Article:
Huang, J., Li, D., Yang, P. et al. (2026) Riemannian spatio-temporal graph neural network for enhanced cognitive load detection using EEG. Neurocomputing, 685. 133650. ISSN: 0925-2312

https://doi.org/10.1016/j.neucom.2026.133650

Riemannian Spatio-Temporal Graph Neural Network for Enhanced Cognitive Load Detection Using EEG

Jiayang Huang[a,b], Dingnan Li[a,b], Pengfei Yang[a,b,*], Jingyi Shi[a,b], Wenjuan Zhang[c], Quan Wang[a,b] and Zhi-Qiang Zhang[d]

[a] School of Computer Science and Technology, Xidian University, Xi'an, 710126, China
[b] The Key Laboratory of Smart Human-Computer Interaction and Wearable Technology of Shaanxi Province, Xi'an, 710126, China
[c] Department of Psychology, the School of Humanities, Xidian University, Xi'an, 710071, China
[d] School of Electronic and Electrical Engineering, Institute of Robotics, Autonomous Systems and Sensing, University of Leeds, Leeds, U.K

ARTICLE INFO

Keywords:
Cognitive load detection
electroencephalography (EEG)
Graph neural networks (GNNs)
Riemannian manifold
temporal attention mechanism

ABSTRACT

Cognitive load detection using EEG signals is crucial for real-time performance monitoring in high-stakes environments, such as aviation, healthcare, and education. Existing methods, however, often fail to effectively capture the complex spatial and temporal dependencies inherent in EEG data. Recent approaches have leveraged graph neural networks (GNNs) for spatial modeling, but many overlook the rich covariance structure of EEG signals and neglect non-local temporal dependencies. To address these challenges, a novel framework, the Riemannian Spatio-Temporal Graph Neural Network (RST-GNN), is proposed for multi-class cognitive load detection based on electroencephalography (EEG). This method integrates Riemannian manifold filtering with temporal graph learning to model both spatial and temporal dynamics jointly. Specifically, EEG signals are filtered using Riemannian spatial filtering to extract discriminative covariance features, which are then represented as nodes in a temporal graph. A Graph-BiMap module is used to extract structured features from these nodes. A temporal attention mechanism is then applied to fuse window-level features, thereby enhancing the model's ability to capture evolving cognitive states. Experiments on two publicly available EEG datasets demonstrate that the proposed method consistently outperforms existing models across four cognitive load levels, achieving classification accuracies exceeding 96% with low inter-subject variability. These results highlight the robustness of the proposed method and its effectiveness for reliable cognitive load detection.

1. Introduction

Cognitive load refers to the total amount of load on an individual's working memory when performing cognitive tasks (Evans et al., 2024). Excessive cognitive load can hinder learning, increase error rates, and pose significant safety risks, particularly in high-stakes environments such as aviation (Lagomarsino et al., 2022). Consequently, accurate detection of cognitive load is vital for supporting effective task performance and preserving mental function under complex operational conditions (Amadori et al., 2021).

Cognitive load assessment methods can be broadly categorized into subjective and objective approaches. Subjective tools such as the NASA Task Load Index (NASA-TLX) (Hart and Staveland, 1988) and the Subjective Workload Assessment Technique (SWAT) (Reid and Nygren, 1988) rely on self-reporting, but suffer from limited reliability and lack real-time responsiveness. In contrast, objective methods leverage task performance metrics or physiological signals, including electroencephalography (EEG), heart rate, eye movement, respiration, electromyography (EMG), and others (Tao et al., 2019). Among these, EEG has attracted increasing attention due to its non-invasiveness, high temporal resolution, and portability, making it well-suited for cognitive load detection in practical scenarios such as monitoring driver attention (Zhou et al., 2023; Ma et al., 2024), evaluating student workload (Skulmowski and Xu, 2022), and supporting cognitive-aware human-machine interactions (Kosch et al., 2023).

Recent advancements have increasingly explored the use of graph neural networks (GNNs) for modeling the complex spatial dependencies in EEG signals. GNNs are particularly well-suited for this task as they can capture the relationships between different electrodes, which are spatially distributed across the scalp (Klepl et al., 2022; Li et al., 2021; Klepl et al., 2023). For example, Demir et al. (Demir et al., 2021) proposed a graph neural network (GNN) model that represents EEG electrodes as nodes with various connectivity policies for edges and processes them through GNN layers. Chen et al.(Chen et al., 2022) proposed a self-attention graph pooling network (SGP-SL) that jointly models local and global EEG topological structures through graph neural networks, while incorporating soft labels to improve detection performance by capturing fine-grained intra-class variations. Pan et al.(Pan et al., 2023) proposed a spatio-temporal self-constructing graph neural network that first extracts activation and connection pattern features via multi-scale convolution, then dynamically constructs adjacency matrices based on input features to perform graph convolution. Shen et al. (Shen et al., 2025) proposed a dynamic sparse directed graph convolutional network (DSDirGCN-AM). The method dynamically constructs directed adjacency matrices

*Corresponding author.
huangjiayang@xidian.edu.cn (J. Huang); 24031212346@stu.xidian.edu.cn (D. Li); pfyang@xidian.edu.cn (P. Yang); 25031212128@stu.xidian.edu.cn (J. Shi); wjuanzhang@xidian.edu.cn (W. Zhang); qwang@xidian.edu.cn (Q. Wang); z.zhang3@leeds.ac.uk (Z. Zhang)

to model directional information flow between brain regions and incorporates attention mechanisms to enhance key features. Li et al.(Li et al., 2025) proposed a spatiotemporal separable graph convolutional network (STSGCN) that first extracts temporal features from raw EEG using a gated temporal convolution block, then applies Chebyshev graph convolution to model spatial relationships, and finally fuses these features using separable convolution for event-related potential (ERP) classification under varying cognitive-load conditions.

However, these models typically construct graphs where nodes represent EEG channels, and edges are defined based on physical proximity or statistical similarity. This approach, while effective, overlooks the rich spatial dependencies captured in the covariance structure of EEG signals (Yin et al., 2024; Yan et al., 2023), which reside on a Riemannian manifold. Neglecting the geometric properties of these covariance matrices can limit the expressive power of the spatial representations learned by GNNs(Nguyen and Artemiadis, 2018; Kalaganis et al., 2022), (Lotte et al., 2018). Beyond spatial modeling, capturing the temporal dynamics of EEG signals remains challenging. Many existing approaches, including recurrent neural networks, temporal convolutional networks, and Transformer-based architectures, are commonly formulated under a sequential dependency assumption, where temporal relationships are primarily modeled through adjacent or locally structured interactions. However, cognitive processes often involve re-entrant neural activity and distributed interactions, where similar mental states may recur across non-adjacent time windows(Cowan, 2001; Dehaene et al., 2014). As a result, constraining temporal modeling to a strictly sequential and locally structured formulation may limit the explicit representation of non-local temporal dependencies that are critical for accurate cognitive state decoding.

To address these limitations, we propose a novel framework, Riemannian Spatio-Temporal Graph Neural Network (RST-GNN), which integrates Riemannian manifold filtering with temporal graph learning. This approach enables the joint modeling of spatial-temporal dynamics in EEG signals, capturing both the rich covariance structure and the complex temporal dependencies inherent in cognitive processes. Our method first applies Riemannian spatial filtering to extract discriminative covariance features. The filtered EEG signals are then segmented into temporal windows, and each window is represented as a covariance matrix, forming the nodes of a temporal graph. Graph-BiMap layers are used instead of graph convolutional layers to extract structured features from these nodes. Furthermore, a temporal attention mechanism is introduced to adaptively fuse information across different time windows, enhancing the model's ability to capture evolving cognitive states. The main contributions of this study are summarized as follows:

- Proposing a RST-GNN framework for EEG-based cognitive load detection, which integrates temporal graph-based representations, GNNs, Riemannian manifold filtering, and temporal attention mechanisms to model both the spatial and temporal dynamics of EEGs effectively.
- Designing a graph learning pipeline that leverages GNNs to capture the spatial dependencies across EEG electrodes, thereby enabling the extraction of structured and discriminative EEG representations.
- Introducing a Riemannian manifold-based spatial filter that enhances the robustness of EEG feature extraction by better modeling the intrinsic non-Euclidean structure of the EEG data.
- Incorporating a temporal attention mechanism to improve the fusion of window-level features, thereby enabling the model to focus on important time windows and capture complex temporal dependencies.
- Validating the effectiveness of the proposed method through extensive experiments on multiple benchmark EEG datasets, demonstrating its superiority in multi-class cognitive load detection.

The rest of the article is arranged as follows: The methodology is introduced in Section 2. In section 3, the experimental settings are presented. Section 4 presents the experimental results and discussions. Finally, the conclusion is presented in Section 5.

2. Methodology

2.1. Main Framework

In this paper, we propose the Riemannian Spatio-Temporal Graph Neural Network (RST-GNN) for cognitive load detection, aiming to effectively capture the spatio-temporal dynamics of EEG signals within the framework of Riemannian geometry. The model consists of three main modules, and the overall framework is illustrated in Fig. 1.

First, a Multiclass Riemannian Geometry-based Spatial Filtering (MCRSF) module is applied to enhance the spatial discriminability of raw EEG signals across multiple cognitive load levels. Given an input EEG sample $\boldsymbol{X}_i \in \mathbb{R}^{N_{c_1} \times N_{d_1}}$, the MCRSF module learns an optimal projection matrix $\boldsymbol{W}^* \in \mathbb{R}^{N_{c_1} \times N_{c_2}}$, and computes the filtered signal as $\boldsymbol{\mathcal{X}}_i = (\boldsymbol{W}^*)^T \boldsymbol{X}_i \in \mathbb{R}^{N_{c_2} \times N_{d_1}}$. This transformation projects the original signals into a more discriminative spatial subspace for subsequent analysis.

