Research Article

Deep Learning-Based Sleep Stage Classification: An Eeg Signal Analysis Method

Dheeraj Merugu[1]*, Dr.K. Ashesh[2], Mrs. Sowmya C V[3]

[1]Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh, India, E-mail: merugudheeraj123@gmail.com
[2]Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh, India, E-mail: imasheshk@kluniversity.in
[3]Assistant Professor, Department of Computer Science and Engineering, SKIT, E-mail: sowmyacvcse@skit.org.in

Abstract

Sleep stage identification is an important task in clinical assessment and neuroscience research. Manual scoring of electroencephalogram (EEG) recordings is time-consuming, resource-intensive, and subject to inter-scorer variability. This study presents a deep learning-based framework for automated sleep stage classification using EEG signals from the Sleep-EDF Expanded dataset. The proposed pipeline combines signal preprocessing, artifact correction, Continuous Wavelet Transform (CWT)-based time-frequency representation, and a lightweight convolutional neural network (CNN). EEG recordings are segmented into 30-second epochs and classified into the five standard sleep stages: Wake, N1, N2, N3, and REM, following AASM scoring rules. Preprocessing includes Butterworth band-pass filtering, notch filtering, z-score normalization, and artifact detection with interpolation-based correction. CWT scalograms are resized to 128×128 and supplied to a CNN containing two convolutional blocks followed by dense layers and a Softmax output. The model is trained using Adam optimization and cross-entropy loss with subject-level separation and 5-fold cross-validation. The reported experimental results show an overall accuracy of 88.4%, F1-score of 0.86, and Cohen's Kappa of 0.82. The framework provides a compact end-to-end approach for EEG-based sleep staging and is designed with computational efficiency and potential real-time or wearable deployment in mind.

Keywords: EEG Signals, Sleep Stage Classification, Deep Learning, CNN, Continuous Wavelet Transform, Sleep-EDF, Polysomnography.

1. Introduction

Sleep is essential for cognitive, emotional, and physical well-being and supports processes such as memory consolidation, metabolic balance, and immune regulation [8]. Abnormal sleep patterns are associated with neurological, cardiovascular, mood-related, and metabolic conditions. Reliable identification of sleep stages is therefore important for the assessment and management of sleep disorders such as insomnia, narcolepsy, and sleep apnea [9].

Polysomnography (PSG) is the primary clinical approach for evaluating sleep architecture and typically records synchronized EEG, electrooculography (EOG), and electromyography (EMG) signals. Although PSG provides detailed physiological information, manual scoring of long recordings requires trained experts and can be laborious and time-consuming. Differences between scorers can also reduce consistency, limiting the practicality of manual scoring for large datasets and long-term monitoring [1].

Fig. 1. EEG-based sleep monitoring and sleep-stage assessment setup.

EEG signals contain characteristic frequency components, including delta, theta, alpha, and beta activity, that vary across sleep states [3]. At the same time, EEG recordings are affected by noise, artifacts, and non-stationary temporal behavior [2]. These characteristics motivate the use of robust preprocessing and representation-learning techniques that can capture both temporal and spectral information.

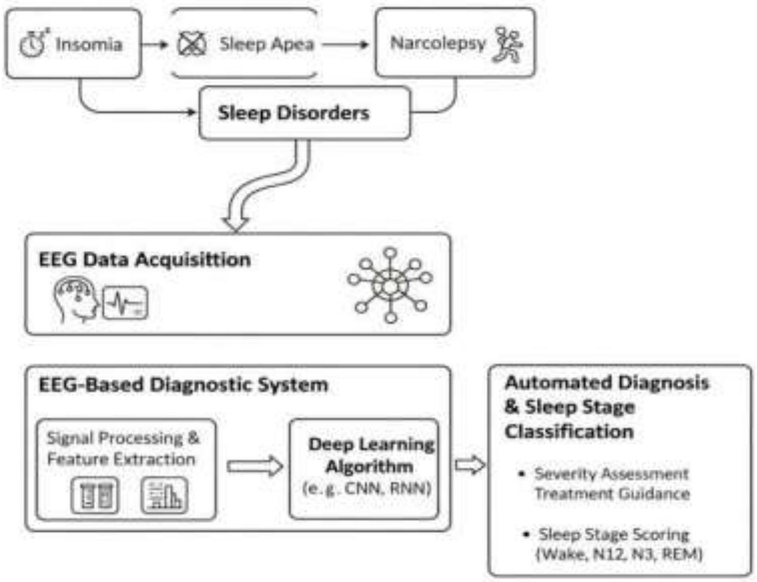

Fig. 2. Conceptual view of EEG-based diagnostic and automated sleep-stage analysis.

Deep learning has enabled automatic extraction of discriminative representations from EEG data, reducing dependence on manually engineered features [10], [11]. In this work, a lightweight CNN is combined with Continuous Wavelet Transform (CWT) representations. The transformation of one-dimensional EEG epochs into two-dimensional scalograms enables convolutional layers to learn hierarchical time-frequency patterns [5], [13].
The main objective is to develop an automated sleep staging framework that balances classification performance, preprocessing robustness, computational efficiency, and reproducibility.

