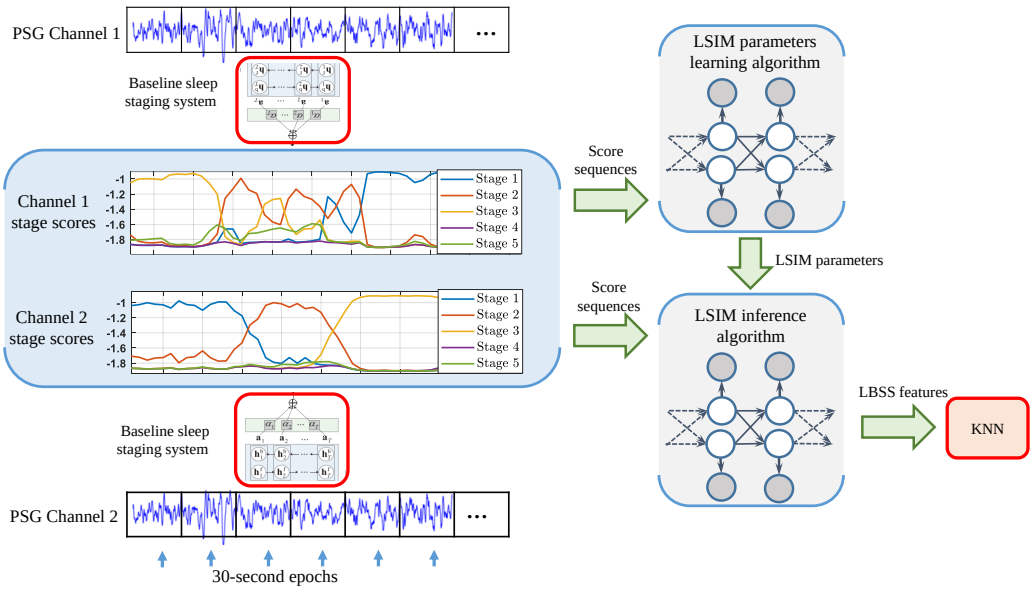

Fig. 3. The structure of proposed post-processing method for 2-channel PSG with the standard LSIM fusion (red blocks use train labels).

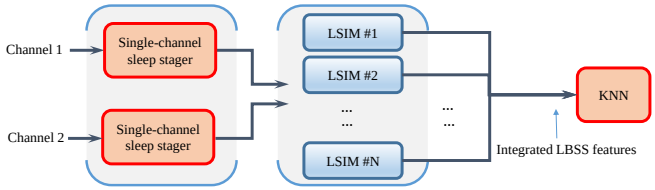

Fig. 4. The proposed post-processing method for 2-channel PSG with the integrated LSIM fusion (red blocks use train labels).

results than the original papers (Fpz-Cz channel). Fewer wakefulness stages (see section III-A) or random parameter initialization in deep learning may cause differences in our test dataset than in the original papers.

Based on previous studies, PSG channels are selected from SleepEDF-20 for two scenarios, including 2-channel (Fpz-Cz and Pz-Oz) and 3-channel (Fpz-Cz, Pz-Oz, and EOG). In each scenario, the 5-dimensional score sequence of a baseline system for single channels is given as a channel observation of LSIMs.

1) LBSS features visualization: A 2-channel LSIM with three states per channel for the DGDSS system allows us to visualize LBSS features in three dimensions for all subjects. This visualization of the EEG channels for all five sleep stages is shown in Fig. 5. Since the DGDSS system is effective, sleep stages are clustered well. As can be seen, LBSS features provide complementary information over different channels. For example, N2 and R stages in Fpz-Cz are neighbor clusters, while in Pz-Oz, they are well separated, or N2 and N3 are neighbors in Pz-Oz, while they are distant in Fpz-Cz.