Second, a Riemannian GNN module is applied to extract structured representations from the temporally segmented and graph-modeled EEG data. The filtered signal $\boldsymbol{\mathcal{X}}_i$ is divided into N_w overlapping time windows, resulting in $\boldsymbol{\mathcal{X}}'_i \in \mathbb{R}^{N_w \times N_{c_2} \times N_{d_2}}$. Each window is mapped to a symmetric positive definite (SPD) covariance matrix $\boldsymbol{P}_{\text{win}} \in \mathbb{R}^{N_{c_2} \times N_{c_2}}$, which serves as a node in the graph. A Riemannian distance-based K-nearest neighbor strategy is used to construct the adjacency matrix $\boldsymbol{A} \in \mathbb{R}^{N_w \times N_w}$. Graph-BiMap layers are then employed to extract high-level representations from the manifold-valued graph nodes. These outputs are mapped to

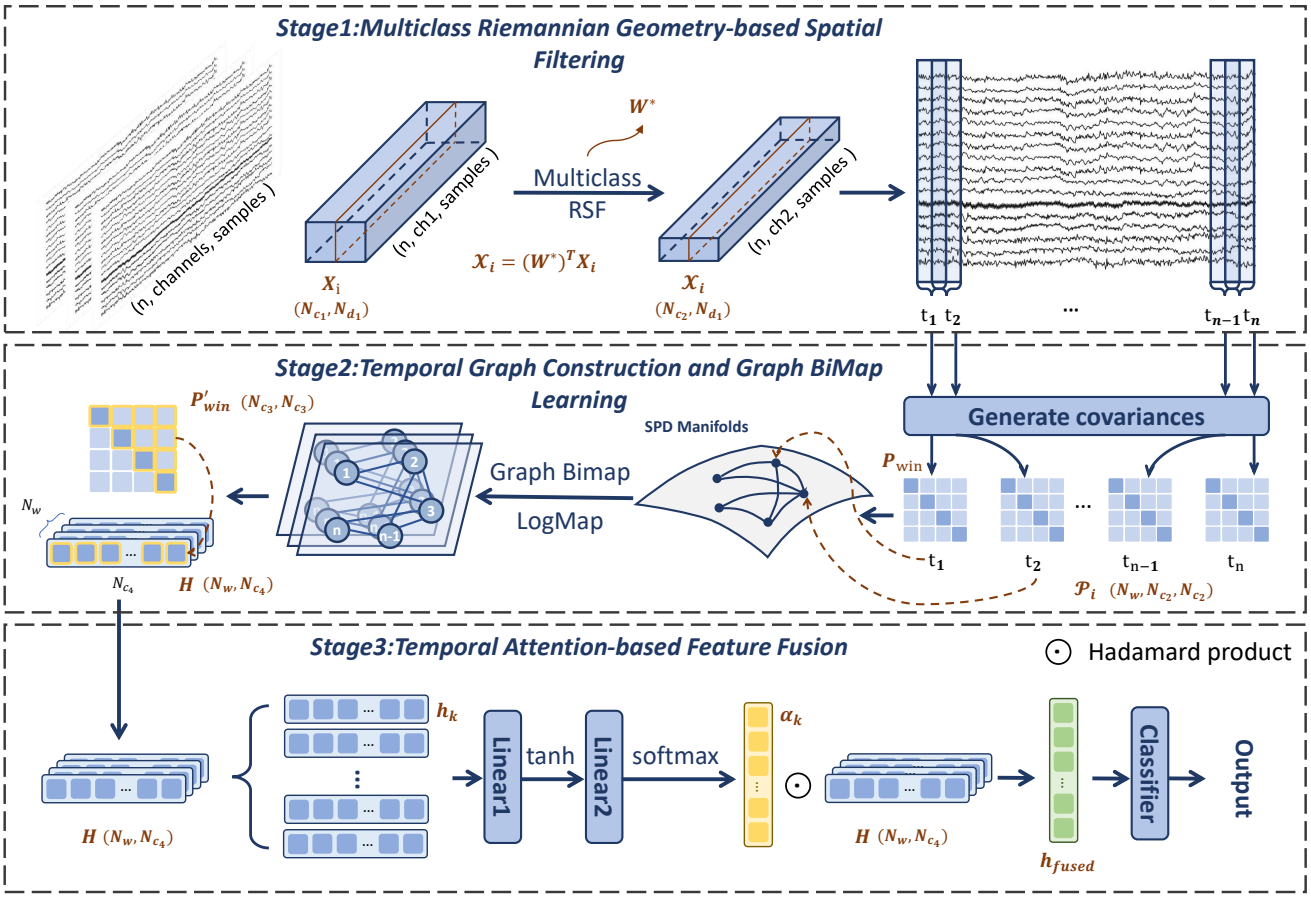


Figure 1: The main framework of Riemannian Spatio-Temporal Graph Neural Network (RST-GNN) for cognitive load detection.

the tangent space and vectorized into an Euclidean feature matrix $\boldsymbol{H} = \{\boldsymbol{h}_k\}_{k=1}^{N_w} \in \mathbb{R}^{N_w \times N_{c_4}}$.

Third, a temporal attention fusion module is utilized to aggregate features across the time windows. Given a sequence of node-level representations $\boldsymbol{H}$, the attention module assigns a learnable importance weight to each window. These weights reflect the relevance of different temporal segments to the classification task. The final fused representation $\boldsymbol{h}_{fused} \in \mathbb{R}^{N_{c_4}}$ is obtained as a weighted combination of all window features, allowing the model to focus more on informative temporal segments while suppressing less relevant or noisy ones. The fused representation $\boldsymbol{h}_{fused}$ is passed through a classifier consisting of fully connected layers to detect the cognitive load level.

2.2. Multiclass Riemannian Geometry-based Spatial Filtering

Let each EEG sample be denoted as $\boldsymbol{X}_i \in \mathbb{R}^{Nc_1 \times N_{d_1}}$, where i represents the index of each sample, N_{c_1} is the number of EEG channels, and N_{d_1} is the number of sampling points per channel. A dataset consisting of N_s samples is denoted as $\mathcal{D} = \{\boldsymbol{X}_i, y_i\}_{i=1}^{N_s}$, where each sample $\boldsymbol{X}_i$ corresponds to a cognitive load label $y_i \in \{1, \cdots, M\}$, with M representing the number of cognitive load levels. The objective of this module is to learn a spatial filter matrix $\boldsymbol{W}^* \in \mathbb{R}^{Nc_1 \times N_{c_2}}$ that maximizes the discriminability of the EEG signals across the cognitive load levels.

2.2.1. Riemannian manifold of SPD matrices

The normalized sample covariance matrix $\boldsymbol{P}_i \in \mathbb{R}^{N_{c_1} \times N_{c_1}}$ can be calculated as:

$$\boldsymbol{P}_i = \frac{\boldsymbol{X}_i \boldsymbol{X}_i^T}{\text{trace}\left(\boldsymbol{X}_i \boldsymbol{X}_i^T\right)}, \tag{1}$$

where trace$(\cdot)$ denotes the trace operation, which sums the diagonal elements of the matrix. The covariance matrices of multi-channel EEG signals are symmetric positive definite (SPD) matrices, meaning they can be described as points on a Riemannian manifold. The Riemannian distance $\delta_R(\boldsymbol{P}_1, \boldsymbol{P}_2)$ between two SPD matrices $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ is defined as:

$$\delta_R\left(\boldsymbol{P}_1, \boldsymbol{P}_2\right) = \left\|\log\left(\boldsymbol{P}_1^{-1}\boldsymbol{P}_2\right)\right\|_F = \left[\sum_{j=1}^{N_{c_1}} \log^2 \lambda_j\right]^{\frac{1}{2}}, \tag{2}$$

where $\log(\cdot)$ denotes the matrix logarithm, $\|\cdot\|_F$ represents the Frobenius norm, and λ_j $(j = 1, \cdots, Nc_1)$ are the eigenvalues of the matrix $\boldsymbol{P}_1^{-1}\boldsymbol{P}_2$. For a set of N sample

covariance matrices, the Riemannian geometric mean is defined as:

$$\bar{\boldsymbol{P}} = \arg\min_{\boldsymbol{P}\in SPD} \sum_{i=1}^{N} \delta_R^2(\boldsymbol{P}, \boldsymbol{P}_i), \tag{3}$$

which minimizes the sum of squared Riemannian distances between $\boldsymbol{P}$ and each of the sample covariance matrices $\boldsymbol{P}_i$. This geometric mean is a key operation for aggregating covariance information in Riemannian space, providing a reference point for further processing.

2.2.2. Multi-classification optimization strategy

For the M-class problem, we aim to maximize the sum of pairwise Riemannian distances between class centroids in the filtered space. The objective function is defined as:

$$\mathcal{F}(\boldsymbol{W}) = \max_{\boldsymbol{W}} \sum_{m_1=1}^{M} \sum_{m_2=m_1+1}^{M} \delta_R^2(\boldsymbol{W}^T \bar{\boldsymbol{P}}_{m_1} \boldsymbol{W}, \boldsymbol{W}^T \bar{\boldsymbol{P}}_{m_2} \boldsymbol{W}), \tag{4}$$

where $\bar{\boldsymbol{P}}_{m_1}$ is the Riemannian geometric mean of class m_1, and $\boldsymbol{W}$ is the transformation matrix that projects the signals into a lower-dimensional space. The optimization problem is solved via Riemannian gradient ascent on the Stiefel manifold, enforcing the orthogonality constraint $\boldsymbol{W}^T\boldsymbol{W} = \boldsymbol{I}$. Initially, $\boldsymbol{W}^{(0)}$ is set as a random orthogonal matrix. At iteration t, the Euclidean gradient $\nabla\mathcal{F}(\boldsymbol{W})$ is computed as:

$$\nabla\mathcal{F}(\boldsymbol{W}) = 2 \sum_{m_1<m_2} \left.\frac{\partial \delta_R^2}{\partial \boldsymbol{W}}\right|_{\boldsymbol{W}=\boldsymbol{W}^{(t)}}, \tag{5}$$

The per-pair gradient is derived via the chain rule as:

$$\begin{aligned}\frac{\partial \delta_R^2}{\partial \boldsymbol{W}} = 4\boldsymbol{W}(&\bar{\boldsymbol{P}}_{m_1} \cdot \mathrm{sym}(\log(\boldsymbol{Q})\boldsymbol{Q}^{-1}) \\ &+ \bar{\boldsymbol{P}}_{m_2} \cdot \mathrm{sym}(\log(\boldsymbol{Q})\boldsymbol{Q}^{-1})),\end{aligned} \tag{6}$$

where

$$\boldsymbol{Q} = \boldsymbol{W}^{-1} \bar{\boldsymbol{P}}_{m_1}^{-1} \bar{\boldsymbol{P}}_{m_2} \boldsymbol{W}, \tag{7}$$

and the symmetrization operation is given by:

$$\mathrm{sym}(\boldsymbol{B}) = (\boldsymbol{B} + \boldsymbol{B}^T)/2. \tag{8}$$

The gradient is projected onto the tangent space of the Stiefel manifold, and the update rule is:

$$\boldsymbol{W}^{(t+1)} = \mathrm{qr}(\boldsymbol{W}^{(t)} + \eta \cdot (\nabla\mathcal{F}(\boldsymbol{W}) - \boldsymbol{W}^{(t)}(\nabla\mathcal{F}(\boldsymbol{W}))^T \boldsymbol{W}^{(t)})), \tag{9}$$

where $\mathrm{qr}(\cdot)$ denotes the QR decomposition for orthonormalization and η is the learning rate. These updates continue until the change in the objective function is below a predefined threshold or a maximum number of iterations is reached, ensuring convergence to an optimal solution.