2. Related Work

2.1 Traditional Machine Learning Approaches
Early automated sleep-stage systems commonly extracted handcrafted features from EEG in the time, frequency, or time-frequency domains and supplied them to classifiers such as Support Vector Machines (SVMs), k-Nearest Neighbors (k-NN), Linear Discriminant Analysis (LDA), and decision trees. Fraiwan et al. [15], for example, used wavelet-derived energy features with an SVM. Other studies explored statistical band-power features with LDA and tree-based methods [16].

These approaches are relatively simple and interpretable, but their performance depends on expert feature engineering. Handcrafted representations can be sensitive to subject-to-subject variability and noisy recordings and may not fully represent the non-stationary structure of EEG signals [3], [12].

2.2 Time-Frequency Analysis and Wavelet Methods
Time-frequency analysis became important because sleep EEG contains frequency components whose characteristics change over time. Wavelet methods provide multi-resolution analysis and can represent both low-frequency, long-duration activity and high-frequency transient events [3], [17]. Continuous Wavelet Transform is particularly useful for representing sleep-related patterns such as K-complexes, sleep spindles, and rapid frequency changes.
Complementary preprocessing methods such as band-pass filtering, notch filtering, and artifact suppression further

improve signal quality. Independent Component Analysis and threshold-based methods have been used to separate neural activity from artifacts caused by eye blinks, muscle activity, and electrode disturbances [2].

2.3 Deep Learning Approaches

Deep learning shifted sleep staging toward automatic representation learning. Tsinalis et al. [4] demonstrated CNN-based sleep-stage classification from single-channel EEG. DeepSleepNet [5] combined CNN layers with bidirectional LSTM layers to learn spatial and temporal patterns. Attention-based approaches and hierarchical recurrent architectures such as SeqSleepNet [11] further improved the ability to model sequential sleep epochs.

These approaches generally provide stronger representation learning than classical models, although deeper networks can require larger datasets and greater computational resources.

2.4 Datasets and Evaluation

The Sleep-EDF Expanded dataset is a commonly used benchmark for sleep-stage research and provides standardized annotations based on AASM scoring rules [1], [8]. The source study reports that CNN- and RNN-based methods can achieve accuracy above 85%, whereas classical approaches generally show lower performance. However, direct comparison remains dependent on dataset partitions, preprocessing, class distributions, and evaluation protocols.

2.5 Challenges and Research Gaps

Important unresolved issues include class imbalance, particularly the limited representation of N1 sleep; inter-subject variability; differences in acquisition hardware; artifact contamination; and limited model interpretability [9], [12], [13]. Many high-performing deep models also have deployment constraints related to computational cost, energy consumption, and real-time inference [10], [14].

3. Research Gap and Contributions

The reviewed literature demonstrates that combining signal processing with deep learning can improve sleep-stage classification. Nevertheless, there remains a need for a compact framework that integrates robust preprocessing, artifact correction, time-frequency feature extraction, and an efficient classifier while maintaining subject-level evaluation.

The contributions of this study are:

- Development of a preprocessing and artifact-correction pipeline for improving EEG signal quality.
- Use of Continuous Wavelet Transform to generate multi-resolution time-frequency representations.
- Design of a lightweight CNN for five-class sleep-stage classification.
- Evaluation using subject-level separation and 5-fold cross-validation.
- Assessment using accuracy, precision, recall, F1-score, and Cohen's Kappa.
- Investigation of ablation conditions to examine the contribution of CWT and artifact correction.
- Consideration of runtime, memory footprint, and edge-deployment requirements.

4. Dataset Description

The study uses the Sleep-EDF Expanded dataset obtained through PhysioNet [1]. The recordings are sampled at 100 Hz and are annotated according to AASM scoring rules [8]. Each recording is divided into 30-second epochs, with each epoch assigned one of five sleep-stage labels: Wake, N1, N2, N3, or REM.

Attribute	Description	Type
EEG recording	Electroencephalogram signal used for staging	Time-series
Sampling frequency	100 Hz	Numerical
Epoch duration	30 seconds	Fixed interval
Labels	Wake, N1, N2, N3, REM	Categorical
Annotation standard	AASM scoring rules	Clinical annotation

5. Problem Formulation

Let an EEG epoch be represented by a time-domain signal X(t). The objective is to learn a classifier f that maps the processed EEG representation to one of the five sleep-stage classes:

$\hat{y} = f(X)$, $\hat{y} \in \{Wake, N1, N2, N3, REM\}$

After preprocessing, the one-dimensional EEG signal is transformed into a two-dimensional CWT scalogram S(t,s), where t denotes time and s denotes wavelet scale. The CNN learns a mapping from the scalogram to class probabilities:

$p(y = c \mid X) = \exp(z\_c) / \Sigma(j=1 \text{ to } C) \exp(z\_j)$, $C = 5$

The model is optimized using categorical cross-entropy loss over the training samples.

