NexusNet: Lightweight Graph Modeling for Motor-Imagery-Based Brain–Computer Interfaces

Zikai Wang, Yuan Si, Zhenyu Wang, *Member, IEEE*, Ting Zhou, *Member, IEEE*,
Tianheng Xu, *Member, IEEE*, and Honglin Hu, *Senior Member, IEEE*

Abstract*—Motor-imagery-based brain–computer interfaces (MI-BCIs) hold significant potential for integration into Internet of Things (IoT) environments, enabling seamless interaction with users and external devices. In MI decoding, graph neural networks (GNNs) are increasingly employed due to their capacity to model the structures of nongrid data. However, current GNN approaches predominantly emphasize pairwise relationships, failing to account for the multinode interactions essential for accurate decoding. To overcome these limitations, we propose NexusNet, a lightweight graph modeling network tailored for MI-BCI and IoT integration. NexusNet incorporates multinode routing and Nexus Fusion, facilitating the modeling of intricate relationships that extend beyond pairwise connections. It effectively captures global information using a shortest path algorithm while complementing it with localized neighboring relationships. This architecture ensures efficient feature extraction and decoding while maintaining low computational overhead, making it suitable for resource-constrained scenarios. Experimental evaluations demonstrate that NexusNet achieves state-of-the-art performance, attaining 78.78% accuracy on BCIC-IV-2a, 87.21% on BCIC-IV-2b, and 94.12% on High Gamma Dataset with only 3.44K learnable parameters and 0.88M floating-point operations. These results highlight its potential for practical applications. Further analyses validate the contributions of its multinode components, with visualizations illustrating their indispensable roles in decoding. NexusNet represents a step forward in connecting MI-BCI research with practical IoT deployment. Our code is available at https://github.com/ZikaiVan/NexusNet.

***Index Terms*—Brain–computer interface (BCI), electroencephalogram (EEG), graph neural network (GNN), motor imagery (MI).**

I. INTRODUCTION

A BRAIN–COMPUTER interface (BCI) is a system that enables direct communication between the brain and external devices. By interpreting electroencephalogram (EEG) signals, BCIs can convert neural activity into commands to control assistive technologies like prosthetic limbs and wheelchairs, while also showing potential in Internet of Things (IoT) applications by enabling hands-free interaction with connected devices, such as smart home appliances [1], [2], [3], [4]. This capability is especially beneficial for individuals with mobility impairments, providing greater accessibility and convenience. Among EEG paradigms, motor imagery (MI) represents an active one within the BCI framework. It relies on the phenomenon where specific areas of the brain exhibit event-related synchronization (ERS) and event-related desynchronization (ERD) when individuals imagine performing a movement without action. This endogenous paradigm allows users to control IoT devices with greater independence, without the need for external stimuli.

To decode MI, various methodologies for feature extraction have been devised by researchers. In machine learning, Liu et al. [5] applied multiwavelet functions with regularized regression for MI classification, while Tang et al. [6] used mutual information for channel selection. Liu et al. [7] introduced a fine-grained spatial–frequency–time framework with multiscale segmentation. In deep learning, Shi et al. [8] integrated Riemannian geometry with neural networks to reduce complexity. Sartipi and Cetin [9] proposed a semi-supervised model combining a spatiotemporal autoencoder and classifier using limited labels. Liang et al. [10] designed EISATC-Fusion with inception, attention, and temporal convolution to enhance decoding and interpretability.

However, convolutional kernels, effective though, inherently rely on the prerequisites of data with grid structures, which limits their ability to study complex spatial dependencies from nongrid data, such as EEG. In this case, graph neural network (GNN) offers a significant advantage. By representing EEG channels as nodes and their relationships as edges in a graph, it naturally models the intricate connectivity patterns of the brain. Consequently, features from nongrid EEG can be further captured by GNN through its adaptable graph structure.

Common methods for graph construction involve static methods shown in Fig. 1(a) (e.g., neighbor-based [11], [12] and distance-based approaches [13], [14]) and dynamic methods depicted in Fig. 1(b) (e.g., functionality-based [15] approaches). In both ways, graphs are generated that only

---

Received 23 March 2025; revised 5 May 2025; accepted 20 May 2025. Date of publication 2 June 2025; date of current version 25 July 2025. This work was supported in part by the National Natural Science Foundation of China under Grant U24A20209, and in part by the Shanghai Municipal Commission of Economy and Information Project under Grant 2024-GZL-RGZN-01027. *(Corresponding author: Honglin Hu.)*

Zikai Wang and Yuan Si are with the Shanghai Advanced Research Institute, Chinese Academy of Sciences, Shanghai 201210, China, and also with the Shanghai Advanced Research Institute, University of Chinese Academy of Sciences, Beijing 101408, China (e-mail: wangzk@sari.ac.cn).

Zhenyu Wang and Honglin Hu are with the Shanghai Advanced Research Institute, Chinese Academy of Sciences, Shanghai 201210, China (e-mail: hlhu@ieee.org).

Ting Zhou is with the School of Microelectronics, Shanghai University, Shanghai 200444, China, and also with the Shanghai Advanced Research Institute, Chinese Academy of Sciences, Shanghai 201210, China.

Tianheng Xu is with the Shanghai Advanced Research Institute, Chinese Academy of Sciences, Shanghai 201210, China, and also with the Shanghai Frontier Innovation Research Institute, Shanghai 201108, China.

Digital Object Identifier 10.1109/JIOT.2025.3574180

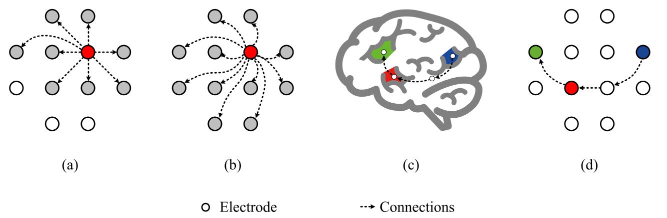


Fig. 1. (a) 8-neighbor graph, where each node is connected to its eight nearest neighbors. (b) Functionality graph, where each node is dynamically connected to other nodes. (c) Illustration of brain region cooperation relationships during a cognitive task. (d) Graph construction method to imitate collaborative work in (c) through routing, enabling information exchange across different electrodes.

represent the pairwise relationships. However, recent studies [16], [17] in neuroscience have revealed the collaborative nature of brain nodes during cognitive tasks. As shown in Fig. 1(c), the completion of a mental function requires cooperative processing of information among these nodes in the brain. Traditional graphs overlook the cooperation among electrodes, leading to the omission of critical routing relationships. This limitation challenges conventional GNNs, which treat the brain as a collection of discrete electrode pairs.

To address this issue, we introduce NexusNet, a novel method that aims to capture multinode cooperation relationships. NexusNet is a lightweight GNN comprising a Nexus fusion layer (NFL) and several convolution layers. During iterations, NFL aggregates Nexuses from various nodes within the graph. It learns the MI trials generated by the brain patterns shown in Fig. 1(b) and models these relationships on the graph, as illustrated in Fig. 1(c). Subsequently, NexusNet conducts feature extraction through convolutions and has EEG classified. Extensive experiments are conducted on two public datasets, providing detailed support for NexusNet.

The contributions of our work are listed as follows.