The final output of the module is obtained by projecting the raw EEG signals through the learned spatial filters. For an input sample $\boldsymbol{X}_i \in \mathbb{R}^{N_{c_1}\times N_{d_1}}$, the filtered signal $\boldsymbol{\mathcal{X}}_i \in \mathbb{R}^{N_{c_2}\times N_{d_1}}$ is computed as:

$$\boldsymbol{\mathcal{X}}_i = (\boldsymbol{W}^*)^T \boldsymbol{X}_i, \tag{10}$$

where $\boldsymbol{W}^* \in \mathbb{R}^{N_{c_1}\times N_{c_2}}$ is the optimized filter matrix.

2.3. Temporal Graph Construction and Graph BiMap Learning

This module is able to capture both spatial and temporal dependencies in EEG signals, ensuring further feature extraction for cognitive load detection by constructing a temporal graph representation and applying the Riemannian graph operations.

2.3.1. Graph construction based on Riemannian distance

First, the filtered EEG signal $\boldsymbol{\mathcal{X}}_i$ is segmented into multiple fixed-length overlapping temporal windows. This results in $\boldsymbol{\mathcal{X}}_i' \in \mathbb{R}^{N_w\times N_{c_2}\times N_{d_2}}$, where N_w is the number of windows, N_{c_2} is the number of channels after spatial filtering, and N_{d_2} is the length of each time window. Next, we compute a covariance matrix $\boldsymbol{P}_{\text{win}} \in \mathbb{R}^{N_{c_2}\times N_{c_2}}$ per window, which represents a graph node, and all windows are represented as $\boldsymbol{\mathcal{P}}_i \in \mathbb{R}^{N_w\times N_{c_2}\times N_{c_2}}$.

The adjacency matrix $\boldsymbol{A} \in \mathbb{R}^{N_w\times N_w}$ encodes dynamic connectivity between the windows and is defined as:

$$\boldsymbol{A}_{pq} = \begin{cases} \exp\left(-\frac{\delta_R^2(\boldsymbol{P}_p, \boldsymbol{P}_q)}{2\sigma^2}\right), & \text{if } q \in \mathcal{N}_K(p); \\ 0, & \text{otherwise .} \end{cases} \tag{11}$$

Here, $\delta_R(\cdot,\cdot)$ denotes the Riemannian distance between two covariance matrices, $\mathcal{N}_K(p)$ denotes the set of the top-K nearest neighbors of node p based on geodesic distance, and σ is a parameter that adapts to the trial-specific distance distribution. Although fixed-length temporal windows are used to segment EEG signals for covariance estimation, the proposed framework does not impose sequential adjacency constraints on temporal relationships. Instead, graph connections are established based on similarity in the Riemannian covariance space, allowing non-adjacent windows with similar neural representations to be directly linked. This enables the explicit modeling of non-local temporal dependencies and recurrent neural states, mitigating potential limitations introduced by fixed-window discretization.

2.3.2. Graph BiMap layer

After constructing the temporal graph to encode relationships between windows, discriminative representations are extracted using an SPD matrix-valued graph neural network, whose layer-wise propagation rule is defined as:

$$\boldsymbol{H}^{(l+1)} \leftarrow \mathrm{RBN}(\mathrm{ReEig}(\mathrm{BiMap}((\bar{\boldsymbol{D}}^{-\frac{1}{2}} \bar{\boldsymbol{A}}^{(l)} \bar{\boldsymbol{D}}^{-\frac{1}{2}}) \boldsymbol{H}^{(l)}))),$$

(12)

where $\bar{\boldsymbol{A}}^{(l)} = \boldsymbol{A}^{(l)} + \boldsymbol{I}$, $\bar{\boldsymbol{D}}_{pp} = \sum_q \bar{\boldsymbol{A}}_{pq}^{(l)}$, and $\boldsymbol{\omega}^{(l)}$ is a trainable transformation matrix with full-row rank. $\boldsymbol{H}^{(l)} \in \mathbb{R}^{N_w \times N_{c_2}^2}$ is the node feature in the l-th layer. The core operation in this propagation is the bilinear mapping (BiMap), which replaces the traditional linear projection in Euclidean GNNs. BiMap is defined as:

$$\text{BiMap}(\boldsymbol{S}) = \boldsymbol{\omega}\boldsymbol{S}\boldsymbol{\omega}^T, \tag{13}$$

where $\boldsymbol{S}$ represents any SPD matrix and $\boldsymbol{\omega}$ is a trainable transformation matrix. This operation transforms the SPD matrix into a lower-dimensional SPD matrix while preserving its symmetric positive definiteness, provided that $\boldsymbol{\omega}$ has full row rank.

The ReEig operation is then applied to compute the matrix logarithm and perform eigenvalue adjustment, as follows:

$$\text{ReEig}(\boldsymbol{S}) = \boldsymbol{U} \cdot \text{diag}(\max(\lambda, \epsilon)) \cdot \boldsymbol{U}^T, \tag{14}$$

where $\boldsymbol{S} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T$ is the eigendecomposition of $\boldsymbol{S}$, and $\epsilon > 0$ is a small constant to avoid numerical instability by ensuring all eigenvalues are strictly positive.

Riemannian Batch Normalization (RBN) is applied to reduce domain shifts and stabilize training across SPD manifolds. It is defined as:

$$\text{RBN}(\boldsymbol{S}) = \boldsymbol{\omega}_r^{\frac{1}{2}} \cdot (\bar{\boldsymbol{P}}^{-\frac{1}{2}} \boldsymbol{S} \bar{\boldsymbol{P}}^{-\frac{1}{2}}) \cdot \boldsymbol{\omega}_r^{\frac{1}{2}}, \tag{15}$$

where $\bar{\boldsymbol{P}}$ is the Riemannian mean of all SPD matrices in the batch and $\boldsymbol{\omega}_r \in SPD$ is a learnable SPD matrix that acts as a scaling parameter.

2.3.3. LogMap layer

After the SPD graph convolutional processing, each node is represented as an SPD matrix that resides on a Riemannian manifold. However, to enable subsequent operations, such as attention-based fusion and classification in the Euclidean space, these manifold-valued features need to be projected back to the tangent space. This is achieved by applying the logarithmic map (LogMap), and the LogMap operation is defined as:

$$\text{LogMap}(\boldsymbol{S}) = \bar{\boldsymbol{P}}^{\frac{1}{2}} \cdot \log(\bar{\boldsymbol{P}}^{-\frac{1}{2}} \boldsymbol{S} \bar{\boldsymbol{P}}^{-\frac{1}{2}}) \cdot \bar{\boldsymbol{P}}^{\frac{1}{2}}, \tag{16}$$

where $\log(\cdot)$ denotes the matrix logarithm, computed via eigenvalue decomposition, and $\bar{\boldsymbol{P}}$ is the Riemannian mean of all SPD matrices in the batch.

After a series of Riemannian graph operations, a sequence of node-level outputs across temporal windows is obtained, each represented as a symmetric matrix $\boldsymbol{P}'_{\text{win}} \in \mathbb{R}^{N_{c_3} \times N_{c_3}}$ in the tangent space, where N_{c_3} indicates the output dimension of this module. To enable subsequent Euclidean processing, the upper triangular part (including the diagonal) of each matrix is flattened into a vector $\boldsymbol{h}_k, k = 1, 2, \cdots, N_w$ of length N_{c_4}. Finally, the matrix of node-level features is represented as $\boldsymbol{H} = \{\boldsymbol{h}_k\}_{k=1}^{N_w} \in \mathbb{R}^{N_w \times N_{c_4}}$, where $N_{c_4} = N_{c_3}(N_{c_3} + 1)/2$.

2.4. Temporal Attention-based Feature Fusion

The attention mechanism is applied to fuse the window-level features. For each window feature vector $\boldsymbol{h}_k$, an attention score s_k is computed as:

$$s_k = \boldsymbol{w}_2^T \tanh(\boldsymbol{W}_1 * \boldsymbol{h}_k + \boldsymbol{b}_1) + b_2, \tag{17}$$

where $\boldsymbol{W}_1$ and $\boldsymbol{w}_2$ are learnable parameters, and $\boldsymbol{b}_1$ and b_2 denote the bias terms. The attention weight α_k for each window is then computed using the softmax function:

$$\alpha_k = \text{softmax}(s_k). \tag{18}$$

Finally, the fused feature vector $\boldsymbol{h}_{fused}$ for the entire EEG sample is obtained as the weighted sum of all window features:

$$\boldsymbol{h}_{fused} = \sum_{k=1}^{N_w} \alpha_k \boldsymbol{h}_k. \tag{19}$$

This attention-based fusion enables the model to automatically focus on the most informative time segments while suppressing less relevant or noisy ones, thereby enhancing classification performance.

3. Experiment Setup

3.1. Dataset Descriptions

In this study, two public cognitive load EEG datasets, namely Datasets I and II, are utilized to evaluate the proposed RST-GNN method. The details of both datasets are described as follows.

3.1.1. Dataset I

Firstly, we employed the publicly available COG-BCI dataset (Hinss et al., 2023), which was designed for passive brain-computer interface (pBCI) applications. The dataset includes EEG recordings from 29 participants across three sessions, covering four cognitive tasks. In this study, we selected only the N-back task for analysis, as it is widely used to elicit distinct cognitive load levels in a controlled setting. EEG signals were recorded using a 64-channel Brain Products ActiCap system following the international 10-20 electrode placement, with a sampling rate of 500 Hz and FPz as the reference. The recordings are provided in BIDS-compliant format and include synchronized behavioral and subjective annotations (e.g., NASA-TLX). This dataset allows for validation of model performance under realistic cognitive workload conditions and across multiple subjects.

3.1.2. Dataset II

To further evaluate the generalizability of the proposed model, the second EEG dataset (Wang et al., 2023) was collected from 20 male participants (mean age: 27.65 years, range: 22-42) who held valid driving licenses. EEG signals were recorded using a 24-channel Waveguard Net system (ANT Neuro) based on the international 10–20 system, with a sampling rate of 500 Hz and reference at Cz. Participants

performed four driving scenarios within a simulated environment (City Car Driving software), each designed to elicit four levels of cognitive load: 1) motorway no cars: low-load baseline; 2) motorway with cars: moderate load; 3) urban no cars: intermediate load; 4) urban with cars: high load. The order of task presentation was counterbalanced across participants to avoid sequence effects. Subjective ratings of cognitive load (1-10 scale) confirmed significant differences across conditions (ANOVA: $p < 0.001$), with "Urban-with cars" receiving the highest average rating (4.9) and "Motorway-no cars" the lowest (1.9).