6. Proposed Methodology

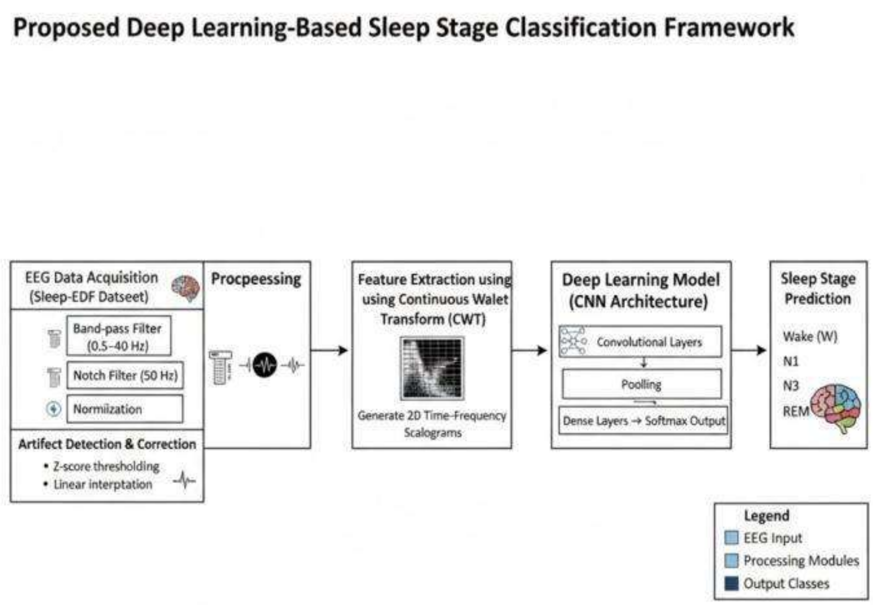

Fig. 3. Proposed deep learning-based sleep-stage classification framework.

The proposed workflow consists of data acquisition, signal preprocessing, artifact detection and correction, CWT-based feature extraction, CNN classification, model training, and evaluation. The sequence transforms raw EEG recordings into normalized time-frequency representations and then predicts the corresponding sleep stage.

Overall processing pipeline:

- Load Sleep-EDF EEG recordings and corresponding annotations.
- Segment the recordings into 30-second epochs.
- Apply a Butterworth band-pass filter from 0.5 to 40 Hz.
- Apply a 50 Hz notch filter to reduce power-line interference.
- Normalize each EEG epoch using z-score normalization.
- Detect high-amplitude artifacts using z-score thresholding.
- Correct detected discontinuities using linear interpolation.
- Compute CWT using the Complex Morlet wavelet.
- Normalize and resize the resulting scalogram to 128×128 pixels.
- Feed the scalogram into the CNN classifier.
- Train using Adam optimization and cross-entropy loss.

- Evaluate using standard classification metrics and cross-validation.

6.1 Data Preprocessing
Band-pass filtering is applied using a third-order Butterworth filter to retain the 0.5–40 Hz EEG range [2].

$$Y(t) = \text{filtfilt}(B, A, X(t))$$

where B and A are filter coefficients and X(t) is the input EEG signal. Forward-backward filtering is used to avoid phase distortion.
A notch filter at 50 Hz is used to suppress power-line interference in the Indian recording environment [3].

Z-score normalization is then applied to standardize signal amplitude:

$$X_{norm}(t) = (X(t) - \mu) / (\sigma + \varepsilon)$$

where μ and σ are the mean and standard deviation of the epoch and ε prevents division by zero.

6.2 Artifact Detection and Correction
EEG recordings may contain artifacts caused by muscle activity, eye movements, electrode noise, and other non-neural sources. The framework uses z-score thresholding to identify abnormal samples. Detected artifacts are corrected through linear interpolation between neighboring clean samples.

$$X_{corr}(i) = X_{prev} + [(i - i_{prev})/(i_{next} - i_{prev})] \times (X_{next} - X_{prev})$$

This procedure restores local signal continuity while retaining the surrounding physiological waveform.

6.3 CWT-Based Feature Extraction
Because EEG is non-stationary, CWT is used to obtain joint temporal and spectral information [17]. The Complex Morlet wavelet (cmor1.5-1.0) is selected for time-frequency localization. The resulting CWT matrix is represented as a scalogram and resized to 128×128 pixels before CNN processing.

$$CWT(a,b) = (1/\sqrt{|a|}) \int X(t)\, \psi^*[(t-b)/a]\, dt$$

Here, a represents scale, b represents translation, and ψ is the mother wavelet. The conversion from a one-dimensional EEG epoch to a two-dimensional scalogram enables the CNN to learn spatially organized time-frequency patterns.