2) Standard LSIM fusion: The number of hidden states ($M(c)$), the number of Gaussian components of each state ($D(c)$), and the K-parameter are three hyperparameters that affect the performance of standard LSIM fusion. The first two hyperparameters are considered equal for all channels and are denoted by M and D, respectively. To determine the optimal hyperparameters, we perform a grid search on M (from 5 to 25 with 5 steps), D (1 and 2), and K (from 5 to 50 with 5 steps), and these grids are selected empirically. Then the optimal hyperparameters are determined based on data-driven selection methods. We apply the reestimation algorithm to the training and testing sets (without using true labels) in each cross-validation fold to learn the LSIM parameters unsupervised for the given M and D. As discussed, the training set determines the optimal M, D, and K-parameter by using average validation accuracy. The average validation accuracy is determined by holding out four subjects from the training dataset and applying KNNs with different K-parameters. Fig. 6 shows the average validation accuracy of each K-parameter in XSleepNet for the first training fold. According to this figure, the optimal K-parameter equal to 30 gives the maximum average validation accuracy.

When the optimal K-parameter has been determined, the optimal LSIM is selected to maximize the validation accuracy. Fig. 7 shows the histogram of the optimal M values from the four sleep staging systems and the 20-fold CV (total 4x20 = 80 values). This figure shows that higher M values are selected more frequently, and complex LSIMs are preferred over simple ones in the fusion task.

TABLE I presents post-processing results with standard LSIM fusion in the "*-ch STD" columns. Compared with the best single channel, there is a performance improvement for 2-channel standard LSIM fusion, and 3-channel fusion provides better results than 2-channel fusion. TinySleepNet and XSleepNet outperform DGDSS and SeqSleepNet in the two scenarios. The 2-channel standard fusion improves accuracy by about 1.2% over the best single-channel sleep staging system. There is a performance improvement of approximately 2.1% in 3-channel fusion, which is more than the improvement in 2-channel fusion.

3) Integrated LSIM fusion: The proposed post-processing with integrated LSIM fusion starts with creating ILBSS from LSIMs with M={5,10,15,20,25} and D={1,2}, resulting in 300-dimensional features for integrated 2-channel fusion and 450-dimensional features for integrated 3-channel fusion. The K-parameter is the only hyperparameter in this fusion, and it is searched in from 10 to 100 with 10 steps. While the range and steps are considered empirically, a data-driven approach determines the optimal hyperparameter by maximizing validation accuracy.

In TABLE I, the results of integrated LSIM fusion are listed under the "*-ch INTG" columns. Performance improvements are

TABLE I

RESULTS OF SINGLE CHANNEL SLEEP STAGING SYSTEMS AND TWO PROPOSED LSIM FUSIONS IN TERMS OF THE OVERALL ACCURACY (ACC) AND MACRO F1-SCORE ($MF1$).

System	Fpz-Cz		Pz-Oz		EOG		2-ch STD		2-ch INTG		3-ch STD		3-ch INTG	
	ACC	$MF1$	ACC	$MF1$	ACC	$MF1$	ACC	$MF1$	ACC	$MF1$	ACC	$MF1$	ACC	$MF1$
DGDSS	82.9	75.8	81	72.7	78.0	70.9	84.2	76.8	84.8	77.6	85.4	78.7	85.9	79.6
TinySleepNet	83.8	79.3	81.7	76.0	77.0	70.9	86.3	81	**86.6**	**81.4**	87.0	82.0	**87.3**	**82.4**
SeqSleepNet	84.7	78.2	81.7	73.1	81.9	74.6	85.4	79.3	85.5	79.7	86.3	80.4	86.8	81.1
XSleepNet	85.5	80.1	80.8	74.5	81.2	74.0	85.7	80.2	85.9	80.4	86.3	80.7	87.0	81.5

Notes: Results in **bold** denote the best results; STD stands for the standard LSIM fusion, and INTG stands for the integrated LSIM fusion. Notation with 2-ch and 3-ch refers to (Fpz-Cz, Pz-Oz) and (Fpz-Cz, Pz-Oz, EOG).