3.2. Data Preprocessing

To ensure signal quality and consistency across datasets, a standardized preprocessing pipeline was applied. EEG signals were first band-pass filtered between 0.2 and 40 Hz to remove low-frequency drifts and high-frequency noise. To suppress physiological artifacts such as eye blinks and muscle activity, Independent Component Analysis (ICA) was performed using the FastICA algorithm. The ICLabel toolbox was employed to automatically classify components, and those identified as eye, cardiac, or muscle with confidence above 90% were removed. For Dataset I, the preprocessed signals were segmented into non-overlapping 4-second epochs. For Dataset II, data segmentation followed the experimental design, where each trial was treated as a separate sample with a duration of 2 seconds. All preprocessed segments were subsequently used for model training and testing.

3.3. Implementation Details

To evaluate model performance, we adopted a 10-fold cross-validation strategy. Data was split into ten equal folds; in each iteration, nine folds were used for training, and one for testing, and this process was repeated ten times. The reported performance was averaged across all folds to reduce sensitivity to data partitioning. All experiments were performed in a subject-dependent setting, i.e., training and testing were conducted within each subject. For Dataset I, cross-validation was independently conducted for each subject, and the final performance was obtained by averaging across subjects. For Dataset II, which contains multiple sessions, cross-validation was conducted within each session. Session-level results were averaged to produce subject-wise performance, which was then aggregated across subjects.

The model was trained using the RiemannianAdam optimizer from the Geoopt library, with an initial learning rate set to 1×10^{-3}. The cross-entropy loss function was employed to supervise the classification task. The training process was carried out for 50 epochs with a mini-batch size of 16.

All experiments were conducted on a workstation equipped with an Intel Xeon Silver 4214R CPU, NVIDIA GeForce RTX 3090 GPU (24 GB VRAM) with CUDA 12.6 and cuDNN v9.5.1, and 256 GB RAM. The model was implemented in PyCharm using PyTorch (v2.7.1) and Geoopt (v0.5.0).

3.4. Baseline Methods

To evaluate the performance of the proposed RST-GNN model, we compared it against several representative EEG-based deep learning models, including both CNNs and GNNs. These baselines were selected based on their effectiveness in prior EEG classification tasks and their architectural diversity.

DeepConvNet(Schirrmeister et al., 2017) is a widely used CNN model that processes raw EEG data via temporal and spatial convolutions. It serves as a strong end-to-end baseline for EEG decoding.

EEGNet(Lawhern et al., 2018) is a compact and efficient CNN architecture tailored for EEG-based BCIs. Its use of depthwise and separable convolutions enables efficient spatio-temporal feature learning.

TSception(Ding et al., 2022) introduces multi-scale temporal kernels and asymmetric spatial filters to model EEG temporal dynamics and spatial asymmetry.

DGCNN(Song et al., 2018) dynamically constructs graphs from EEG signals using learned functional connectivity. It applies spectral graph convolution to capture time-varying inter-channel relationships.

RGNN(Zhong et al., 2020) is a biologically motivated graph neural network that combines anatomical priors and emotion-related connectivity, and includes domain adaptation and label smoothing mechanisms.

All baseline models were re-implemented or adapted from publicly available repositories and trained under the same preprocessing pipeline and data splits as our method.

3.5. Evaluation Criteria

To comprehensively assess the performance of the proposed model in the cognitive load detection task, we adopt four commonly used evaluation metrics: accuracy, precision, recall, and F1-score. Let y_i and $\hat{y}_i$ denote the ground truth and the predicted label for the i-th sample. Accuracy is defined as the proportion of correctly predicted samples among all samples, formulated as:

$$Accuracy = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}\left(y_i = \hat{y}_i\right), \tag{20}$$

where $\mathbb{I}$ is the indicator function and N is the total number of samples.

For each class, precision and recall are calculated based on the number of true positives (TP), false positives (FP), and false negatives (FN). Specifically, precision measures the proportion of correctly predicted positive samples among all predicted positives:

$$Precision = \frac{TP}{TP + FP}. \tag{21}$$

Recall reflects the proportion of correctly predicted positive samples among all actual positives:

$$Recall = \frac{TP}{TP + FN}. \tag{22}$$

Table 1
Comparison of RST-GNN and baseline methods on Datasets I and II using four classification metrics (mean ± standard deviation).

Dataset	Methods	Accuracy(%)	Precision(%)	Recall(%)	F1-score(%)
I	DeepConvNet	76.72 ± 9.54 ***	78.84 ± 9.09 ***	76.70 ± 9.55 ***	75.82 ± 10.05 ***
	EEGNet	63.61 ± 7.73 ***	64.40 ± 7.71 ***	63.61 ± 7.72 ***	63.22 ± 7.79 ***
	TSception	85.12 ± 10.40 ***	86.56 ± 9.14 ***	85.12 ± 10.40 ***	84.02 ± 12.09 ***
	DGCNN	92.28 ± 4.54 ***	92.67 ± 4.38 ***	92.26 ± 4.54 ***	92.23 ± 4.56 ***
	RGNN	59.37 ± 5.55 ***	60.25 ± 5.61 ***	59.37 ± 5.55 ***	58.77 ± 5.68 ***
	RST-GNN	**96.65 ± 2.43**	**96.46 ± 2.51**	**96.64 ± 2.43**	**96.33 ± 2.65**
II	DeepConvNet	75.61 ± 12.86 ***	76.66 ± 12.52 ***	74.90 ± 12.84 ***	74.06 ± 13.46 ***
	EEGNet	59.99 ± 14.65 ***	59.62 ± 14.69 ***	58.74 ± 14.38 ***	58.68 ± 14.48 ***
	TSception	85.21 ± 9.07 ***	86.32 ± 8.07 ***	84.39 ± 9.40 ***	84.14 ± 9.58 ***
	DGCNN	89.22 ± 6.77 ***	89.41 ± 6.87 ***	88.68 ± 6.88 ***	88.65 ± 7.03 ***
	RGNN	74.16 ± 13.80 ***	74.47 ± 13.57 ***	73.17 ± 13.72 ***	72.99 ± 13.80 ***
	RST-GNN	**94.40 ± 3.85**	**94.64 ± 3.96**	**93.92 ± 4.27**	**93.98 ± 4.32**

The asterisks indicate significant differences between the five baseline methods and the proposed RST-GNN method obtained by two-tailed Wilcoxon signed-rank tests (∗p<0.05, ∗∗p<0.01, ∗∗∗p<0.001).

The F1-score is the harmonic mean of precision and recall:

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}. \tag{23}$$

4. Results and Discussions

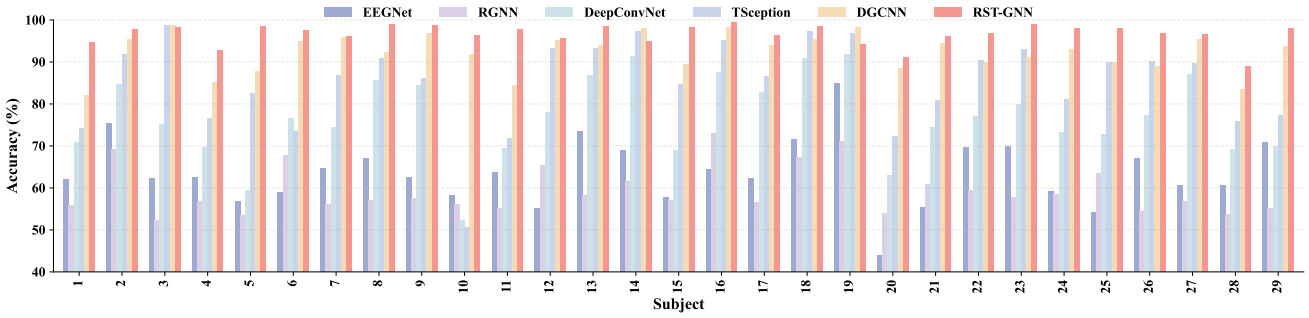
4.1. Overall Detection Performance

To assess the effectiveness of the proposed RST-GNN model, experiments were conducted on two publicly available EEG datasets, comparing against representative CNN- and GNN-based baselines. For fair comparison, all baseline models were implemented following the architectures and hyperparameter configurations reported in their original publications. All models, including the proposed RST-GNN, were trained under the same preprocessing pipeline, cross-validation protocol, and training settings to ensure a fair and consistent evaluation.

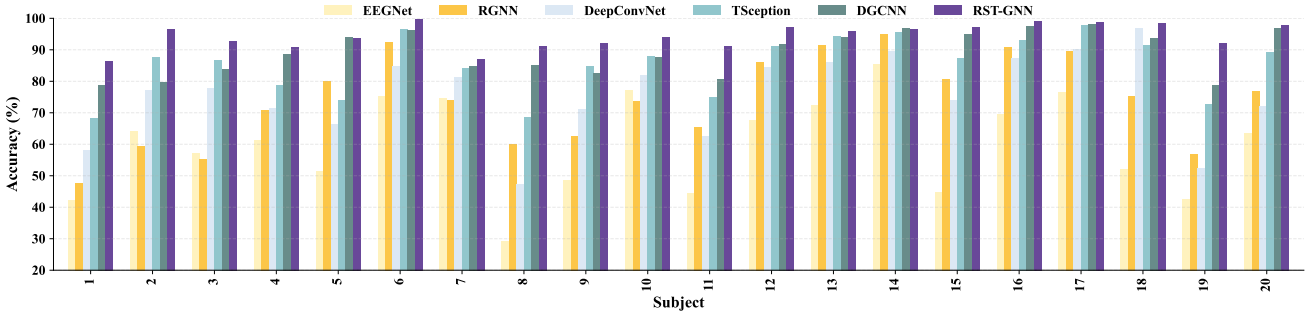
As shown in Table 1, RST-GNN consistently outperformed all baseline models across both datasets in terms of accuracy, precision, recall, and F1-score. The two-tailed Wilcoxon signed-rank test was conducted on subject-wise accuracies between RST-GNN and each baseline model. The results indicate that the performance gains are statistically significant across both Dataset I (n = 29) and Dataset II (n = 20), with all comparisons reaching $p < 0.001$ (***). These findings confirm that the observed improvements are consistent across subjects rather than being driven by a small subset of participants. On Dataset I, a clear performance margin was observed between RST-GNN and competing methods, demonstrating its superior capability in modeling spatiotemporal dependencies for cognitive load decoding. Similarly, on Dataset II, RST-GNN maintained consistent advantages across all evaluation metrics while exhibiting lower performance variance.