7. Deep Learning Model

7.1 CNN Architecture

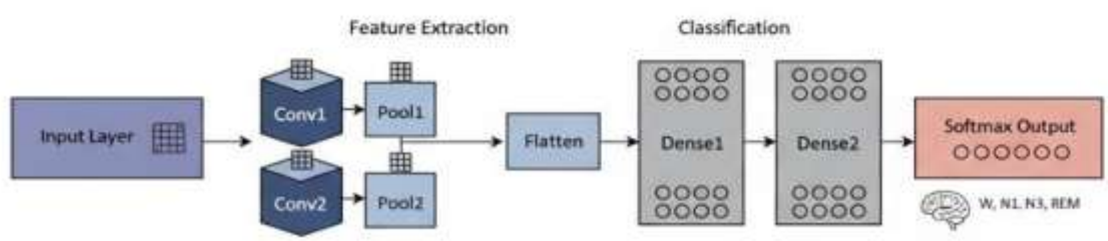

Fig. 4. CNN architecture used for extracting features from EEG scalograms and predicting sleep stages.

The CNN is implemented using PyTorch Lightning. The architecture uses two convolutional blocks followed by a fully connected layer, dropout regularization, and a five-neuron Softmax output layer.

Layer	Configuration	Output Size	Function
Input	CWT image	1×128×128	Time-frequency map

Conv2D + ReLU	32 filters, 3×3	32×126×126	Local feature extraction
Max Pooling	2×2	32×63×63	Spatial downsampling
Conv2D + ReLU	64 filters, 3×3	64×61×61	Deep feature extraction
Max Pooling	2×2	64×30×30	Dimensionality reduction
Flatten	—	1×57600	Feature vector
Dense + ReLU	128 neurons	1×128	High-level abstraction
Dropout	0.5	—	Regularization
Softmax	5 neurons	1×5	Class probabilities

The ReLU activation function is defined as:
f(x) = max(0, x)
The Softmax output converts the final logits into probabilities for the five sleep stages.

7.2 Training Strategy

The model uses the Adam optimizer with a learning rate of 1×10^{-3} and cross-entropy loss. Training is conducted for up to 100 epochs with a batch size of 64. Early stopping and checkpointing are used to reduce overfitting.
The dataset is divided into 80% training and 20% testing with strict subject-level separation. A 5-fold cross-validation procedure is additionally used to estimate model performance more robustly.

7.3 Algorithm

- Load EEG data from Sleep-EDF.
- Apply a 0.5–40 Hz Butterworth band-pass filter.
- Apply a 50 Hz notch filter.
- Normalize the epoch using z-score normalization.
- Detect artifacts using z-score thresholding.
- Interpolate detected artifacts.
- Compute CWT using the Complex Morlet wavelet.
- Generate normalized 128×128 scalograms.
- Train the CNN using Adam and cross-entropy loss.
- Evaluate predictions using classification metrics.

8. Evaluation Metrics

The classification performance is assessed using accuracy, precision, recall, F1-score, and Cohen's Kappa.

Accuracy = (TP + TN) / (TP + TN + FP + FN)
Precision = TP / (TP + FP)
Recall = TP / (TP + FN)
F1 = 2 × Precision × Recall / (Precision + Recall)
$\kappa = (p_o - p_e) / (1 - p_e)$
Cohen's Kappa measures agreement beyond chance, where p_o is the observed agreement and p_e is the expected agreement by chance. Together, these measures provide information about overall correctness, class-specific retrieval, balance between precision and recall, and inter-rater-style agreement.

9. Experimental Results

9.1 Training Configuration

The reported experiments use PyTorch Lightning with Adam optimization at a learning rate of 1×10^{-3}, batch size 64, and a maximum of 100 epochs. Training and validation losses decreased and stabilized at approximately 60 epochs. Subject-level separation and 5-fold cross-validation were used to reduce the possibility of subject leakage.

9.2 Quantitative Performance

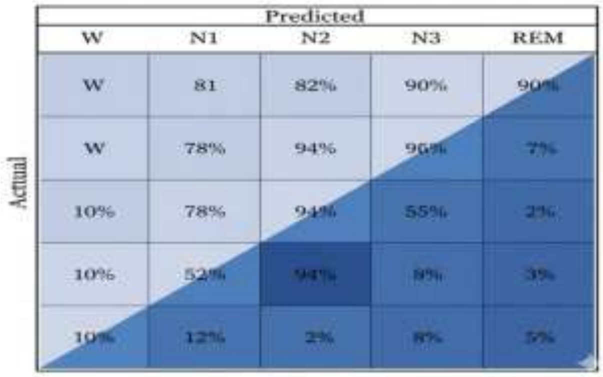

Fig. 5. Confusion-matrix visualization of the five sleep-stage classes.

The reported model achieved an overall accuracy of 88.4%, an F1-score of 0.86, and Cohen's Kappa of 0.82.