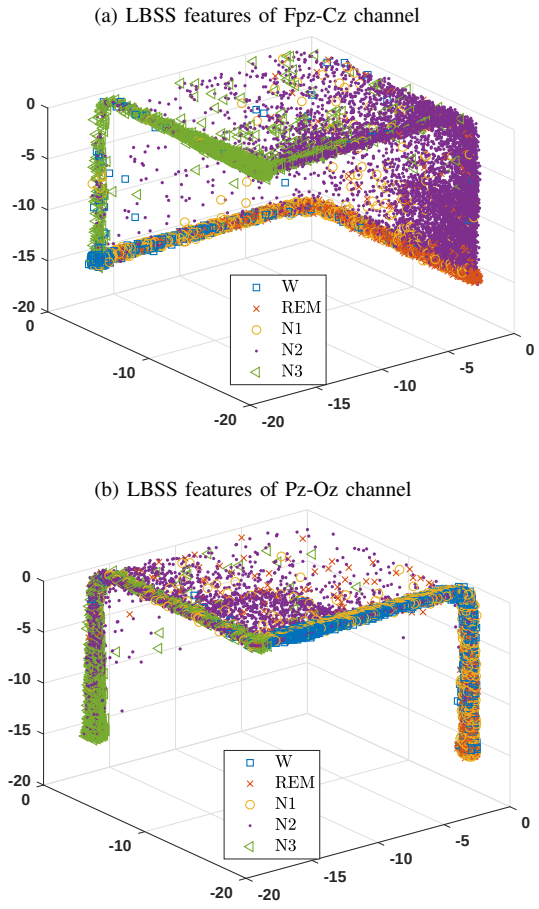

Fig. 5. A visualization of LBSS features for a standard LSIM fusion with three states per channel.

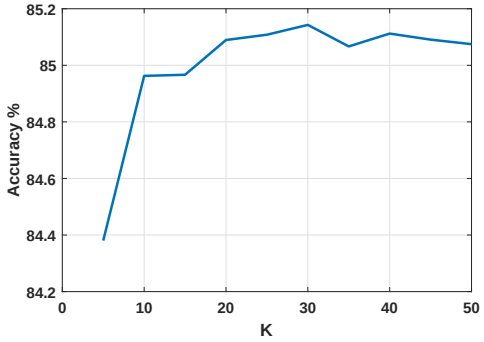

Fig. 6. Average validation accuracy for each specific K-parameter in XSleepNet for the first training fold.

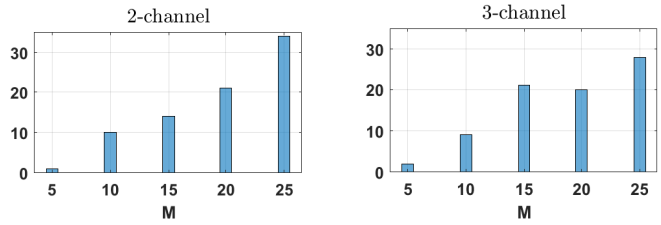

Fig. 7. Histogram of the optimal number of states for two (Fpz-Cz and Pz-Oz) and three (Fpz-Cz, Pz-Oz, and EOG) channels in the standard LSIM fusion.

again evident for LSIM fusion of two or three channels. Again, TinySleepNet and XSleepNet have superior performance for two and three channels fusion with integrated LSIM. In 2-channel integrated fusion, accuracy is improved by about 1.5% over the best single channel. In 3-channel integrated fusion, accuracy is improved by about 2.5%. As integrated LSIM fusion incorporates different dynamics from different trained LSIMs, it performs better than standard LSIM fusion. Furthermore, TABLE I indicates that integrated LSIM fusion offers robust performance regardless of the baseline systems. Applying the proposed post-processing to two and three channels from the deep base systems, overall accuracy values of 86% and 87% can be easily achieved. The integrated LSIM fusion method achieves the overall accuracy values of 86.6% for two channels and 87.3% for three channels on the Sleep-EDF SC* database. Thanks to recent deep systems that have improved sleep staging accuracy, the proposed post-processing method can achieve these accuracy levels.

In Fig. 8, the confusion matrix and per class accuracy for all sleep stages are shown for the best case (3-channel integrated LSIM fusion with TinySleepNet) in TABLE I. The wakefulness stage has the highest accuracy, at 93.9%, while the N1 stage has the lowest accuracy, at 52.3%.