To further illustrate inter-subject consistency, Fig. 2 presents the subject-wise classification accuracies for both datasets. As shown in the bar charts, RST-GNN achieves higher accuracy for nearly all subjects compared with the baseline models, visually demonstrating stable and consistent performance gains across individuals.

To further visualize model performance, Fig. 3 presents a comparative distribution of classification accuracy across all methods on both datasets. Each subplot combines scatter plots, boxplots, and violin plots, capturing subject-level performance variability, central tendency, and overall distribution. Across both datasets, the proposed RST-GNN consistently outperforms. It exhibits the highest median accuracy and the most compact interquartile ranges, indicating strong central performance and reduced variability. In contrast, baseline models such as EEGNet and RGNN show wider dispersions and more pronounced outliers, suggesting less consistent generalization across subjects. These distributional patterns reinforce earlier quantitative findings: by explicitly modeling relational spatio-temporal dependencies in EEG data, RST-GNN not only improves average performance but also ensures more stable predictions across individuals.

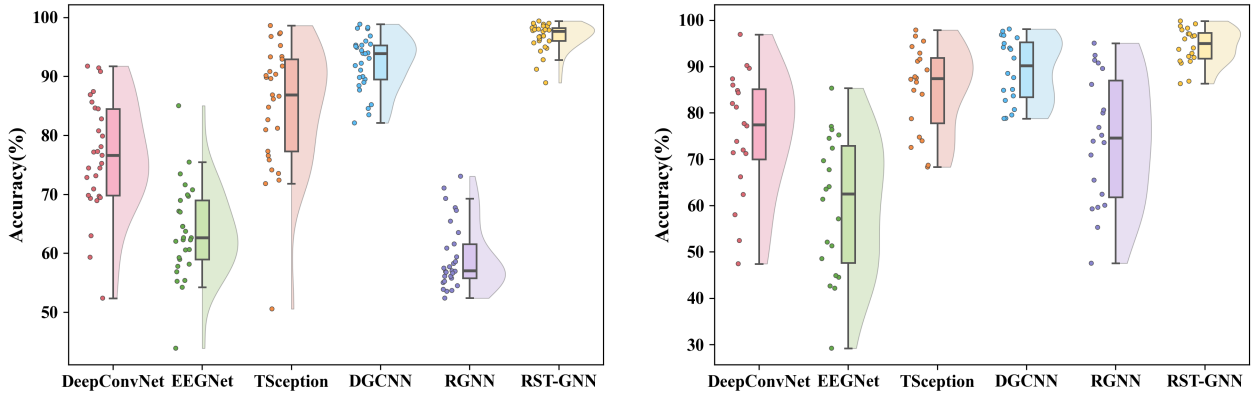


(a) Dataset I



(b) Dataset II

Figure 2: Subject-wise classification accuracy (%) of RST-GNN and baseline methods on Dataset I and Dataset II.



(a) Dataset I

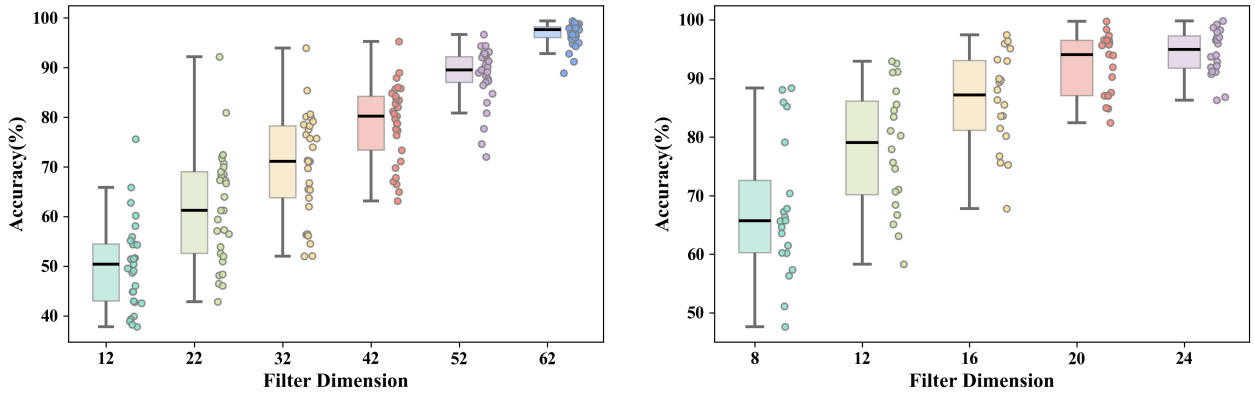
(b) Dataset II

Figure 3: Performance comparison of DeepConvNet, EEGNet, TSception, DGCNN, RGNN, and RST-GNN on the Datasets I and II. The violin plots depict the distribution of accuracy across subjects; the boxplots indicate the median and interquartile range; and the scatter points represent subject-wise individual accuracy scores.

4.2. Parameter Sensitivity Analysis

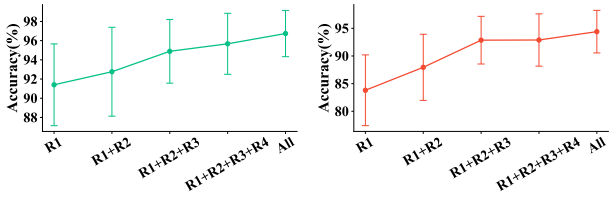
4.2.1. The dimension of the spatial filter

To investigate the effect of the spatial filter number N_{c_2}, we conducted a sensitivity analysis on this hyperparameter. For Dataset I, N_{c_2} was varied from 12 to 62 in increments of 10, where 62 corresponds to the original number of EEG channels. For Dataset II, the value was adjusted from 8 to 24 with a step size of 4, with 24 matching the original channel count. As shown in the Fig. 4, the median classification accuracy increases steadily as the number of filters grows. At the same time, variability between subjects is reduced, as reflected in narrower interquartile ranges and more compact scatter distributions. This indicates that larger filter numbers allow the model to capture richer spatial information from EEG signals, thereby enhancing performance stability and generalization. In particular, the best performance on both datasets is achieved when the filter number is equal to the original EEG channel count.



(a) Dataset I (b) Dataset II

Figure 4: Effect of filter number on classification accuracy for Dataset I and Dataset II. The boxplots indicate the median and interquartile range, and the scatter points represent subject-wise individual accuracy scores.



(a) Dataset I (b) Dataset II

Figure 5: Effect of brain regions selection on classification accuracy for Dataset I and Dataset II. The line plots show the mean and standard deviation of accuracy as the number of included brain regions increased from frontal (R1) to full coverage (R1+R2+R3+R4+R5).

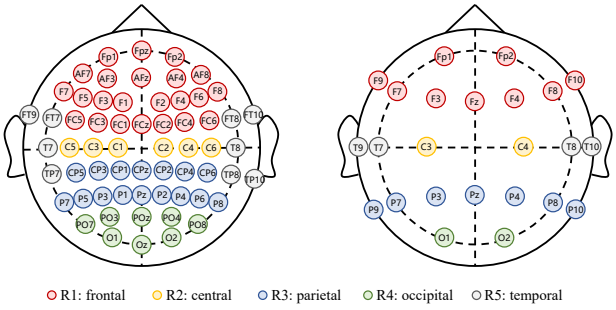
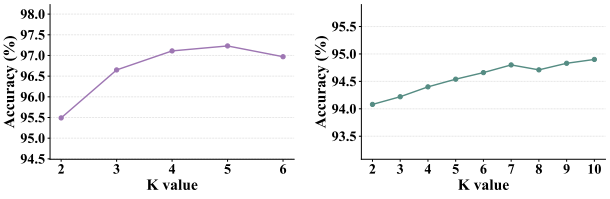


Figure 6: Topographical representation of the EEG electrode configurations for the two datasets. Dataset I (left) and Dataset II (right) include different subsets of electrodes following the international 10–20 system. Only the recorded electrodes are displayed, with regions color-coded as follows: frontal (red), central (yellow), parietal (blue), occipital (green), and temporal (gray).

4.2.2. Brain regions selection

To investigate how different cortical regions contribute to cognitive load classification, we gradually increased the number of EEG electrodes involved in the model input.



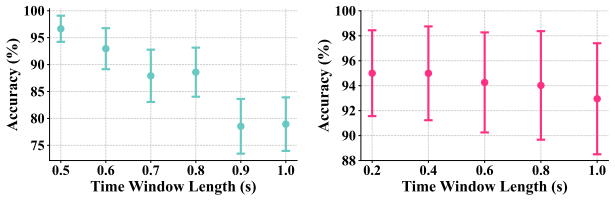
(a) Dataset I (b) Dataset II

Figure 7: Effect of the temporal graph parameter K on classification accuracy for Dataset I and Dataset II.

As shown in Fig.6, the electrodes were grouped into five cumulative regions according to the 10–20 system: frontal (R1), central (R2), parietal (R3), occipital (R4), and temporal(R5). Fig.5 shows the changes in classification accuracy for Dataset I and Dataset II. For both datasets, the model achieved a reasonably high accuracy even when using only the frontal electrodes, indicating that the frontal region alone contains rich task-relevant information related to cognitive effort. The inclusion of the central region led to a substantial improvement, suggesting that motor-related and midline activities further support cognitive load discrimination. Adding the parietal region provided an additional but smaller gain, while including the occipital region caused the performance to plateau, implying that most discriminative information was already captured by the frontal, central, and parietal electrodes. Overall, these results suggest that while the frontal region plays a dominant role in reflecting cognitive load, integrating signals from additional regions, particularly the central and parietal cortices, provides complementary spatial information that enhances classification performance.

4.2.3. The Number of Temporal Neighbors K

The parameter K determines the number of nearest neighbors used during graph construction and therefore



(a) Dataset I (b) Dataset II

Figure 8: Effect of time window length on classification accuracy for Dataset I and Dataset II. The vertical error bars represent standard deviations.