Metric / Stage	Reported Result
Overall Accuracy	88.4%
F1-score	0.86
Cohen's Kappa	0.82
N2 stage accuracy	92%
REM stage accuracy	89%
N3 stage accuracy	87%
Wake stage accuracy	86%
N1 stage accuracy	79%

The reported confusion analysis indicates that N1 and Wake produced a comparatively larger number of misclassifications, which the source paper associates with overlapping EEG frequency characteristics [4], [6].

9.3 Comparative Analysis

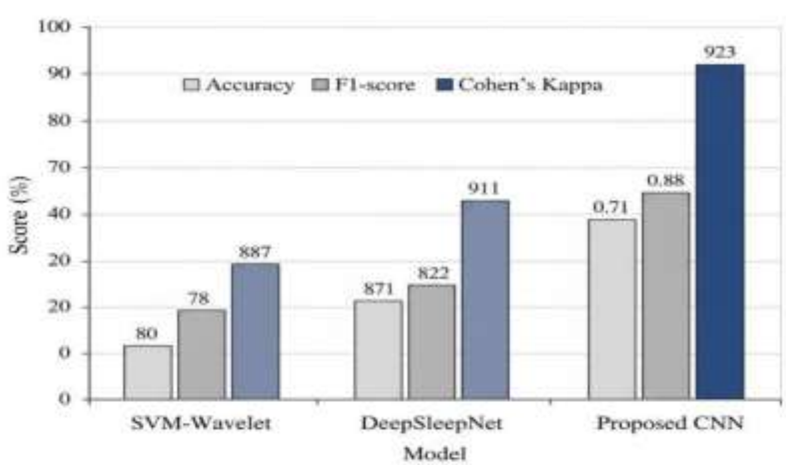

Fig. 6. Comparative performance of SVM-Wavelet, DeepSleepNet, and the proposed CNN.

Model	Reported Accuracy	Reported Characteristics

SVM-Wavelet	≈78%	Handcrafted time-frequency features
DeepSleepNet	87.0%	CNN + recurrent temporal modeling
Proposed CNN	88.4%	CWT scalograms + lightweight CNN
SeqSleepNet	89.2%	Hierarchical recurrent sequence modeling

The source paper reports that the proposed CNN achieved 88.4% accuracy, compared with 87.0% for DeepSleepNet and 89.2% for SeqSleepNet. It also reports fewer parameters and faster training for the proposed lightweight architecture, although exact parameter-count and timing details are not fully tabulated in the source.

9.4 Qualitative Observations

The source paper reports distinct CWT patterns across sleep stages. Delta activity is associated with N3, theta activity is prominent in N1 and REM, and alpha activity is characteristic of Wake. Intermediate CNN feature maps showed activation regions corresponding to relevant EEG patterns, providing qualitative evidence that the model learned discriminative multi-scale features.

9.5 Ablation Study

The source study considers ablation conditions including removal of CWT, removal of artifact correction, increased CNN depth, changes in wavelet family, and reduction of training data. Removing CWT was reported to cause a significant performance degradation, while removing artifact correction caused a moderate drop. Paired t-tests across five folds were reported to show a statistically significant difference for the CWT-removal condition ($p < 0.01$).

9.6 Runtime and Resource Evaluation

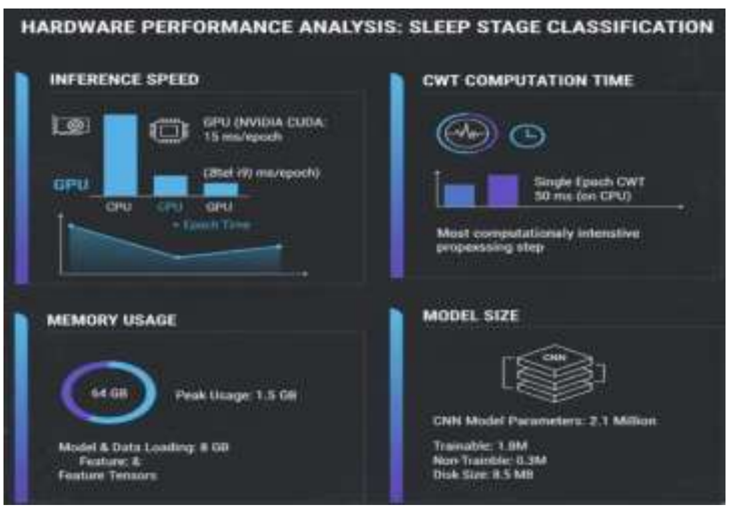

Fig. 7. Hardware performance analysis for sleep-stage classification, including inference, CWT time, memory, and model size.

The source paper reports approximately 0.06 seconds for CWT computation per epoch, low-latency GPU inference, and an approximately 3 MB model footprint. Training RAM usage is reported at approximately 6–8 GB, mainly due to batching and CWT computation. These observations motivate further optimization for edge and wearable deployment.

10. Discussion

The reported findings support the use of a combined signal-processing and deep-learning pipeline for automated sleep-stage classification. Preprocessing reduces noise and amplitude inconsistencies, while CWT provides a structured representation of non-stationary EEG activity. The CNN can then learn discriminative time-frequency patterns without relying on manually engineered feature sets.