1) We propose a lightweight GNN, NexusNet, designed to capture complex relationships beyond pairwise connections. By incorporating Nexus Fusion, it effectively aggregates multinode information among electrodes. This lightweight design is particularly well suited for IoT applications, enabling efficient operation in resource-limited devices.
2) We conduct thorough experiments on two public datasets to validate NexusNet. Specifically, it achieves an average accuracy of 78.78% on the BCIC-IV-2a dataset and 87.21% on the BCIC-IV-2b dataset, with 3.44K learnable parameters and 0.88M floating-point operations (FLOP), demonstrating state-of-the-art performance.
3) We visualize Nexuses to quantitatively analyze the relationships reconstructed by NexusNet. Analysis reveals that for challenging subjects, the Route Nexus offers additional information for the decoding task, whereas in other cases, the Spatial Nexus remains the primary component in graph construction.

The remainder of this article is organized as follows. Section II reviews related work. Section III provides a comprehensive description of the NexusNet. Experiments and results are detailed in Section IV. Section V presents the visualizations and analyses of different Nexuses. Section VI offers in-depth discussions. Finally, conclusions are drawn in Section VII.

II. RELATED WORK

Inspired by the excellent performance in social network analysis [18], GNN and its applications have also been proposed for topology learning of EEG. GNN operates by propagating information across nodes based on the underlying graph structure, allowing each node to iteratively incorporate contextual cues from its neighbors. This relational inductive bias enables GNNs to infer unknown labels by leveraging the local topology and feature distribution.

Sun et al. [19] constructed an adaptive graph convolutional layer that dynamically integrated the electrode information to represent the EEG features. Wang et al. [11] employed a neighbor-based method to integrate feature-level graphs into convolutional layers. Ju and Guan [13] introduced a distance-based network using SPD manifold-valued graph convolutions for time–frequency EEG analysis. She et al. [20] introduced a discriminative adversarial network based on spatial–temporal-graph fusion to enhance the generalization capability of MI classification. Zhang et al. [21] developed a graph structure representing EEG nodes and applied convolutional recurrent attention to capture spatial and temporal features.

However, these neighbor-based and distance-based graphs primarily focus on local information, neglecting long-range dependencies. To address this limitation, a functionality-based graph approach is proposed, which captures global information and offers a holistic perspective of the entire graph. Demir et al. [15] introduced a flexible functionality graph that aggregates information across all electrode pairs. Vivek et al. [22] utilized a dynamic adjacency graph to learn the optimal connectivity within the brain. Cai et al. [23] built subgraphs by exploiting biological topologies among brain regions to capture both local and global relationships across specific channels.

These studies have successfully integrated EEG with electrodes graphs, utilizing adjacency matrices and message aggregation to achieve better results. However, according to recent neuroscience findings [16], [17], multinode cooperation is crucial to the cognitive tasks that emerges from interactions within cortical areas. Therefore, incorporating the interactions among electrodes can advance the understanding of cognitive differences in different individuals, thereby enhancing the performance of MI classification.

III. METHODOLOGY

A. Preliminary

To aid readers in better understanding NexusNet, several key concepts are briefly introduced first.

1) Graph: Graph is a kind of structure that models nongrid data, consisting of nodes and edges to denote entities and their interrelations, respectively. In GNN, the adjacency matrix is a crucial component for representing the relationships between node pairs. For undirected graphs, the adjacency matrix is symmetric, whereas for directed graphs, it is asymmetric. This

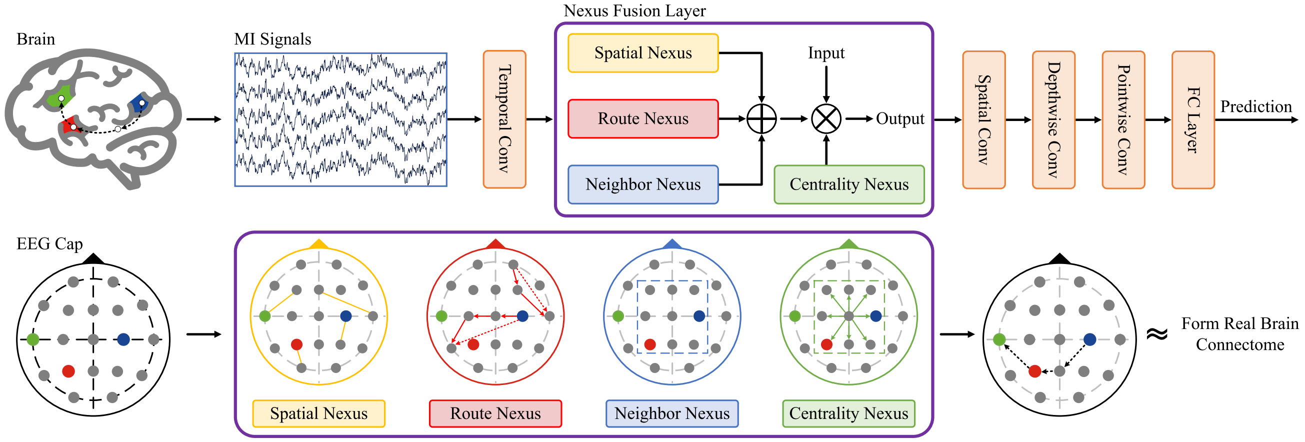


Fig. 2. Overall framework of the proposed NexusNet. The aim of the NFL is to model the cooperation in brain by aggregating Nexuses from different nodes.

matrix delineates the graph structure: an entry of 1 indicates the presence of an edge between two nodes, while 0 shows the absence of such a connection. Considering the nature of cooperation across electrodes, directed graphs is adopted in our work to model the relationships.

2) Neighbor: For a given node v_i, its K-nearest nodes are identified as its neighbors. In our work, we select the 8-nearest nodes (i.e., up-left, up, up-right, left, right, down-left, down, and down-right). This choice balances directional diversity with minimal redundancy among neighboring nodes compared to other K values.

B. Nexus Fusion Layer

Inspired by previous works [21], [23], we design an NFL to construct a graph with multinode information. In NFL, the Spatial Nexus captures pairwise information, serving as the foundation for the novel multinode routing graph, Route Nexus. Additionally, we develop the Neighbor Nexus to capture local information by modeling node adjacency and compute degree vectors to form the Centrality Nexus. Fig. 2 presents a conceptual illustration of the various Nexus structures, serving as a visual aid to convey our design principles. However, this schematic does not reflect the precise interregional relationships involved in MI signal generation.

1) Spatial Nexus: In graph construction, Euclidean distance is a common way to describe the spatial relationships between electrodes [23]. However, flexibility is limited due to the variations in the brain across subjects. Thus, a Euclidean distance-based embedding matrix, Spatial Nexus, is devised to study the topological relationships among nodes.

Specifically, considering a node pair (v_i, v_j), the Euclidean distance $d_{(i,j)}$ is first calculated. For every feasible distance $d_{(i,j)}$, a learnable scalar u is then assigned to model the spatial relationships between electrode pairs. Denoting the spatial relationship matrix as S, each element $S_{(i,j)}$ follows

$$S_{(i,j)} = u_{d_{(i,j)}}. \tag{1}$$

It is important to note that (1) serves as the basis for Spatial Nexus. The preliminary experiments (Section IV-H) show that the precise $d_{(i,j)}$ introduces redundant vectors, which impedes topological relationship construction and increases unnecessary computation. To address this issue, a tradeoff factor q is adopted to quantize the Euclidean distance as

$$d'_{(i,j)} = \left[\frac{d_{(i,j)}}{q} \right] \times q \tag{2}$$

where setting q to 5 enables the construction of an effective Spatial Nexus with an appropriate number of vectors.

Compared to other GNNs that focus solely on neighboring nodes, Spatial Nexus considers a broader range of electrodes. This approach allows the identification of key nodes influencing MI through the iterative updates of the embedding u.

2) Route Nexus: Recent studies highlight the significance of collaborative natures during cognitive tasks. Nonetheless, the common approach in existing GNNs, which focuses solely on electrode pairs, is inherently constrained. As far as our knowledge extends, little attention has been paid to incorporating routing relationships among brain regions. In response to this, we propose a new Route Nexus which aims to model collaborative relationships. The details are as follows.