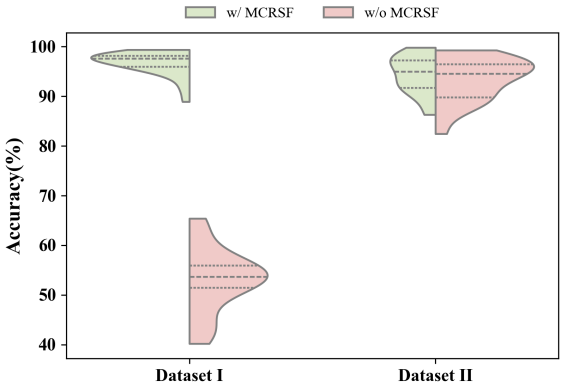


Figure 9: Performance comparison of "w/ MCRSF" and "w/o MVRSF" configurations on the Dataset I and Dataset II.

controls the connectivity of the EEG graph. To evaluate the sensitivity of the proposed RST-GNN to this parameter, experiments were conducted with different values of K on both datasets.

The range of K differs for the two datasets due to the different temporal lengths of the EEG segments. In our framework, EEG signals are segmented using a sliding window of 0.5 s, where each window corresponds to a temporal node in the constructed graph. For Dataset I, each trial lasts only 2 s, resulting in a limited number of temporal nodes after segmentation. Therefore, K is varied from 2 to 6 to ensure that meaningful neighborhood relationships can be constructed without exceeding the available nodes. In contrast, Dataset II contains longer EEG segments, producing more temporal nodes and allowing a wider exploration range of K from 2 to 10.

The experimental results are illustrated in Fig.7, where the classification accuracy is plotted as a function of K for both datasets. As shown in the figure, RST-GNN achieves stable performance across different values of K. For Dataset I, the classification accuracy increases rapidly as K grows from 2 to 5 and reaches the best performance at $K = 5$, followed by a slight decrease when K increases to 6. This suggests that incorporating an appropriate number of neighboring nodes helps capture richer temporal relationships, while overly large neighborhood sizes may introduce redundant connections. For Dataset II, the accuracy gradually improves as K increases and becomes relatively stable when K is larger than 6, indicating that the proposed model is not overly sensitive to the choice of K. Overall, these results demonstrate that RST-GNN maintains robust performance within a reasonable range of K.

4.2.4. Influence of time window length

To investigate the effect of the time window length on the performance of the proposed method, we conducted a parameter sensitivity analysis by varying the window length while keeping all other experimental settings unchanged. The results are shown in Fig.8, where the points represent the mean classification accuracy across subjects and the error bars denote the corresponding standard deviations. For Dataset I, the window length was varied from 0.2 s to 1.0 s with an interval of 0.1 s, while for Dataset II, it ranged from 0.5 s to 1.0 s with an interval of 0.2 s.

As illustrated in Fig.8, the best performance is achieved with relatively short window lengths (0.2 s for Dataset I and 0.5 s for Dataset II). As the window length increases, the classification accuracy gradually decreases, and the variability across subjects becomes larger. This phenomenon suggests that excessively long temporal windows may smooth out transient neural dynamics, thereby reducing the discriminative temporal information available for the model.

Overall, these results indicate that maintaining a relatively high temporal resolution is beneficial for the proposed RST-GNN framework, as it enables the temporal graph to capture more detailed dynamic patterns in EEG signals related to cognitive workload.

4.3. Ablation Study

To further investigate the contribution of each module within the proposed framework, ablation studies were conducted from three perspectives: the multiclass Riemannian geometry-based spatial filter, the temporal graph construction and graph BiMap learning, and the temporal attention. The results are summarized in Table 2.

4.3.1. The ablation of the multiclass Riemannian geometry-based spatial filter

To investigate the effect of the multiclass Riemannian geometry-based spatial filter, we remove this module and keep the rest unchanged. From Table 2 and Fig. 9, it is evident that removing the spatial filter substantially degrades both accuracy and stability, particularly on Dataset I. With MCRSF, the distributions shift upward with narrower interquartile ranges, indicating higher accuracy and greater robustness. In contrast, removing it results in a dramatic 43.73% drop on Dataset I and a 0.86% decline on Dataset II, accompanied by wider spreads and reduced consistency across subjects. This confirms that spatial filtering plays a critical role in enhancing the discriminative power of EEG representations by projecting raw channel signals into a more informative subspace.

Table 2
Comparative results from the ablation study on two benchmarking datasets, highlighting the impact of each component (mean ± standard deviation).

Dataset	Conditions	Accuracy(%)	Precision(%)	Recall(%)	F1-score(%)
I	w/o MCRSF	52.92 ± 6.45	39.71 ± 8.90	53.07 ± 6.42	41.88 ± 8.06
	w/ transformer	93.32 ± 3.74	93.68 ± 3.62	93.32 ± 3.74	93.21 ± 3.91
	w/o attention	91.61 ± 4.49	90.57 ± 5.43	91.59 ± 4.50	90.12 ± 5.48
	RST-GNN	**96.65 ± 2.43**	**96.46 ± 2.51**	**96.64 ± 2.43**	**96.33 ± 2.65**
II	w/o MCRSF	93.54 ± 4.35	93.88 ± 4.11	93.08 ± 4.56	93.15 ± 4.54
	w/ transformer	89.88 ± 5.08	90.23 ± 5.23	89.34 ± 5.43	89.29 ± 5.51
	w/o attention	92.10 ± 5.51	92.74 ± 5.39	91.41 ± 6.09	91.22 ± 6.54
	RST-GNN	**94.40 ± 3.85**	**94.64 ± 3.96**	**93.92 ± 4.27**	**93.98 ± 4.32**

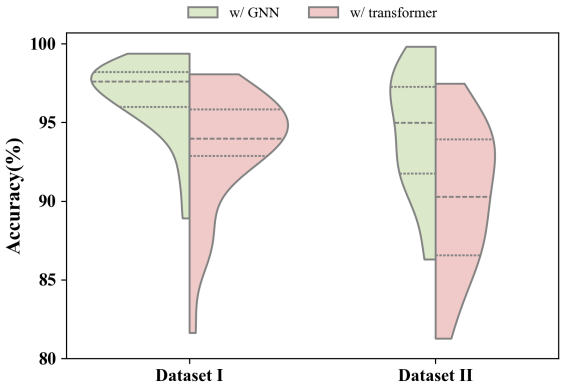


Figure 10: Performance comparison of "w/ GNN" and "w/ transformer" configurations on the Dataset I and Dataset II.

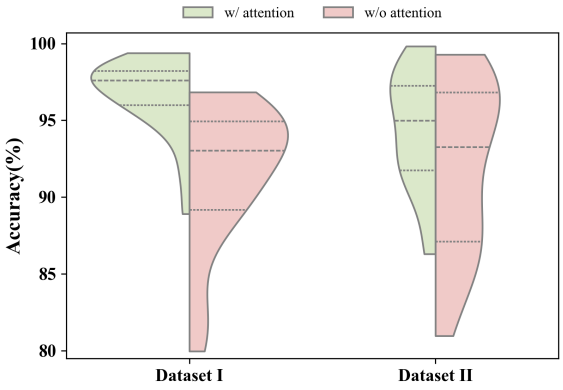


Figure 11: Performance comparison of "w/ attention" and "w/o attention" configurations on the Dataset I and Dataset II.

4.3.2. The ablation of the temporal graph construction and graph BiMap learning

In order to assess the importance of the temporal graph-based modeling strategy, the GNN backbone was replaced with a Transformer architecture. Specifically, after dividing the time window and calculating the covariance matrix, take the upper diagonal pair (including the diagonal) of the covariance matrix and input it into the transformer. From Table 2 and Fig. 10, we observe that replacing GNN with Transformer leads to a consistent reduction in both accuracy and stability across datasets. The GNN-based model achieves higher medians with more compact distributions, reflecting stronger central performance and robustness. In contrast, the Transformer variant shows a decline of 3.33% on Dataset I and 4.52% on Dataset II, with wider spreads indicating greater variability across subjects. The results suggests that capturing the similarity structure among temporal segments provides more effective representations for this task compared with modeling their strict sequential order.

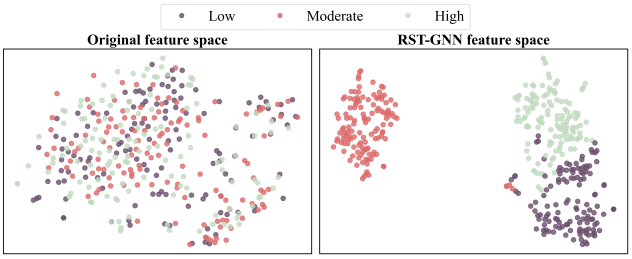
4.3.3. The ablation of the temporal attention

To investigate the effect of the temporal attention, we remove this module and keep the rest unchanged. From Table 2 and Fig. 11, it can be seen that incorporating the attention mechanism improves both accuracy and stability on the two datasets. The model with attention exhibits higher medians and more concentrated distributions, indicating enhanced robustness. In contrast, removing attention results in a decrease of 5.04% on Dataset I and 2.3% on Dataset II, with wider spreads and more pronounced variability across subjects. This suggests that the attention module is beneficial for adaptively emphasizing more informative features, thereby further boosting the classification accuracy.

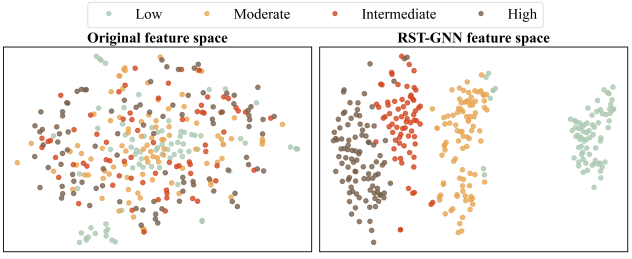
4.4. Discussion

4.4.1. Visualization of feature distributed with t-SNE

To better understand the representational capacity of the proposed RST-GNN for cognitive load detection, we employed t-SNE to visualise the feature distributions of both the raw EEG features and the features extracted by RST-GNN on two datasets (Fig. 12). For Dataset I, the raw features appear highly scattered with substantial overlaps among the three cognitive load conditions, suggesting limited discriminability in the original feature space. By contrast, the RST-GNN features exhibit a markedly improved structure, where samples belonging to the same class form more compact clusters and inter-class boundaries become clearer(Fig. 12a). A similar phenomenon can be observed for Dataset II. As shown in Fig. 12b, the raw features remain entangled and



(a) Dataset I



(b) Dataset II

Figure 12: t-SNE visualization of RST-GNN in the subject-dependent experiments on two datasets. (a) Original feature space and RST-GNN feature space on Dataset I. (b) Original feature space and RST-GNN feature space on Dataset II.