Fig. 9 shows the first 500 output hypnograms for subject 2 on the first night that were processed by TinySleepNet and then post-processed using 3-channel integrated LSIM fusion. This figure also plots the output hypnograms of baseline methods for the Fpz-Cz channel. Ground truth and output hypnogram of LSIM fusion are in close alignment, as can be seen. The figure also shows that the LSIM fusion reduces the incorrect sleep stage transitions in baseline systems and effectively filters their output hypnograms.

4) Baseline systems with multi-channel inputs: A comprehensive evaluation of the proposed method must answer the question: Is integrated LSIM fusion more effective than baseline systems with multi-channel inputs? In order to answer this question, baseline systems are used with multi-channel PSG inputs instead of single channels. Both SeqSleepNet and XSleepNet can accept two and three channels as inputs, but DGDSS can handle two channels. TinySleepNet is a single-channel staging system, so it will not be discussed in this segment. The accuracy of integrated LSIM

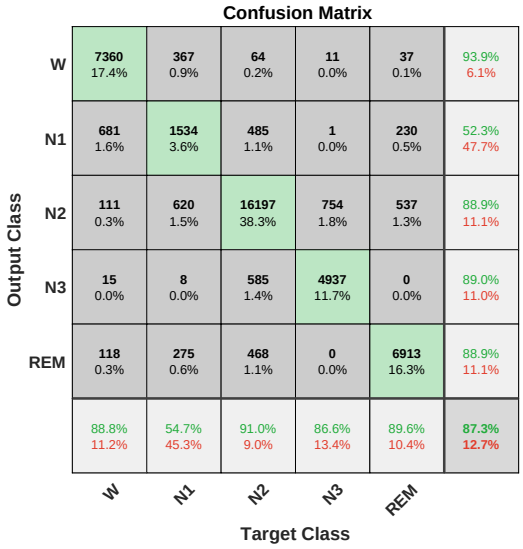

Fig. 8. The confusion matrix of proposed integrated LSIM fusion of three channels processed by TinySleepNet with an overall accuracy 87.3%.

TABLE II
THE COMPARISON BETWEEN INTEGRATED LSIM FUSION AND BASELINE SLEEP STAGING SYSTEMS WITH MULTI-CHANNEL INPUTS IN TERMS OF THE OVERALL ACCURACY (ACC).

System	2-ch base	2-ch INTG	3-ch base	3-ch INTG
DGDSS	84.4	**84.8**	-	-
SeqSleepNet	85.1	**85.5**	85.8	**86.8**
XSleepNet	83.9	**85.9**	85.3	**87.0**

Notes: 2-ch and 3-ch refer to (Fpz-Cz, Pz-Oz) and (Fpz-Cz, Pz-Oz, EOG).

fusion is compared to multi-channel baseline systems in TABLE II. This table displays the results of the baseline systems in the "*-ch base" columns. TABLE II shows that the proposed integrated LSIM fusion performs better than multi-channel baseline systems. Compared to multi-channel baseline systems, the proposed integrated fusion improves the overall accuracy order by 0.6% on two PSG channels and 1.5% on three channels.

Finally, for each sleep staging system, integrated LSIM fusion train and testing codes for all 20-fold LOSOCV on the Sleep-EDF SC* database require 260 minutes and 450 minutes (faster than deep learning systems) for two and three channels fusion, respectively, using Intel Core i9-11900K processor.

C. Discussion

TABLE I is statistically tested by the two-way analysis of variance (ANOVA), considering the four sleep staging systems as the first factor. The second factor includes single-channel processing, two-channel fusion, and three-channel fusion groups. For the integrated LSIM fusion, the p-value is 0.001, which is significant at the significance level of 0.05. A p-value of 0.003 indicates a significant result, with a significant level of 0.05 for standard LSIM fusion. Thus, accuracy improvement by the proposed post-processing method is marginal but statistically significant. According to Fig. 8, except for stage N1, the rest of the stages are well classified with accuracy above 89%. The unbalanced low sample size causes stage N1 to have the poorest accuracy, and it is commonly misclassified with neighboring stages W and N2.