For each neighboring node pair V, we establish a connection with an edge e_V, assigning it a weight of d'_V. For any node pair (v_i, v_j), the shortest path algorithm $\alpha(v_i, v_j)$ can be applied to determine a path $r_{(i,j)} = \{e_1, e_2, \ldots, e_M\}$ from v_i to v_j. In our work, the Floyd–Warshall algorithm [24] is adopted as α.

To facilitate the learning of collaborations between different nodes, a predefined label l_k as (3) is assigned to edge $e_k \in r_{(i,j)}$, which represents the connectivity during cognitive tasks in the brain

$$l_k = (T, B_1, B_2), \quad k \in [1, M] \tag{3}$$

where T indicates whether the edge e_k spans different brain regions, and B_1 and B_2 represent the brain region numbers at each endpoint of edge e_k. In our work, a value of 0 for T indicates that edge e_k remains within the same region, while a value of 1 signifies the opposite. The labels of brain regions vary depending on the dataset but follow a top-to-bottom numbering rule, which will be specifically discussed in Section IV-A.

TABLE I
NETWORK ARCHITECTURE AND PARAMETERS. HERE, k_1, k_2, AND k_3 ARE, RESPECTIVELY, SET AS 8, 16, AND 16. E REPRESENTS THE NUMBER OF ELECTRODES USED BY DIFFERENT DATASETS

Layer	In	Out	Kernel	Stride
Temporal Conv	1	k_1	(1, 64)	(1, 1)
Nexus Fusion	k_1	k_1	-	-
Spatial Conv	k_1	k_2	(E, 1)	(1, 1)
Average Pooling	k_2	k_2	(1, 8)	(1, 8)
Depthwise Conv	k_2	k_2	(1, 16)	(1, 1)
Pointwise Conv	k_2	k_3	(1, 1)	(1, 1)
Average Pooling	k_3	k_3	(1, 8)	(1, 8)

For each predefined label l, a learnable vector w is assigned as its encoder. Denote the adjacency matrix obtained by Route Nexus as R, we have

$$R_{(i,j)} = \frac{1}{M} \sum_{k=1}^{M} w_{l_k} (b_k)^T \tag{4}$$

where b_k is the kth learnable weight embedding.

In our proposed Route Nexus, several advantages can be discussed. First, conventional graphs are limited to pairwise interactions, constraining the receptive field to more electrodes. In contrast, the Route Nexus records features along the paths, which captures the information throughout the route. Second, with labels l_k, all regions in cooperation can be represented in the Route Nexus, thus allowing the network to acquire long-range dependencies.

3) Neighbor Nexus: In previous Nexuses, we primarily focus on multinode information at a global level. However, recognizing that proximal regions may exert greater influence on each other, we have augmented the network with additional neighbor relations.

Denoting the adjacency matrix constructed by Neighbor Nexus as N, we have

$$N_{(i,j)} = \begin{cases} 1, \text{if } \left(v_i, v_j\right) \text{ neighbor OR } i = j \\ 0, \text{otherwise.} \end{cases} \tag{5}$$

To balance the dissemination of information between nodes and mitigate the undue emphasis on certain nodes within the network, symmetric normalization is performed on N

$$\hat{N} = D^{(-1/2)} \cdot N \cdot D^{(-1/2)} \tag{6}$$

where $D = \operatorname{diag}(\sum_j N_{(1,j)}, \sum_j N_{(2,j)}, \ldots, \sum_j N_{(|E|,j)})$, $|E|$ is the number of electrodes.

4) Centrality Nexus: Centrality is a metric assessing the importance of different electrodes. In Centrality Nexus, we encode centrality using the degree of nodes, represented by $D_{(i,i)}$, which are the diagonal elements of the matrix D. For each possible degree, we embed it to $\mathbb{R}$, resulting in a learnable vector C that represents the centrality of each node as

$$C_i = D_{(i,i)} \to \mathbb{R}. \tag{7}$$

C. Overall Framework

The backbone and parameters of NexusNet are detailed in Table I, with its corresponding algorithm summarized in Algorithm 1. Inspired by [25], we have designated the number of kernels in the distinct convolutional layers as follows: $k_1 = 8$, $k_2 = 16$, and $k_3 = 16$. For Spatial Conv Layer, the size of the kernel is determined by the number of channels E of the MI trials.

Algorithm 1 Application Procedure of NexusNet

```
Input: MI trials (split into Training set $X_{tr}$, Validation set $X_{va}$,
    and Testing set $X_{te}$).
Output: Well-trained model $M$.
Input: All kernels $\theta_f$, $S$ (1), $R$ (4), $N$ (6), and $C$ (7).
while parameters have not converged do
    Obtain training trials $\{(x^i, y^i)\} \in X_{tr}$.
    Acquire prediction from NexusNet
    Calculate loss $\mathcal{L}$ using (12).
    Update parameters using backpropagation:
        Optimize kernels $\theta_f$ with $\mathcal{L}$.
        Update learnable Nexus $S$, $R$, and $C$.
        Update the learning rate using cosine annealing.
    Evaluate model $M$ with $\{(x^i, y^i)\} \in X_{va}$.
    Save model $M$.
end while
Return:Well-trained model $M$.
Testing:
    Acquire testing trials $\{(x^i, y^i)\} \in X_{te}$.
    Decode MI trials using model $M$.
```

The input trials first pass through a temporal convolutional layer, extracting a richer set of features without time-scale transformation. The NFL then encodes the connectivity of brain regions and merges it with the temporal features f_i.

For f_i, NexusNet first sums the Spatial Nexus, Route Nexus, and Neighbor Nexus, and then multiplies the result with f_i. The Centrality Nexus is broadcast to a matrix of the same shape as f_i and then undergoes Hadamard multiplication

$$f_o = \operatorname{Broadcast}(C) \odot \left[\left(S + R + \hat{N} \right) f_i \right] \tag{8}$$

where S, R, and N have dimensions of $E \times E$, C has a dimension of $E \times 1$, f_i and f_o have dimensions of $E \times T$, E is the number of electrodes used in MI experiments, and T is the number of sample points.

The combined features f_o are then processed sequentially through spatial convolution, depthwise convolution, and separable convolution layers to achieve a high-dimensional abstraction. Finally, a fully connected layer facilitates the classification process, ensuring accurate and efficient categorization.

IV. EXPERIMENTS AND RESULTS

A. Datasets and Preprocessing

Extensive evaluations of the NexusNet were conducted based on three public datasets: 1) BCI Competition IV Dataset 2a [26]; 2) BCI Competition IV Dataset 2b [27]; and 3) High Gamma Dataset [28]. The electrode distributions are illustrated in Fig. 3.

1) *Dataset I: BCI Competition IV Dataset 2a* contains MI EEG data from nine subjects (A1–A9) recorded with 22 electrodes (plus 3 EOG) over two sessions, each

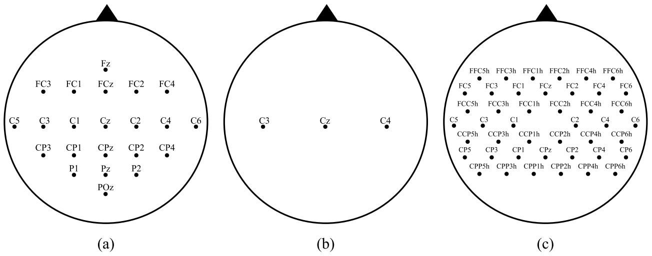


Fig. 3. Electrode distributions in (a) Dataset I, (b) Dataset II, and (c) Dataset III.

comprising 288 trials of four MI tasks (left hand, right hand, feet, and tongue). Signals were sampled at 250 Hz, and data between 2 and 6 s were used for analysis [29]. Six brain regions (Frontal to Parietal-Occipital) were indexed from 0 to 5 for use in (3).