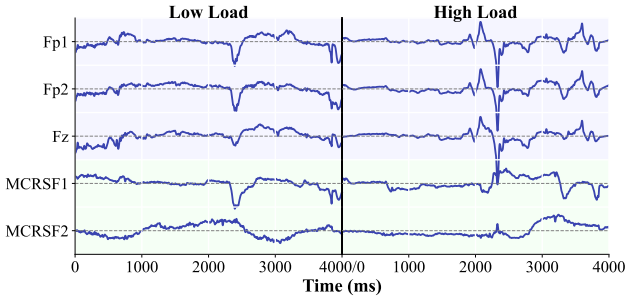
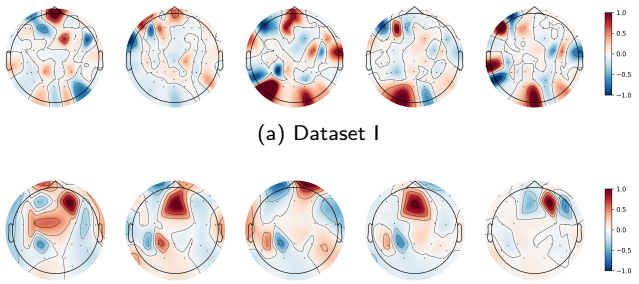


Figure 13: Visualization of EEG signals before and after spatial filtering. The first three rows show raw EEG channels (Fp1, Fp2, and Fz), while the last two rows present the signals after the proposed MCRSF spatial filtering. The left and right panels correspond to the low and high cognitive load conditions.

lack class-specific separability. In comparison, the RST-GNN features demonstrate a well-organised distribution, where distinct clusters emerge corresponding to different cognitive load levels. The results indicate that RST-GNN adapts well to different cognitive load conditions and learns features with stronger class discriminability by effectively capturing more discriminative spatio-temporal characteristics of EEG signals.

4.4.2. Visualization of the MCRSF

To illustrate the effectiveness of the proposed spatial filtering method, we visualize representative EEG signals before and after spatial filtering in Fig.13. The first three rows correspond to raw EEG channels (Fp1, Fp2, and Fz), while the last two rows show the signals obtained by the



(a) Dataset I

(b) Dataset II

Figure 14: Representative MCRSF filters learned from Subject 1 on two datasets. Five representative components were selected for each dataset. The topographic maps reveal the spatial distribution patterns captured by the Riemannian spatial filters.

MCRSF. For visualization purposes, all signals are normalized. The left half of the figure represents the low cognitive load condition, and the right half represents the high cognitive load condition. Compared with the raw EEG channels, the spatially filtered signals exhibit clearer amplitude variations and more structured temporal patterns. This indicates that the MCRSF enhances discriminative signal components while suppressing noise and irrelevant activity, resulting in more informative signals for subsequent cognitive load classification.

To further evaluate the effect of the proposed Riemannian spatial filtering, we visualized the learned spatial filters from Subject 1 (Fig.14). The selected components show clear activations in frontal, parietal, and occipital regions, which are commonly involved in cognitive processing. These results indicate that the spatial filters enhance task-relevant brain activity, supporting their effectiveness in improving discriminative EEG representations for cognitive load classification.

4.4.3. Additional comparison on Dataset I

To further strengthen the experimental comparison, we additionally evaluated our method against several recent models reported in (Kongwudhikunakorn et al., 2025) on Dataset I. Since these methods were implemented under a different preprocessing pipeline and experimental protocol, we followed the same settings described in (Kongwudhikunakorn et al., 2025) to ensure a fair comparison.

Specifically, the EEG signals were preprocessed using band-pass filtering between 4-12 Hz and downsampled to 250 Hz, following the procedure reported in (Kongwudhikunakorn et al., 2025). Moreover, the original three-condition task was reformulated as a binary classification problem by considering the 0-back condition as low mental workload and the 2-back condition as high mental workload. In terms of the evaluation protocol, the within-subject classification setting adopted a 70:20:10 split for training, validation, and testing sets, instead of the ten-fold cross-validation used in our main experiments.

Table 3
Performance comparison of RST-GNN and existing methods on Dataset I under the experimental protocol reported in (Kongwudhikunakorn et al., 2025). The results are presented as mean ± standard deviation across subjects.

Methods	ACC(%)	SEN(%)	SPEC(%)	F1(%)
FBCSP-SVM	68.81 ± 2.75	64.65 ± 2.35	69.98 ± 1.96	67.20 ± 2.24
EEGNet	71.07 ± 3.68	78.18 ± 5.49	55.38 ± 6.11	64.47 ± 3.52
EEGLearn	79.23 ± 2.61	76.43 ± 3.96	83.23 ± 4.24	76.73 ± 2.59
LSCCN	49.47 ± 1.20	54.04 ± 6.26	44.90 ± 6.74	43.22 ± 2.20
BLSTM	54.57 ± 2.38	55.10 ± 4.71	54.04 ± 4.82	52.18 ± 2.63
MuLHiTA	58.75 ± 2.63	61.35 ± 4.83	56.15 ± 5.28	56.12 ± 2.93
EEG Conformer	60.69 ± 4.52	50.91 ± 6.58	84.15 ± 4.56	59.14 ± 4.16
EEG-Deformer	68.99 ± 3.19	67.60 ± 6.26	70.38 ± 6.42	64.59 ± 4.13
CTNet	84.42 ± 2.56	81.73 ± 3.64	87.12 ± 3.51	83.88 ± 2.73
EEGMeNet	86.46 ± 3.52	82.45 ± 4.92	92.46 ± 3.41	85.25 ± 4.08
RST-GNN	**95.72 ± 5.33**	**96.06 ± 6.38**	**95.40 ± 8.04**	**95.64 ± 5.31**

Table 4
Comparison of computational complexity across different models in terms of trainable parameters (Params) and average inference time per sample. The input dimensions for each model are reported in the last column.

Methods	Params	Inference Time (ms)	Input
TSception	16,154	1.0	(1, 1, 24, 2000)
DeepConvNet	164,754	1.2	(1, 1, 24, 2000)
EEGNet	10,532	0.5	(1, 1, 24, 2000)
DGCNN	4,140	1.6	(1, 24, 20)
RGNN	6,704	2.1	(1, 24, 20)
RST-GNN	106,860	7.6	(1, 24, 2000)

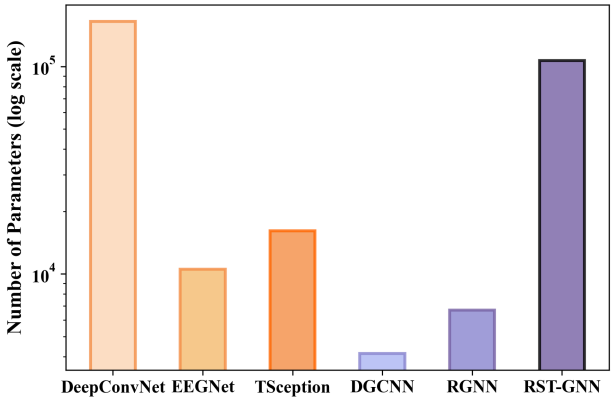


Figure 15: Parameter size comparison between the proposed and baseline models.

Under this experimental configuration, RST-GNN was retrained and evaluated on Dataset I. The comparison results with the reported methods are summarized in Table 3. As can be observed, RST-GNN achieves the best performance across all evaluation metrics, significantly outperforming the compared models. In particular, RST-GNN improves the classification accuracy by a large margin compared with the strongest baseline EEGMeNet, demonstrating the effectiveness of the proposed framework even under different preprocessing and evaluation settings.

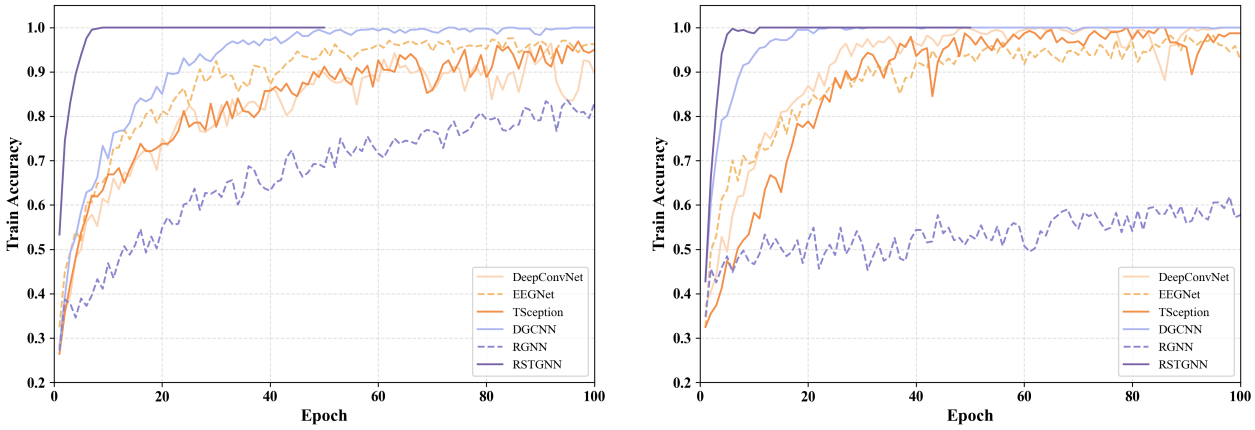
4.4.4. Efficiency and complexity analysis

To provide a comprehensive assessment of the proposed RST-GNN, computational efficiency is analyzed from three complementary perspectives: parameter scale, convergence behavior, and theoretical complexity. Together, these aspects demonstrate not only the feasibility of the model in practice but also its efficiency compared with baseline approaches.

Table 4 summarizes the computational characteristics of all compared models, including the number of trainable parameters (Params), the average inference time per sample, and the input tensor dimensions. The inference time represents the average forward-pass time per sample measured over multiple runs on the same hardware platform (NVIDIA GeForce RTX 3090 GPU), excluding preprocessing operations. The input dimensions are also reported since they influence the computational cost during inference. Notably, the relatively small input dimension used by DGCNN and RGNN is due to the use of differential entropy (DE) features as model inputs.

As shown in Table 4, RST-GNN contains a moderate number of trainable parameters compared with existing models. Specifically, it has fewer parameters than DeepConvNet but more than lightweight architectures such as EEGNet and DGCNN. For visual comparison, Fig. 15 further illustrates the parameter counts across models using a bar chart. In terms of inference efficiency, RST-GNN requires a longer inference time per sample than most baseline methods. Despite this increased inference cost, the training curves (see Fig. 16) clearly indicate that RST-GNN converges faster and more stably than the other models. This observation suggests that the additional parameters and structured operations introduced by RST-GNN contribute effectively to representation learning and optimization stability rather than causing unnecessary redundancy.