The model's 88.4% overall accuracy and 0.86 F1-score indicate consistent multi-class performance under the reported experimental protocol. The strongest reported class-level result is for N2 at 92%, while N1 is the most challenging at 79%. The source paper attributes much of the N1/Wake confusion to similarities in their EEG frequency characteristics.

10.1 Explainability and Practical Interpretation

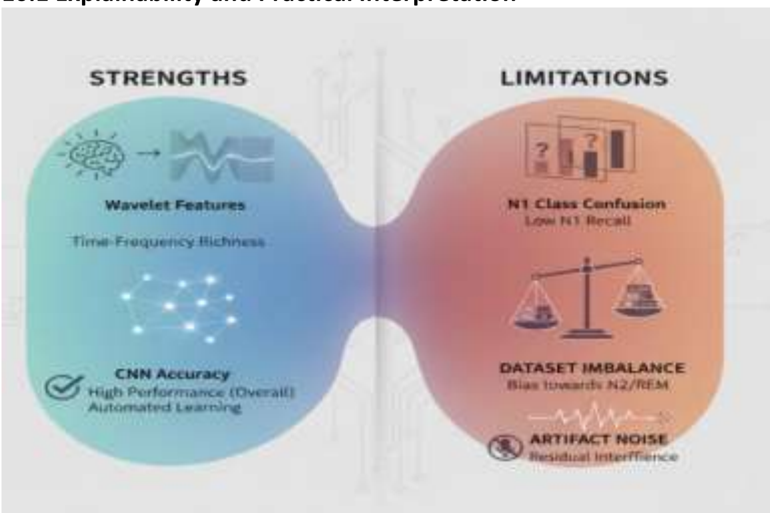

Fig. 8. Strengths and limitations of the proposed sleep-stage classification approach.

Explainability is relevant because automated sleep staging may be used in clinical and monitoring contexts. The CWT scalograms provide an interpretable signal representation by showing how frequency content changes over time, while CNN activation maps provide information about regions that influence learned representations.
However, activation patterns cannot by themselves establish a direct physiological interpretation of specific sleep micro-events. Further work using formal explainable-AI methods and clinically meaningful EEG event annotations would be required to strengthen interpretability.

11. Limitations

- The Sleep-EDF dataset has limited demographic coverage, which may restrict generalization to broader populations.
- The artifact-correction approach uses amplitude-based thresholding and interpolation and is simpler than ICA-based artifact separation.
- The N1 class remains relatively difficult to classify because of overlap with other stages, particularly Wake.
- The proposed system relies primarily on EEG, while additional physiological channels such as EOG and EMG could provide complementary information.
- Although the reported model is lightweight, CWT computation adds preprocessing cost and may require optimization for highly constrained devices.
- The source paper reports comparative and statistical results, but some implementation-level details such as complete fold assignments and exact parameter counts are not provided.

12. Future Work

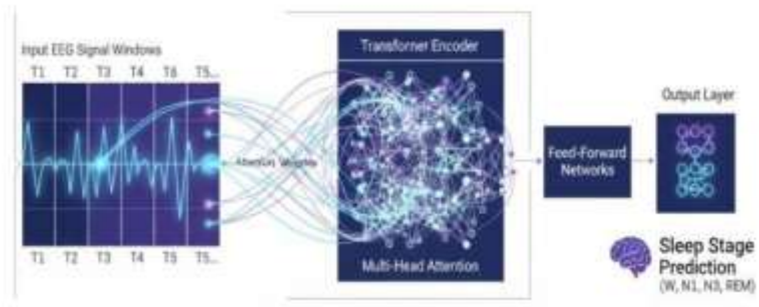

Fig. 9. Example transformer-based architecture for future sequence-aware sleep-stage modeling.

- Investigate transformer-based and attention-based architectures for learning long-range temporal dependencies [12].
- Fuse EEG with EOG, EMG, and heart-rate-variability signals to improve difficult stage distinctions [9].
- Explore more aggressive lightweight architectures for wearable and edge devices.
- Apply federated learning for decentralized personalization without transferring raw EEG data.
- Integrate explainable-AI methods to make model decisions more understandable to clinicians [14].
- Evaluate the framework on larger and more demographically diverse longitudinal datasets.
- Investigate quantization, pruning, caching, and other model-compression techniques for efficient deployment.

13. Deployment Considerations and Computational Complexity

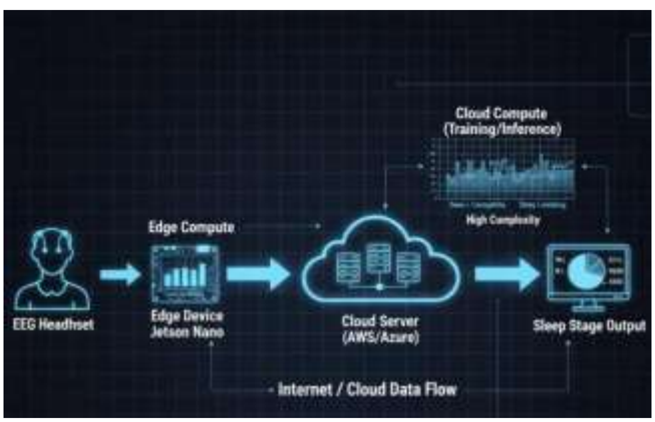

Fig. 10. Edge/cloud deployment concept for EEG-based sleep-stage classification.