In TABLE II, a paired-sample t-test is also conducted, which yields a p-value of 0.028, which is significant at the 0.05 level. This table can also compare the proposed LSIM fusion with LSTM or RNN fusion. For example, LSTM and RNN are used in XSleepNet with two or three channels inputs to process different channels and involve the dynamic of sleep stages. Based on TABLE II, LSIM fusion of channels performs better than LSTM or RNN fusion in SeqSleepNet and XSleepNet. The integrated LSIM fusion can achieve high classification accuracy thanks to the accurate single-channel processing of these deep systems.

IV. CONCLUSION

This study proposes a new post-processing method using an LSIM-based fusion algorithm to improve the accuracy of existing sleep staging systems. Single-channel scores are taken from the four state-of-the-art automated sleep staging systems, including DGDSS, TinySleepNet, SeqSleepNet, and XSleepNet. The contributions consist of fusing PSG channels and developing LBSS and ILBSS features from scores of PSG channels using LSIMs. This study aims to improve the accuracy of existing sleep staging systems by fusing different channels. The nonlinearity and dynamic nature of LSIM fusion are explained by visualizing LBSS features in belief state space. It has been shown that the proposed post-processing method with integrated LSIM fusion performs better than the standard LSIM fusion. Integrated LSIM fusion with two and three channels improves the accuracy by about 1.5% and 2.5% over the maximum accuracy of single channels in baseline systems. Improvements are consistent regardless of the number of channels or the system. Results also show that the proposed post-processing method is superior to baseline systems with multi-channel PSG inputs. Moreover, the proposed method produces stable results, and we reran the training and testing fusion algorithms for TinySleepNet and XSleepNet, which produced similar results. The proposed LSIM fusion framework can address a wide range of research problems with complicated dynamics and multi-channel structures because of its flexibility and reliability.

The proposed method is substantially faster than deep learning systems for two and three channels post-processing, resulting in good excess performance. The disadvantage of this method is that the learning duration increases with the square of the number of channels. Accordingly, if we were working with a 9-channel dataset in this study, integrated LSIM fusion training and testing could take approximately 60 hours (450x9 minutes), similar to deep learning. This method, however, becomes intractable for a dataset with 100 channels. Future studies may be possible to select the state number and GMM number with some proper criteria to train only one optimal LSIM instead of several to speed up the learning process. It allows the proposed method to be applied to systems with a large number of channels.

REFERENCES

[1] K. Šušmáková and A. Krakovská, "Discrimination ability of individual measures used in sleep stages classification," *Artificial intelligence in medicine*, vol. 44, no. 3, pp. 261–277, 2008.
[2] A. Rechtschaffen and A. Kales, "Standardized terminology, techniques and scoring system for sleep stages of human subjects," *A Manual of Public Health Service. US Government Printing Office, Washington*, 1968.
[3] C. Iber, S. Ancoli-Israel, A. L. Chesson, S. F. Quan, *et al.*, *The AASM manual for the scoring of sleep and associated events: rules, terminology and technical specifications*, vol. 1. American academy of sleep medicine Westchester, IL, 2007.
[4] R. G. Norman, I. Pal, C. Stewart, J. A. Walsleben, and D. M. Rapoport, "Interobserver agreement among sleep scorers from different centers in a large dataset.," *Sleep*, vol. 23, no. 7, pp. 901–908, 2000.
[5] F. Yaghouby, P. Modur, and S. Sunderam, "Naive scoring of human sleep based on a hidden markov model of the electroencephalogram," in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 5028–5031, IEEE, 2014.