2) *Dataset II: BCI Competition IV Dataset 2b* includes EEG from nine subjects (B1–B9) recorded with three electrodes over five sessions (120 trials/session) during two MI tasks (left versus right hand). Signals were sampled at 250 Hz, and data from 3 to 7 s were retained [29]. All electrodes were located in the Central region, indexed as 0 in (3).
3) *Dataset III: High Gamma Dataset* comprises EEG data from 14 subjects performing four MI tasks (left hand, right hand, feet, and rest), recorded with 128 electrodes at 500 Hz, then resampled to 250 Hz with a 0.5-Hz high-pass filter. Trials were clipped from 0 to 4 s, and 44 relevant channels were selected per protocol [28]. Seven brain regions were indexed from 0 to 6 for (3).

For all collected trials, we first detrend the signals to remove any linear trends, followed by applying a 50-Hz notch filter to suppress powerline interference. Next, a 413-order bandpass FIR filter with a frequency range of [2,40] Hz is designed to eliminate both low- and high-frequency noise in the EEG signals. After filtering, we apply exponential moving standardization to mitigate fluctuations, defined as

$$x_t' = \frac{x_t - \mu_t}{\delta_t} \tag{9}$$

where x_t' is the output and x_t is the bandpass-filtered signal at time t. The mean μ_t and standard deviation δ_t are computed using an attenuation rate $\alpha = 0.999$ as

$$\mu_t = (1-\alpha)x_t + \alpha\mu_{t-1} \tag{10}$$

$$\delta_t^2 = (1-\alpha)(x_t - \mu_t)^2 + \alpha\delta_{t-1}^2. \tag{11}$$

B. Experiments Setup

In accordance with the competition guidelines [29], a hold-out validation protocol was adopted. For Dataset I, the first session was partitioned such that the first 80% of trials were allocated to the training set, while the remaining 20% within the same session served as the validation set. The second session was reserved for testing. For Dataset II, the first three sessions were designated for training and validation, with 80% of trials within each session assigned to training and the final 20% of each session held out for validation. Testing was performed on the final two independent sessions. For Dataset III, 80% of the 880 presplit trials per subject were used for training, 20% for validation, and the reserved 160 trials were reserved for testing.

The experiments were conducted with an RTX 2080 Ti GPU based on PyTorch 1.10.0 in Python 3.8. With respect to the hyperparameters, Adam optimizer was adopted to minimize the cross-entropy loss following (12), the learning rate of the optimizer was set to 0.005, the strategy for learning rate optimization was cosine annealing following (13), batch size was 72, and epoch was set to 2000. The early-stop strategy [30] was adopted to avoid overfitting as much as possible, with a threshold of 500.

Here, cross-entropy loss $\mathcal{L}$ was defined as

$$\mathcal{L} = -\frac{1}{N_b}\sum_{i=0}^{N_b}\sum_{c=1}^{N_c} y \log \hat{y} \tag{12}$$

where N_b was the batch size, N_c represented the number of MI categories, and y and $\hat{y}$ were the ground truth and the prediction, respectively.

The cosine annealing strategy adjusts the learning rate τ_t over training epochs as

$$\tau_t = \tau_{\min} + \frac{1}{2}(\tau_{\max} - \tau_{\min})\left(1 + \cos\left(\frac{T_{\text{cur}}}{T_{\max}}\pi\right)\right) \tag{13}$$

where $\tau_{\min}$ and $\tau_{\max}$ denote the minimum and maximum learning rates, $T_{\max}$ is the total number of epochs, and T_{cur} is the current epoch. Initially, $\tau_t = \tau_{\max}$ and gradually decreases toward $\tau_{\min}$ following a cosine curve. This smooth, non-monotonic schedule facilitates better convergence and weight refinement throughout training.

To comprehensively demonstrate the performance of NexusNet, several state-of-the-art methods based on a variety of architectures were employed for comparison, with their description coming as follows.

1) *FBCSP* [29] is a machine-learning method that applies a set of band-pass filters to EEG data to isolate different frequency bands. It then extracts spatial patterns that are consistent across these bands.
2) *EEGNet* [25] is a compact convolutional neural network for EEG analysis using deep convolution and separable convolution.
3) *ConvNet* [28] is a convolutional neural network that performs both temporal and spatial convolutions on EEG data for feature extraction.
4) *FBCNet* [31] introduces a novel Variance Layer that captures temporal information from EEG data. By leveraging a multiview data representation, it can extract spectro-spatial features that are discriminative for MI classification.
5) *G-CRAM* [21] is a graph-based model which constructs a graph to represent electrode positions and leverages a recurrent attention mechanism to capture features.
6) *GSAN* [32] constructs a brain graph based on the spatial distribution of electrodes and their functional connections. By aligning data distributions within the same class domain, the network improves classification performance for target domains.

TABLE II
PERFORMANCE COMPARISON BETWEEN DIFFERENT STATE-OF-THE-ART METHODS ON DATASET I WITH WILCOXON SIGNED-RANK TEST ($p < 0.05$:*, $p < 0.005$:**). *Avg* DENOTES AVERAGE ACCURACY AND *Std* REPRESENTS THE SAMPLE STANDARD DEVIATION. *Std* IS CALCULATED BASED ON THE ACCURACY ACROSS NINE SUBJECTS TO INDICATE ROBUSTNESS

Dataset	Method	A1	A2	A3	A4	A5	A6	A7	A8	A9	Avg	Std	F1	Kappa
I	FBCSP	76.00	56.50	81.25	61.00	55.00	45.25	82.75	81.25	70.75	67.75**	13.73	0.670	0.570
	EEGNet	86.11	61.81	91.32	62.50	68.06	59.03	88.19	82.99	88.54	76.51	12.58	0.751	0.687
	ConvNet	76.48	56.72	85.68	71.75	75.17	53.44	75.08	77.85	80.35	72.50*	10.64	0.715	0.633
	FBCNet	83.95	58.94	89.84	76.41	74.57	53.87	84.77	80.41	80.60	75.93	12.02	0.747	0.679
	G-CRAM	82.98	63.54	95.11	70.42	69.02	62.19	84.68	84.24	77.96	76.68	11.08	0.755	0.689
	GSAN	86.78	59.53	90.06	81.31	56.50	50.71	93.93	87.91	87.29	77.11	16.63	0.763	0.695
	GCNs-Net	85.27	64.03	95.16	65.19	75.87	61.60	88.20	83.53	77.55	77.38	11.11	0.751	0.698
	MSVTNet	84.26	61.83	89.63	70.12	70.26	56.23	88.36	85.29	82.08	76.45	11.49	0.726	0.686
	Conformer	85.54	60.54	93.43	78.67	50.61	64.71	90.02	86.28	89.23	77.67	15.27	0.771	0.702
	Ours	82.29	60.42	96.53	71.18	70.49	67.71	89.24	85.42	85.76	78.78	11.81	0.787	0.717

TABLE III
PERFORMANCE COMPARISON BETWEEN DIFFERENT STATE-OF-THE-ART METHODS ON DATASET II WITH WILCOXON SIGNED-RANK TEST ($p < 0.01$:*, $p < 0.005$:**). *Avg* DENOTES AVERAGE ACCURACY AND *Std* REPRESENTS THE SAMPLE STANDARD DEVIATION. *Std* IS CALCULATED BASED ON THE ACCURACY ACROSS NINE SUBJECTS TO INDICATE ROBUSTNESS