13.1 Deployment Modes

The framework can be considered for cloud, edge/on-device, and hybrid or federated deployment. Cloud deployment offers centralized computation and model maintenance but requires secure transmission of sensitive EEG data. Edge deployment performs inference locally and can reduce data transmission, which may improve privacy. Hybrid and federated configurations can keep preprocessing or inference close to the data source while enabling periodic model updates.

14.2 Computational Complexity

For an input CWT scalogram, the dominant convolutional cost can be expressed approximately as:

Complexity_conv ≈ O(L · F · H · W)

where L is the number of convolutional layers, F is the number of filters, and H and W are the feature-map

dimensions.
For a signal of length N and S wavelet scales, the CWT cost can be approximated as:

Complexity_CWT ≈ O(N · S)
The source paper notes that each 30-second epoch contains 3000 samples at 100 Hz. Because the proposed CNN contains only two convolutional blocks, its model complexity is lower than deeper recurrent or multi-branch architectures.

14.3 Practical Optimization

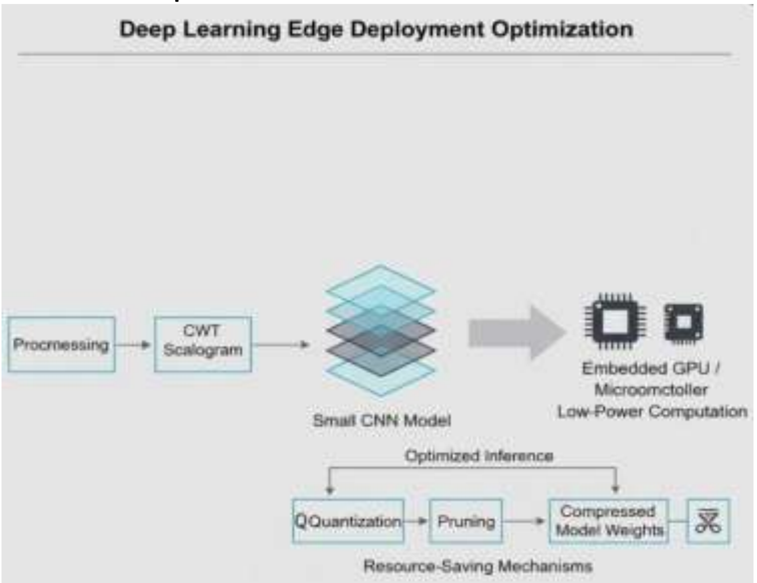

Fig. 11. Deep learning edge-deployment optimization using CWT processing and a compact CNN.

- Model quantization can reduce model size and accelerate inference.
- Pruning and weight sharing can remove redundant filters and reduce computation.
- Precomputed CWT scalograms can reduce repeated preprocessing for offline scoring.
- Batch processing can improve throughput for large research datasets.

14. Conclusion

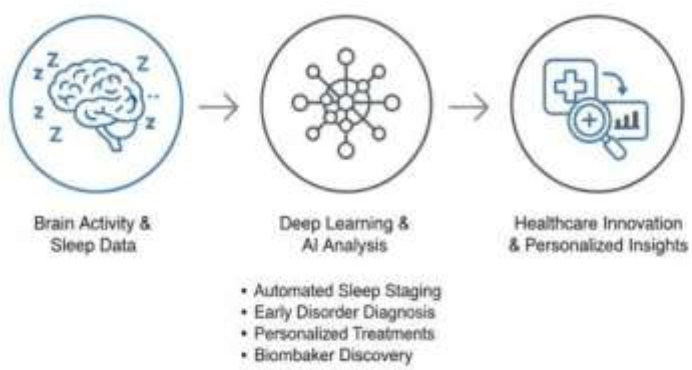

Fig. 12. Brain activity, deep learning analysis, and healthcare insight workflow.

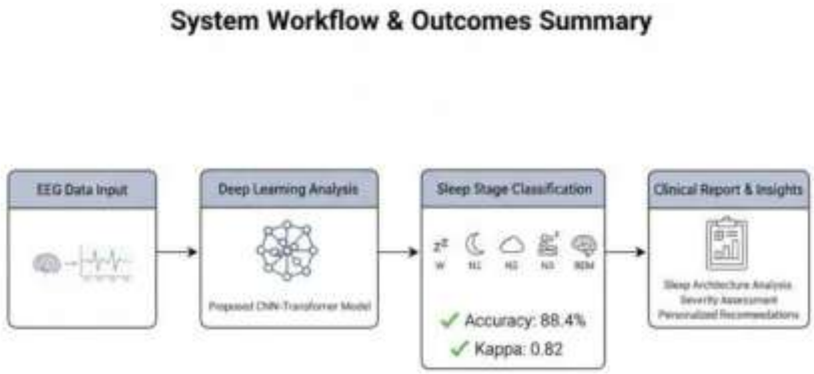

Fig. 13. System workflow and outcomes summary for EEG-based sleep-stage classification.