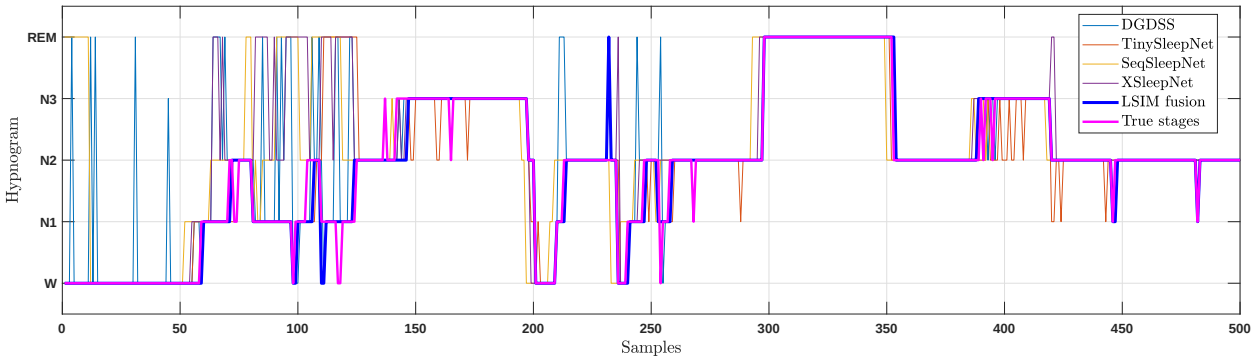

Fig. 9. The output hypnogram of the proposed integrated LSIM fusion of three channels processed by TinySleepNet and the output hypnograms of selected baseline systems for the Fpz-Cz channel in subject 2.

[6] H. Phan, F. Andreotti, N. Cooray, O. Y. Chén, and M. De Vos, "Seqsleepnet: end-to-end hierarchical recurrent neural network for sequence-to-sequence automatic sleep staging," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 3, pp. 400–410, 2019.

[7] A. Supratak, H. Dong, C. Wu, and Y. Guo, "Deepsleepnet: A model for automatic sleep stage scoring based on raw single-channel eeg," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 11, pp. 1998–2008, 2017.

[8] H. Ghimatgar, K. Kazemi, M. S. Helfroush, and A. Aarabi, "An automatic single-channel eeg-based sleep stage scoring method based on hidden markov model," *Journal of neuroscience methods*, vol. 324, p. 108320, 2019.

[9] G.-R. Liu, Y.-L. Lo, J. Malik, Y.-C. Sheu, and H.-T. Wu, "Diffuse to fuse eeg spectra–intrinsic geometry of sleep dynamics for classification," *Biomedical Signal Processing and Control*, vol. 55, p. 101576, 2020.

[10] D. Novák, Y. Al-Ani, and L. Lhotska, "Electroencephalogram processing using hidden markov models," in *Proceedings of the 5th EUROSIM Congress on Modelling and Simulation*, 2004.

[11] Y. Liu, L. Yan, B. Zeng, and W. Wang, "Automatic sleep stage scoring using hilbert-huang transform with bp neural network," in *2010 4th International Conference on Bioinformatics and Biomedical Engineering*, pp. 1–4, IEEE, 2010.

[12] C.-E. Kuo and S.-F. Liang, "Automatic stage scoring of single-channel sleep eeg based on multiscale permutation entropy," in *2011 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, pp. 448–451, IEEE, 2011.

[13] A. R. Hassan, S. K. Bashar, and M. I. H. Bhuiyan, "Automatic classification of sleep stages from single-channel electroencephalogram," in *2015 annual IEEE India conference (INDICON)*, pp. 1–6, IEEE, 2015.

[14] A. R. Hassan and M. I. H. Bhuiyan, "Dual tree complex wavelet transform for sleep state identification from single channel electroencephalogram," in *2015 IEEE International Conference on Telecommunications and Photonics (ICTP)*, pp. 1–5, IEEE, 2015.

[15] A. R. Hassan and M. I. H. Bhuiyan, "Automated identification of sleep states from eeg signals by means of ensemble empirical mode decomposition and random under sampling boosting," *Computer methods and programs in biomedicine*, vol. 140, pp. 201–210, 2017.

[16] A. R. Hassan and M. I. H. Bhuiyan, "Automatic sleep scoring using statistical features in the emd domain and ensemble methods," *Biocybernetics and Biomedical Engineering*, vol. 36, no. 1, pp. 248–255, 2016.