Dataset	Method	B1	B2	B3	B4	B5	B6	B7	B8	B9	Avg	Std	F1	Kappa
II	FBCSP	70.00	60.36	60.94	97.50	93.12	80.63	78.13	92.50	86.88	80.01*	13.85	0.798	0.600
	EEGNet	75.94	57.64	58.43	98.13	81.25	88.75	84.06	93.44	89.69	80.81	13.64	0.802	0.612
	ConvNet	75.55	49.15	52.06	95.29	92.97	83.91	84.73	90.43	85.07	78.80**	17.00	0.785	0.576
	GCNs-Net	75.50	56.77	57.91	98.79	80.44	89.59	83.94	93.27	89.20	80.60	14.01	0.806	0.612
	MSVTNet	81.15	62.62	63.65	96.19	92.81	88.18	85.32	95.34	89.40	83.85	11.93	0.811	0.677
	Conformer	80.83	65.34	63.75	97.61	85.89	91.29	88.05	94.11	91.31	84.26	12.12	0.839	0.685
	Ours	79.38	66.79	83.44	97.19	95.00	88.44	92.81	93.44	88.44	87.21	9.53	0.875	0.744

TABLE IV
PERFORMANCE COMPARISON BETWEEN DIFFERENT STATE-OF-THE-ART METHODS ON DATASET III. *Avg* DENOTES AVERAGE ACCURACY AND *Std* REPRESENTS THE SAMPLE STANDARD DEVIATION. *Std* IS CALCULATED BASED ON THE ACCURACY ACROSS 14 SUBJECTS TO INDICATE ROBUSTNESS

Dataset	Method	Avg	Std	F1	Kappa
III	EEGNet	92.24	5.57	0.917	0.897
	GCNs-Net	93.88	4.20	0.932	0.918
	MSVTNet	93.33	4.75	0.928	0.911
	Ours	94.12	2.27	0.935	0.922

7) *GCNs-Net* [33] is a graph-based deep learning framework leveraging functional topological relationships among electrodes for MI classification.
8) *MSVTNet* [34] integrates multiscale CNNs and Transformers to capture spatiotemporal and cross-frequency features for robust MI decoding.
9) *Conformer* [35] is a method that integrates both convolutional layers and Transformer. This integration enables Conformer to capture both local and global features within a unified EEG classification framework.

C. Baseline Comparison

The comparative results recorded in Tables II–IV showed that NexusNet outperformed the baselines in terms of average accuracy on two datasets. To ensure a comprehensive evaluation, Accuracy (14), F1-score (15), and Kappa value (16) were selected as the evaluation metrics, as defined in the following equations:

$$\text{Acc} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{14}$$

$$\text{F1} = \frac{2 \cdot \text{TP}}{2 \cdot \text{TP} + \text{FP} + \text{FN}} \tag{15}$$

where TP, TN, FP, and FN denote the number of true positive, true negative, false positive, and false negative samples, respectively

$$\text{Kappa} = \frac{p_o - p_e}{1 - p_e} \tag{16}$$

where p_o represents the observed accuracy and p_e is the expected probability of chance agreement.

Specifically, on Dataset I, where 22 electrodes are employed.

1) The average accuracy of all deep learning methods outperforms that of the machine learning method FBCSP by over 5%–10%, revealing the greater efficacy of deep learning in feature extraction.
2) Among all CNNs, NexusNet demonstrates a significant performance advantage, surpassing others by 3%–7% in accuracy and 0.03–0.06 in F1 score. This not only confirms the importance of collaboration across different nodes in cognitive tasks but also emphasizes the necessity of incorporating such relationships into EEG classification tasks.
3) For GNNs (G-CRAM, GSAN, GCNs-Net, and MSVTNet), NexusNet employs a distinct graph that differs from all. By incorporating multinode information, NexusNet addresses the limitation of GNNs that focus solely on pairwise connections. This novelty enables NexusNet to capture structural relationships comprehensively, resulting in its improved performance.
4) Table II further underscores the supportive role of the self-attention mechanism in Transformers for MI decoding. Still, limitations persist. Whereas MI-CAT and

Conformer demonstrate remarkable accuracy, their standard deviations are higher than those of other methods, indicating the need for improved robustness.

On Dataset II, where only three electrodes are involved, several conclusions can be drawn.

1) Deep learning methods generally outperform FBCSP in accuracy, though. The improvements are limited, which suggests that for low-density data, machine learning could be a better choice.
2) Compared to Conformer, NexusNet shows no significant advantages. This indicates that a sufficient number of electrodes are essential for constructing brain connectome, which should be considered when an experimental paradigm is designed.
3) NexusNet maintains a similar standard deviation level on nine subjects as observed in Dataset I. In contrast to ConvNet, whose performance variability is evident, NexusNet demonstrates enhanced capacity for generalization.
4) Similar performance trends are observed in NexusNet and Conformer across both datasets, whereas the remained methods show poorer performance in high-density scenarios. This consistency highlights the adaptability of NexusNet and Conformer across varying data scales.

On Dataset III, which includes 44 EEG channels, our proposed NexusNet demonstrates consistently strong and balanced performance across all evaluation metrics. When compared with the results on Dataset I (22 channels) and Dataset II (3 channels), NexusNet maintains competitive accuracy and robustness, indicating its strong adaptability to varying channel configurations. This performance can be attributed to NexusNet's ability to model complex inter-channel dependencies through multinode relational learning. By effectively capturing both spatial and functional connectivity patterns, particularly in high-density setups, such as Dataset III and Dataset I, NexusNet exhibits superior generalization capability and scalability across diverse EEG classification scenarios.

D. Profile Analysis

The profile analysis of NexusNet and several baselines was conducted on a laptop with an 8th Gen Intel Core i5 CPU @ 1.6 GHz (denoted as Laptop) and an Android device with a Qualcomm Snapdragon 855 @ 2.84 GHz (denoted as Mobile). The corresponding results were summarized in Table V. For hyperparameters, the batch size was set to 1, with all other parameters matching the optimal values reported in the respective works. In brief, the results offered empirical evidence substantiating the lightweight characteristics of NexusNet, including the number of parameters (Params), FLOPs, inference latency (Latency), and RAM usages (RAM).

As detailed in Table V, despite achieving comparable classification accuracy to Conformer, NexusNet operates with 99.6% fewer parameters (3.44K versus 789.82K) and 98.6% lower FLOPs (0.88M versus 63.86M). This efficiency extends to real-time deployment: NexusNet reduces inference latency by 83.5% on Laptop (12.09 ms versus 73.25 ms) and 83.6% on Mobile (7.85 ms versus 47.87 ms) compared to Conformer, while concurrently lowering RAM consumption by 69.3% (2.20 MB versus 7.17 MB) and 80.2% (1.12 MB versus 5.66 MB) in respective environments.

TABLE V

COMPARISON OF PROFILE ANALYSIS ON DATASET I, WHERE *PARAMS* DENOTES THE NUMBER OF LEARNABLE PARAMETERS, AND *FLOPS* REPRESENTS THE NUMBER OF FLOATING-POINT OPERATIONS. LATENCY AND RAM CONSUMPTION ARE MEASURED IN BOTH LAPTOP AND MOBILE ENVIRONMENTS, WITH DETAILED SPECIFICATIONS PROVIDED IN THE TEXT

Model	Params (K)	FLOPs (M)	Latency (ms)		RAM (MB)	
			Laptop	Mobile	Laptop	Mobile
FBCNet	11.81	1.30	22.89	13.77	5.10	2.24
G-CRAM	420.36	52.32	147.42	112.21	**1.36**	**1.06**
Conformer	789.82	63.86	73.25	47.87	7.17	5.66
Ours	**3.44**	**0.88**	**12.09**	**7.85**	2.20	1.12

For Conformer, NexusNet achieves comparable results using minimal parameters and FLOPs, with greater stability observed across all subjects. These results underscore the importance of modeling node relationships within networks and validate the efficacy of NexusNet.