From the perspective of theoretical complexity, let C denote the number of EEG channels, T the number of time points, N_w the number of temporal windows with length L, d the embedding dimension after BiMap projection, and



(a) Dataset I (b) Dataset II

Figure 16: Accuracy convergence curves of the compared models on the Dataset I and Dataset II.

$|E|$ the number of graph edges. The overall per-sample complexity can then be approximated as:

$$\mathcal{O}\left(N_w C^2 L + N_w \left(dC^2 + d^2 C\right) + L_g |E| d^2 + N_w d^3\right). \tag{24}$$

which corresponds respectively to covariance estimation, BiMap projection, graph propagation with L_g layers, and Riemannian operations (e.g., RBN and Log). This formulation shows that the complexity grows linearly with the number of windows and quadratically or cubically with the embedding dimension. In practice, the use of sparse temporal graphs and moderate embedding sizes ensures that RST-GNN remains computationally tractable.

4.4.5. Limitations and future work

While the proposed RST-GNN framework demonstrates strong within-subject performance, several limitations remain that warrant future investigation.

First, the current evaluation is confined to within-subject settings, and the experiments on different datasets were conducted independently due to substantial differences in experimental paradigms and label definitions. This limits the model's generalizability across individuals or heterogeneous datasets. In real-world applications, inter-subject variability and differences in recording environments can significantly impact model robustness. Therefore, future work should explore cross-subject learning, cross-dataset generalization, and domain adaptation strategies to further evaluate and improve the generalizability of the proposed framework.

Second, the model's performance is partly dependent on the quality of covariance matrix estimation, which underpins the Riemannian representation learning. When the number of EEG channels is not substantially smaller than the number of time samples, the estimated covariance matrices may become ill-conditioned, potentially degrading downstream performance. Although spatial filtering helps by reducing dimensionality, overly aggressive dimensionality reduction can discard informative features, thus limiting classification accuracy. Future efforts could investigate more stable covariance estimation techniques, such as shrinkage estimators or adaptive channel selection, to strike a balance between stability and representational richness.

Third, the current study focuses solely on EEG data, without incorporating other physiological modalities such as electromyography (EMG) or electrooculography (EOG). These signals could provide complementary cues that enrich the understanding of cognitive load and enhance classification performance. Multimodal integration, therefore, represents a promising direction for future research, enabling more comprehensive and resilient cognitive load monitoring systems.

5. Conclusion

In this paper, we propose the Riemannian Spatio-Temporal Graph Neural Network (RST-GNN) framework for multi-class cognitive load detection using EEG signals. By integrating Riemannian manifold filtering with temporal graph learning, the proposed method effectively captures both the spatial and temporal dynamics of EEG signals, addressing the limitations of previous approaches. Our experiments on benchmark EEG datasets demonstrate that RST-GNN outperforms existing models in terms of classification performance, particularly in detecting multi-class cognitive load levels. The proposed framework provides a promising approach for cognitive load detection, with potential applications in areas such as driver attention monitoring, classroom performance assessment, and other human performance tracking scenarios. Future work could explore the extension of this framework to handle more complex multi-modal data, improve the efficiency of the model for deployment in real-world systems, and further investigate the interpretability of the learned features.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding

This work was supported by the following projects: Key Research and Development Program of Shaanxi (No. 2024GX-YBXM-039 and No. 2024GX-ZDCYL-02-15), Natural Science Basic Research Program of Shaanxi (No. 2025JC-JCQN-079 and No. 2025JC-YBQN-853), China Postdoctoral Science Foundation (No. 2025M771510), and the Ministry of Education of China (MOE) Project of Humanities and Social Sciences (No. 19YJC190028).

References

P. Evans, M. Vansteenkiste, P. Parker, A. Kingsford-Smith, S. Zhou, Cognitive load theory and its relationships with motivation: A self-determination theory perspective, Educational Psychology Review 36 (2024) 7.

M. Lagomarsino, M. Lorenzini, E. De Momi, A. Ajoudani, An online framework for cognitive load assessment in industrial tasks, Robotics and Computer-Integrated Manufacturing 78 (2022) 102380.

P. V. Amadori, T. Fischer, R. Wang, Y. Demiris, Predicting secondary task performance: A directly actionable metric for cognitive overload detection 14 (2021) 1474–1485.

S. G. Hart, L. E. Staveland, Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research, in: Advances in psychology, volume 52, Elsevier, 1988, pp. 139–183.

G. B. Reid, T. E. Nygren, The subjective workload assessment technique: A scaling procedure for measuring mental workload, in: Advances in psychology, volume 52, Elsevier, 1988, pp. 185–218.

D. Tao, H. Tan, H. Wang, X. Zhang, X. Qu, T. Zhang, A systematic review of physiological measures of mental workload, International journal of environmental research and public health 16 (2019) 2716.

Y. Zhou, P. Wang, P. Gong, F. Wei, X. Wen, X. Wu, D. Zhang, Cross-Subject Cognitive Workload Recognition Based on EEG and Deep Domain Adaptation 72 (2023) 1–12.

S. Ma, X. Yan, J. Billington, N. Merat, G. Markkula, Cognitive load during driving: EEG microstate metrics are sensitive to task difficulty and predict safety outcomes, Accident Analysis & Prevention 207 (2024) 107769.

A. Skulmowski, K. M. Xu, Understanding cognitive load in digital and online learning: A new perspective on extraneous cognitive load, Educational psychology review 34 (2022) 171–196.

T. Kosch, J. Karolus, J. Zagermann, H. Reiterer, A. Schmidt, P. W. Woźniak, A survey on measuring cognitive workload in human-computer interaction, ACM Computing Surveys 55 (2023) 1–39.

D. Klepl, F. He, M. Wu, D. J. Blackburn, P. Sarrigiannis, EEG-based graph neural network classification of Alzheimer's disease: An empirical evaluation of functional connectivity methods 30 (2022) 2651–2660.

R. Li, X. Yuan, M. Radfar, P. Marendy, W. Ni, T. J. O'Brien, P. M. Casillas-Espinosa, Graph signal processing, graph neural network and graph learning on biological data: a systematic review, IEEE Reviews in Biomedical Engineering 16 (2021) 109–135.

D. Klepl, F. He, M. Wu, D. J. Blackburn, P. Sarrigiannis, Adaptive gated graph convolutional network for explainable diagnosis of Alzheimer's disease using EEG data 31 (2023) 3978–3987.

A. Demir, T. Koike-Akino, Y. Wang, M. Haruna, D. Erdogmus, EEG-GNN: Graph neural networks for classification of electroencephalogram (EEG) signals, in: 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), IEEE, 2021, pp. 1061–1067.

T. Chen, Y. Guo, S. Hao, R. Hong, Exploring self-attention graph pooling with EEG-based topological structure and soft label for depression detection 13 (2022) 2106–2118.

J. Pan, R. Liang, Z. He, J. Li, Y. Liang, X. Zhou, Y. He, Y. Li, ST-SCGNN: a spatio-temporal self-constructing graph neural network for cross-subject EEG-based emotion recognition and consciousness detection 28 (2023) 777–788.

K. Shen, Q. She, X. Yang, Y. Gao, Y. Fan, Dynamic sparse directed graph convolutional network with attention mechanisms for eeg emotion recognition, Neurocomputing (2025) 131749.

Y. Li, K. Li, S. Wang, H. Wu, P. Li, A spatiotemporal separable graph convolutional network for oddball paradigm classification under different cognitive-load scenarios, Expert Systems with Applications 262 (2025) 125303.

Z. Yin, J. Tang, J. Zhang, Z. Wang, J. Zhang, Y. Wang, B. Zhang, Generic Mental Workload Measurement Using a Shared Spatial Map Network With Different EEG Channel Layouts 73 (2024) 1–13.

Y. Yan, L. Ma, Y.-S. Liu, K. Ivanov, J.-H. Wang, J. Xiong, A. Li, Y. He, L. Wang, Topological EEG-Based Functional Connectivity Analysis for Mental Workload State Recognition 72 (2023) 1–14.

C. H. Nguyen, P. Artemiadis, EEG feature descriptors and discriminant analysis under Riemannian Manifold perspective, Neurocomputing 275 (2018) 1871–1883.

F. P. Kalaganis, N. A. Laskaris, V. P. Oikonomou, S. Nikopolopoulos, I. Kompatsiaris, Revisiting Riemannian geometry-based EEG decoding through approximate joint diagonalization, Journal of Neural Engineering 19 (2022) 066030.

F. Lotte, L. Bougrain, A. Cichocki, M. Clerc, M. Congedo, A. Rakotomamonjy, F. Yger, A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update, Journal of neural engineering 15 (2018) 031005.

N. Cowan, The magical number 4 in short-term memory: A reconsideration of mental storage capacity, Behavioral and brain sciences 24 (2001) 87–114.

S. Dehaene, L. Charles, J.-R. King, S. Marti, Toward a computational theory of conscious processing, Current opinion in neurobiology 25 (2014) 76–84.

M. F. Hinss, E. S. Jahanpour, B. Somon, L. Pluchon, F. Dehais, R. N. Roy, Open multi-session and multi-task EEG cognitive Dataset for passive brain-computer Interface Applications, Scientific Data 10 (2023) 85.

Q. Wang, D. Smythe, J. Cao, Z. Hu, K. J. Proctor, A. P. Owens, Y. Zhao, Characterisation of cognitive load using machine learning classifiers of electroencephalogram data, Sensors 23 (2023) 8528.

R. T. Schirrmeister, J. T. Springenberg, L. D. J. Fiederer, M. Glasstetter, K. Eggensperger, M. Tangermann, F. Hutter, W. Burgard, T. Ball, Deep learning with convolutional neural networks for EEG decoding and visualization, Human brain mapping 38 (2017) 5391–5420.

V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, B. J. Lance, EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces, Journal of neural engineering 15 (2018) 056013.

Y. Ding, N. Robinson, S. Zhang, Q. Zeng, C. Guan, TSception: Capturing temporal dynamics and spatial asymmetry from EEG for emotion recognition 14 (2022) 2238–2250.

T. Song, W. Zheng, P. Song, Z. Cui, EEG emotion recognition using dynamical graph convolutional neural networks 11 (2018) 532–541.

P. Zhong, D. Wang, C. Miao, EEG-based emotion recognition using regularized graph neural networks 13 (2020) 1290–1301.

S. Kongwudhikunakorn, W. Ponwitayarat, S. Kiatthaveephong, W. Polpakdee, T. Yagi, V. Senanarong, P. Ittichaiwong, T. Wilaiprasitporn, Eegmenet: End-to-end multi-task neural network for brain-based mental workload classification, IEEE Internet of Things Journal (2025).