[17] H. Phan, O. Y. Chén, M. C. Tran, P. Koch, A. Mertins, and M. De Vos, "Xsleepnet: Multi-view sequential model for automatic sleep staging," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[18] H. Korkalainen, J. Aakko, S. Nikkonen, S. Kainulainen, A. Leino, B. Duce, I. O. Afara, S. Myllymaa, J. Töyräs, and T. Leppänen, "Accurate deep learning-based sleep staging in a clinical population with suspected obstructive sleep apnea," *IEEE journal of biomedical and health informatics*, vol. 24, no. 7, pp. 2073–2081, 2019.

[19] O. Tsinalis, P. M. Matthews, and Y. Guo, "Automatic sleep stage scoring using time-frequency analysis and stacked sparse autoencoders," *Annals of biomedical engineering*, vol. 44, no. 5, pp. 1587–1597, 2016.

[20] O. Tsinalis, P. M. Matthews, Y. Guo, and S. Zafeiriou, "Automatic sleep stage scoring with single-channel eeg using convolutional neural networks," *arXiv preprint arXiv:1610.01683*, 2016.

[21] 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," *Journal of neuroscience methods*, vol. 271, pp. 107–118, 2016.

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

[23] J. Kim, J.-S. Lee, P. Robinson, and D.-U. Jeong, "Markov analysis of sleep dynamics," *Physical review letters*, vol. 102, no. 17, p. 178104, 2009.

[24] L. Doroshenkov and V. Konyshev, "Usage of hidden markov models for automatic sleep stages classification," in *Russian-Bavarian Conference on Bio-Medical Engineering*, p. 19, Citeseer, 2007.

[25] K. Pillay, A. Dereymaeker, K. Jansen, G. Naulaers, S. Van Huffel, and M. De Vos, "Automated eeg sleep staging in the term-age baby using a generative modelling approach," *Journal of neural engineering*, vol. 15, no. 3, p. 036004, 2018.

[26] S.-T. Pan, C.-E. Kuo, J.-H. Zeng, and S.-F. Liang, "A transition-constrained discrete hidden markov model for automatic sleep staging," *Biomedical engineering online*, vol. 11, no. 1, p. 52, 2012.

[27] S. Aydın, "Computer based synchronization analysis on sleep eeg in insomnia," *Journal of medical systems*, vol. 35, no. 4, pp. 517–520, 2011.

[28] S. Aydın, M. A. Tunga, and S. Yetkin, "Mutual information analysis of sleep eeg in detecting psycho-physiological insomnia," *Journal of medical systems*, vol. 39, no. 5, pp. 1–10, 2015.

[29] M. Brand, "Coupled hidden markov models for modeling interacting processes," Tech. Rep. 405, MIT Media Lab Perceptual Computing/Learning and Common Sense, 1997.

[30] S. Karimi and M. B. Shamsollahi, "Tractable inference and observation likelihood evaluation in latent structure influence models," *IEEE Transactions on Signal Processing*, vol. 68, pp. 5736–5745, 2020.

[31] S. Karimi and M. B. Shamsollahi, "Tractable maximum likelihood estimation for latent structure influence models with applications to eeg & ecog processing," *TechRxiv*, 2022.

[32] S. Mallat, "Group invariant scattering," *Communications on Pure and Applied Mathematics*, vol. 65, no. 10, pp. 1331–1398, 2012.

[33] J. Andén and S. Mallat, "Deep scattering spectrum," *IEEE Transactions on Signal Processing*, vol. 62, no. 16, pp. 4114–4128, 2014.

[34] Ü. Doğan, T. Glasmachers, and C. Igel, "A unified view on multi-class support vector classification," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 1550–1831, 2016.

[35] A. Supratak and Y. Guo, "Tinysleepnet: An efficient deep learning model for sleep stage scoring based on raw single-channel eeg," pp. 641–644, 2020.

[36] A. L. Goldberger, L. A. 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, 2000.