E. Confusion Matrix

Despite excellent performance, the precision and sensitivity of NexusNet remained unclear. To investigate the issues of misclassification and imbalance, confusion matrix analyses were conducted for subjects with the superior and poor average accuracy from each dataset, as well as for the overall dataset. The results were shown in Fig. 4.

For subjects with the poor average accuracy (A2, B2), the confusion matrices reveal that NexusNet maintains relatively balanced false positive and false negative rates. In contrast, the matrices for subjects with the high average accuracy (A3, B5) demonstrate that the network is capable of achieving high true positive rates, even in the four-class tasks.

The average results corroborate these observations, showing that NexusNet maintains precision and sensitivity above 75% in the four-class tasks of Dataset I and above 87% in the two-class tasks of Dataset II. These findings indicate that NexusNet has no concentration of misclassifications in any particular category, suggesting a consistent discriminatory performance across tasks and the balanced nature of NexusNet.

F. Ablation Study

To comprehensively examine the importance of different Nexuses in modeling brain connectivity, a series of ablation experiments were carried out on both Dataset I and Dataset II. The results, as presented in Tables VI and VII, clearly showed that NexusNet consistently outperformed its degraded models. Further, the following conclusions can be drawn.

1) Exclusive reliance on traditional adjacency-based relationships (NN alone) proves insufficient, as evidenced by the poorest performing models (No. 2 and No. 6 in Dataset I: 52.48% and 52.62%; Dataset II: 79.14% and 79.34%). These configurations suffer from more

TABLE VI

Performance Comparison Between Different Degraded NexusNet on Dataset I With Wilcoxon Signed-Rank Test ($p < 0.005$:*). *Avg* Denotes Average Accuracy. *w/o* Stands for Without. *SN* Represents Spatial Nexus, *RN* Represents Route Nexus, *NN* Represents Neighbor Nexus, and *CN* Represents Centrality Nexus. Optimal Results Are Highlighted in Bold

No.	Design	A1	A2	A3	A4	A5	A6	A7	A8	A9	Avg
1	w/o SN	74.43	54.14	90.72	54.22	63.98	62.04	80.26	76.05	75.25	70.12*
2	w/o SN, RN	53.87	36.86	68.27	29.12	52.20	50.24	60.34	61.47	59.98	52.48*
3	w/o SN, NN	77.99	49.16	87.60	59.57	63.33	66.29	87.97	78.83	74.56	71.70*
4	w/o SN, CN	78.94	59.87	91.46	63.89	57.49	59.96	82.58	82.30	76.64	72.57*
5	w/o SN, RN, NN	78.50	56.28	85.96	60.22	61.09	63.43	86.34	82.30	75.95	72.23*
6	w/o SN, RN, CN	67.90	41.84	70.31	29.80	51.54	44.68	54.38	55.91	57.20	52.62*
7	w/o SN, NN, CN	76.89	60.38	87.52	62.95	68.69	63.43	80.32	80.91	72.48	72.62*
8	w/o SN, RN, NN, CN	79.83	53.20	94.79	60.82	57.68	55.10	86.84	80.91	78.73	71.99*
9	w/o RN	73.85	60.38	91.21	59.18	65.53	66.21	86.13	78.13	76.64	73.03*
10	w/o RN, NN	68.06	54.69	89.65	59.08	61.79	56.49	82.38	76.74	78.03	69.66*
11	w/o RN, CN	75.14	55.55	91.19	54.92	69.08	62.74	88.20	73.97	72.48	71.47*
12	w/o RN, NN, CN	74.71	53.50	93.63	55.37	64.83	64.13	78.04	78.13	67.61	70.00*
13	w/o NN	79.76	59.32	91.72	65.85	64.62	65.17	82.00	82.65	82.20	74.81*
14	w/o NN, CN	77.02	58.26	92.63	61.12	60.40	66.56	87.19	84.73	75.60	73.72*
15	w/o CN	80.02	54.36	94.34	70.61	68.40	58.57	84.76	82.99	75.95	74.44*
16	NexusNet	**82.29**	**60.42**	**96.53**	**71.18**	**70.49**	**67.71**	**89.24**	**85.42**	**85.76**	**78.78**

TABLE VII

Performance Comparison Between Different Degraded NexusNet on Dataset II With Wilcoxon Signed-Rank Test ($p < 0.005$:*). *Avg* Denotes Average Accuracy. *w/o* Stands for Without. *SN* Represents Spatial Nexus, *RN* Represents Route Nexus, *NN* Represents Neighbor Nexus, and *CN* Represents Centrality Nexus. Optimal Results Are Highlighted in Bold

No.	Design	A1	A2	A3	A4	A5	A6	A7	A8	A9	Avg
1	w/o SN	74.63	64.71	81.37	96.25	87.19	81.75	90.75	91.13	77.81	82.84*
2	w/o SN, RN	62.13	60.79	77.31	90.94	87.50	78.94	86.38	86.13	82.19	79.14*
3	w/o SN, NN	76.94	66.50	74.19	96.25	93.13	82.69	86.06	90.81	77.19	82.64*
4	w/o SN, CN	78.69	63.64	80.44	96.25	85.94	80.19	91.69	91.06	79.38	83.03*
5	w/o SN, RN, NN	72.06	59.14	74.25	88.63	85.44	83.00	83.38	91.25	75.06	79.13*
6	w/o SN, RN, CN	71.88	60.79	76.56	90.25	82.19	82.37	86.06	82.37	81.56	79.34*
7	w/o SN, NN, CN	75.88	64.36	77.63	96.25	92.19	83.63	91.06	90.19	87.19	84.26*
8	w/o SN, RN, NN, CN	76.50	64.36	78.56	93.63	87.81	79.88	87.31	87.69	85.00	82.30*
9	w/o RN	74.63	63.29	72.63	95.63	90.94	79.25	91.25	89.88	78.44	81.77*
10	w/o RN, NN	74.31	59.00	78.88	94.69	87.81	81.13	89.50	85.50	87.19	82.00*
11	w/o RN, CN	74.63	60.79	77.94	95.63	93.13	83.94	90.75	84.88	85.94	83.07*
12	w/o RN, NN, CN	72.44	61.86	77.94	94.69	89.06	80.19	91.06	88.00	83.13	82.04*
13	w/o NN	76.81	63.21	78.25	95.63	93.13	86.13	88.56	84.56	85.31	83.51*
14	w/o NN, CN	77.12	62.21	78.88	96.25	90.94	84.56	87.75	90.19	82.81	83.41*
15	w/o CN	76.19	65.43	74.50	95.00	93.75	82.69	91.69	92.38	84.06	83.96*
16	NexusNet	**79.38**	**66.79**	**83.44**	**97.19**	**95.00**	**88.44**	**92.81**	**93.44**	**88.44**	**87.21**

than 25% accuracy degradation compared to NexusNet in Dataset I ($p<0.005$), highlighting the limitations of static neighborhood graphs in modeling complex neurodynamic interactions.

2) The interchangeable impact of SN and RN removal reveals latent redundancy. In Dataset I, models excluding SN (No. 1: 70.12%) or RN (No. 9: 73.03%) show comparable degradation, while their simultaneous removal (No. 2: 52.48%) amplifies performance loss. This pattern persists in Dataset II (No. 1: 82.84% versus No. 9: 81.77%), implying overlapping contributions to connectivity modeling. The nonadditive nature of their individual removals suggests shared capacity in encoding spatial dependencies, warranting future investigation into disentangling their unique representational subspaces.