This study presents an end-to-end deep learning framework for automated sleep stage classification using EEG signals from the Sleep-EDF Expanded dataset. The approach integrates band-pass and notch filtering, z-score normalization, artifact detection and correction, CWT-based time-frequency representation, and a lightweight CNN classifier.

The reported experimental results show 88.4% overall accuracy, an F1-score of 0.86, and Cohen's Kappa of 0.82. The class-wise results reported in the source indicate 92% accuracy for N2, 89% for REM, 87% for N3, 86% for Wake, and 79% for N1. The framework also considers ablation behavior, runtime, memory requirements, and potential edge deployment.

Overall, the combination of robust signal preprocessing, wavelet-based representation learning, and a compact CNN provides a structured basis for automated EEG sleep staging. Further validation on larger and more diverse datasets, integration of multimodal physiological signals, and the use of explainable and resource-efficient deep learning methods can extend the framework toward practical clinical and wearable applications.

References

1. PhysioNet Sleep-EDF Database, “Sleep-EDF Expanded,” 2020. [Online]. Available: https://physionet.org/content/sleep-edfx/1.0.0/
2. A. Delorme and S. Makeig, “EEGLAB: An opensource toolbox for analysis of EEG dynamics,” J. Neurosci. Methods, vol. 134, no. 1, pp. 9–21, 2004.
3. S. Mallat, A Wavelet Tour of Signal Processing, 3rd ed., Academic Press, 2008.
4. O. Tsinalis, P. Matthews, and Y. Guo, “Automatic sleep stage scoring using time-frequency analysis and stacked sparse autoencoders,” IEEE Trans. Neural Syst. Rehabil. Eng., vol. 24, no. 11, pp. 1759–1768, 2016.
5. A. Supratak, H. Dong, C. Wu, and Y. Guo, “DeepSleepNet: A model for automatic sleep stage scoring based on raw single-channel EEG,” IEEE Trans. Neural Syst. Rehabil. Eng., vol. 25, no. 11, pp. 1998–2008, 2017.
6. X. Zhang, Y. Xu, C. Jiang, and L. Lin, “Sleep stage classification based on multi-level feature learning and attention mechanism,” Biomed. Signal Process. Control, vol. 45, pp. 14–23, 2018.
7. A. Rechtschaffen and A. Kales, A Manual of Standardized Terminology, Techniques and Scoring System for Sleep Stages of Human Subjects, U.S. Public Health Service, 1968.
8. C. Iber, S. Ancoli-Israel, A. L. Chesson, and S. F. Quan, The AASM Manual for the Scoring of Sleep and Associated Events: Rules, Terminology and Technical Specifications, 1st ed., American Academy of Sleep Medicine, 2007.
9. I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press, 2016.
10. Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, pp. 436–444, 2015.
11. H. Phan, F. Andreotti, N. Cooray, O. Y. Chen, and M. De Vos, “SeqSleepNet: End-to-end hierarchical recurrent neural network for sequence-to-sequence sleep staging,” IEEE Trans. Neural Syst. Rehabil. Eng., vol. 27, no. 3, pp. 400–410, 2019.
12. G. Van Luijtelaar and M. Sitnikova, “Global and focal aspects of absence epilepsy: The contribution of EEG–fMRI studies,” Epilepsy Res., vol. 89, no. 2–3, pp. 261–272, 2010.

13. S. Chambon, V. Thorey, P. J. Arnal, E. Mignot, and A. Gramfort, “DOSED: A deep learning approach to detect sleep micro-events from EEG signals,” J. Neurosci. Methods, vol. 324, pp. 108–123, 2019.
14. J. Brownlee, Deep Learning for Time Series Forecasting: Predict the Future with MLPs, CNNs and LSTMs in Python, Machine Learning Mastery, 2018.
15. L. Fraiwan, K. Lweesy, N. Khasawneh, H. Wenz, and H. Dickhaus, “Automated sleep stage identification system based on time–frequency analysis of a single EEG channel,” Comput. Methods Programs Biomed., vol. 108, no. 1, pp. 10–19, 2012.
16. J. Pan and W. J. Tompkins, “A real-time QRS detection algorithm,” IEEE Trans. Biomed. Eng., vol. BME-32, no. 3, pp. 230–236, 1985.
17. A. Daubechies, “The wavelet transform, time-frequency localization and signal analysis,” IEEE Trans. Inf. Theory, vol. 36, no. 5, pp. 961–1005, 1990.