3) These ablation studies validate NexusNet's efficacy through three synergistic principles: a) multiscale Nexuses mitigate single-relation graph limitations; b) functional redundancy (e.g., SN/RN) enhances noise robustness; and c) hybrid learned (SN/RN) and prior-driven (NN/CN) relations balance adaptability with neuroanatomical plausibility. The accuracy gaps between NexusNet and variants across datasets underscore the necessity of holistic integration over isolated optimizations.

G. *Duration Sensitivity*

To assess the impact of NFL with varying MI durations, an experiment was designed to compare the classification performance of NexusNet with or without NFL, as recorded in Fig. 5.

For durations up to 2 s, minimal differences in average performance are observed. However, beyond this threshold, the network with NFL demonstrates a progressively superior performance compared to its counterpart. This enhanced performance of the NFL-equipped network is attributed to the embedding of different electrodes. As the duration increases, NFL is better able to adjust the cooperative relationships between different channels, which highlights the effectiveness of NFL for extended MI trials.

H. *Tradeoff Factor Analysis*

The results from the preliminary experiments detailed in Section III-B1 are presented in Fig. 6, illustrating the impact

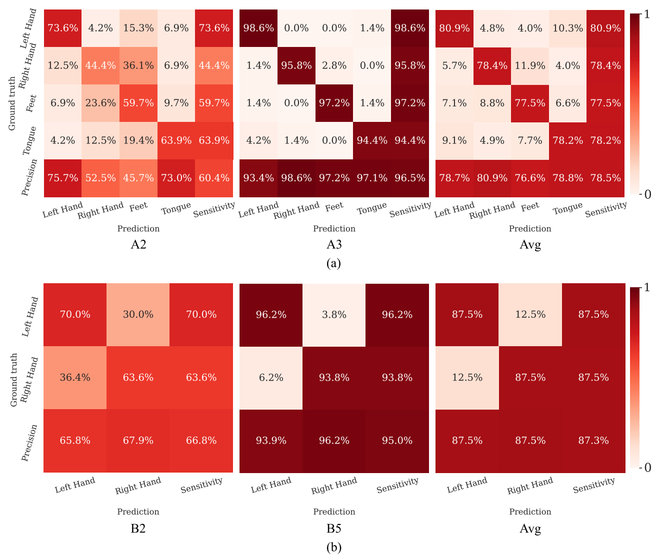


Fig. 4. Confusion matrices of NexusNet, where (a) shows the results for A2, A3, and the overall average in Dataset I, and (b) shows the results for B2, B5, and the overall average in Dataset II.

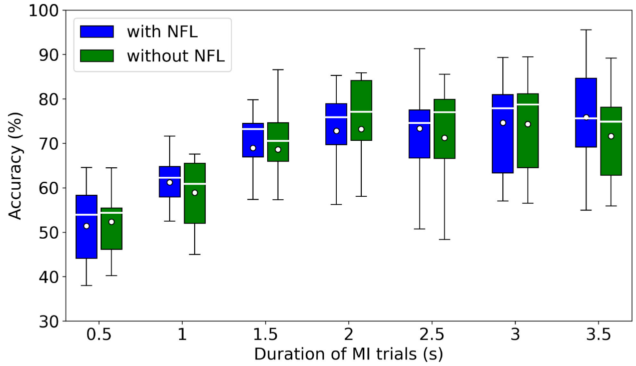


Fig. 5. Classification performance comparison across different MI trial durations, with or without NFL. The lines represent the median, and the dots indicate the average accuracy. The box plot illustrates the effect of NFL in enhancing network performance for longer duration MI trials.

of the tradeoff factor q on classification performance. When q is set to 5, NexusNet mostly demonstrates a significant advantage over other methods, and omitting q leads to the poorest performance. This finding supports the conclusion that the quantified distances effectively integrate similar spatial information, allowing the network to focus on areas with larger disparities. In addition, the influence of q shows nonlinearity, indicating that during applications, q must be carefully adjusted based on different electrode montages to achieve optimal decoding performance.

V. VISUALIZATION

A. Explanation of Nexus

To improve the interpretability of NexusNet, Neighbor Nexus, Route Nexus, Spatial Nexus, and the total Nexus (i.e., sum of the three) from four subjects in Dataset I were extracted and visualized from the trained models. The corresponding results were depicted in Fig. 7. Cosine similarity was adopted to support the data analysis, which could be summarized as

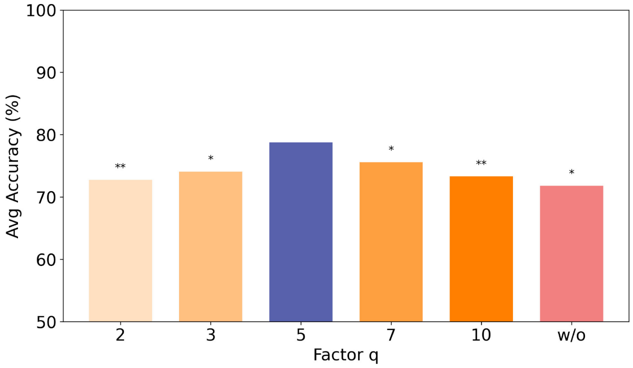


Fig. 6. Average accuracy of NexusNet under different tradeoff factors q, with Wilcoxon signed-rank test ($p < 0.05$:*, $p < 0.01$:**). *w/o* stands for without.

$$\text{Similarity} = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} \tag{17}$$

where $\mathbf{A}$ and $\mathbf{B}$ were two vectors, and $\|\mathbf{A}\|$ and $\|\mathbf{B}\|$ denoted the respective norms.

A comparison across four subjects reveals that the Route Nexus derived from A2 (Fig. 7 A2.Route) lacks interchannel connectivity, suggesting the network failed to reconstruct meaningful inter-regional relationships. In contrast, NexusNet constructs distinct and coherent Route Nexus structures for A1 and A3. Notably, subjects with higher classification accuracy tend to have a Total Nexus more aligned with the Spatial Nexus than the Route Nexus, implying that MI tasks are more strongly linked to spatial patterns, while the Route Nexus serves as a complementary mechanism to enhance performance.

To further explore Nexus formation, we averaged the Spatial and Route embeddings from models trained on Dataset I. Fig. 8(a) shows that the Spatial Nexus emphasizes local relationships within or between nearby regions, especially in Central, Fronto-Central, and Centro-Parietal areas. In contrast, Fig. 8(b) reveals that the Route Nexus reduces intraregion links while enhancing long-range inter-region connections, notably between Frontal and Parietal areas. These findings highlight the Route Nexus's role in modeling functional pathways relevant to MI processing.

B. Features Distribution

T-distributed stochastic neighbor embedding (t-SNE) is a nonlinear dimensionality reduction technique that effectively visualizes the distribution of multiscale data. Fig. 9 illustrates the feature distributions of A1 under different conditions. In the absence of NFL [Fig. 9(a) and (c)], the features exhibit significant overlap, indicating that the extracted features are insufficiently distinctive. However, with NFL [Fig. 9(b) and (d)], the features derived from both the training and testing data exhibit improved separation, demonstrating the effectiveness of NFL in feature extraction.

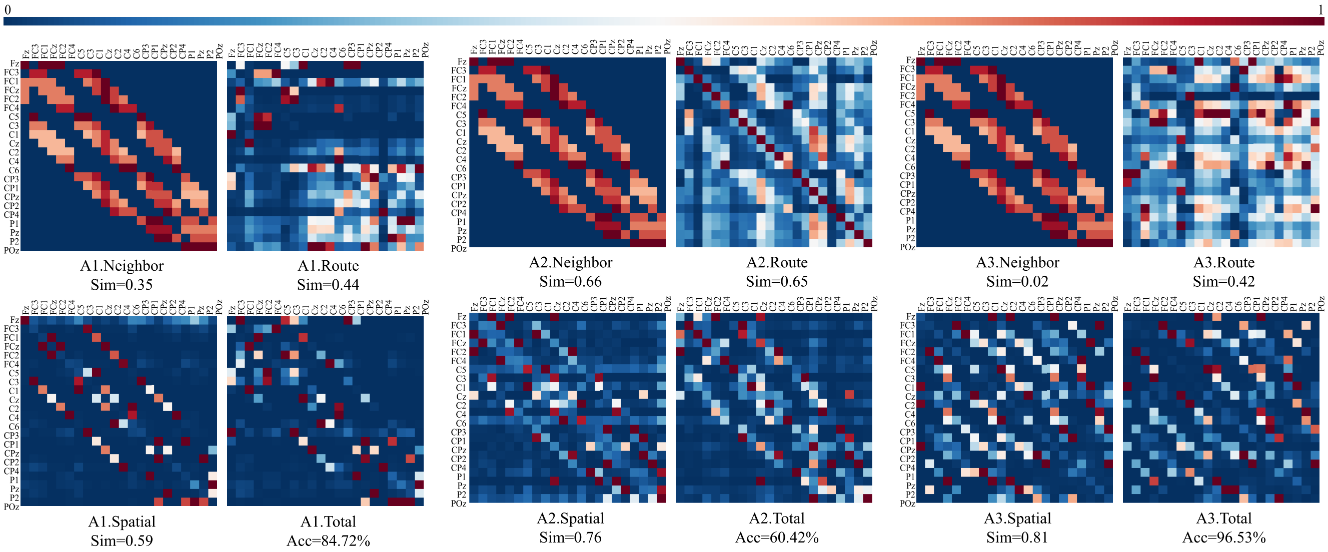


Fig. 7. Visualization of the Neighbor Nexus, Route Nexus, Spatial Nexus, and total Nexus (i.e., sum of the three) in Dataset I for subjects A1, A2, and A3. *sim* shows the cosine similarity between current Nexus and total Nexus, and *acc* marks the corresponding average accuracy presented in Table II. The color bar above indicates the normalized weights across different EEG channels.

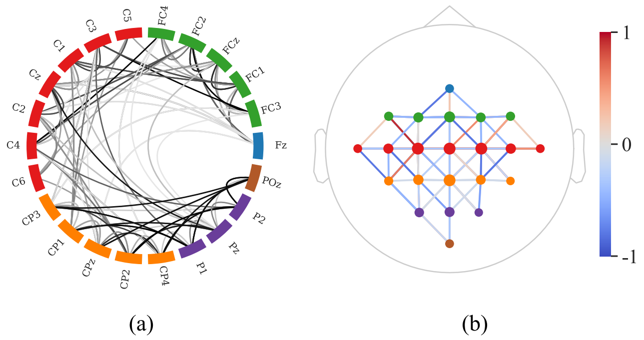


Fig. 8. Visualization of the (a) average Spatial Nexus and (b) average Route Nexus in Dataset I. Different brain regions are marked with corresponding colors, and in (b), the size of the scatter points represents the proportion of routes passing through the respective electrodes.

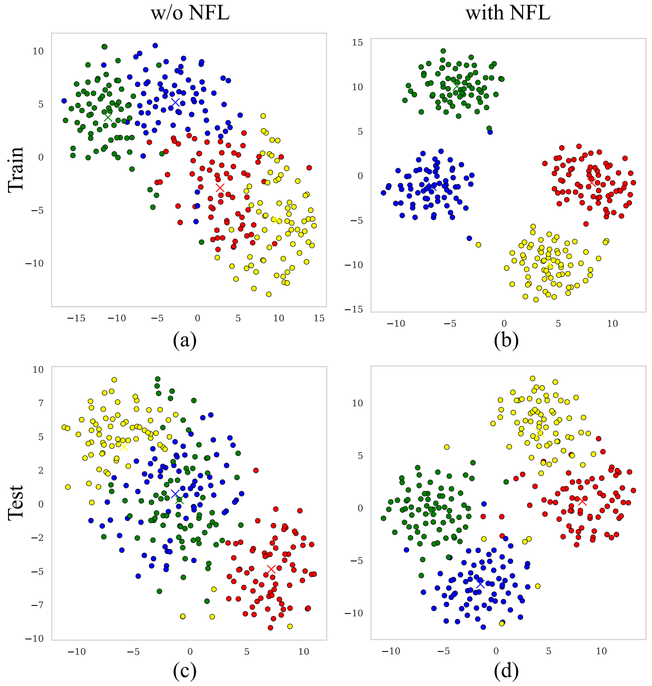


Fig. 9. T-SNE feature distribution from A1, where *w/o* stands for without, different colors represent distinct MI tasks, dots indicate individual features, and crosses mark the *K*-means cluster centers.

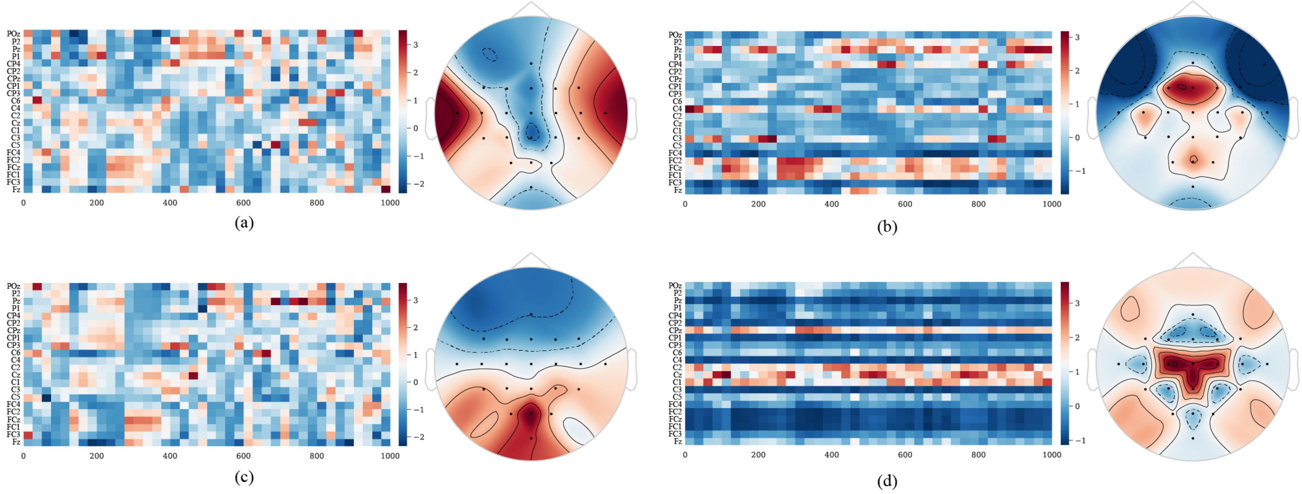
C. Energy Distribution

To visually compare the differences between raw trials and processed trials through connectome reconstruction, A1 and A2 were selected for spatiotemporal energy analysis. Fig. 10 illustrates the energy distribution differences among subjects following

$$\text{Energy}_i = \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T} x_{n,i,t}^2,\ i \in [1, E] \tag{18}$$

where $x_{n,i,t}$ is the signal value for the nth trial, ith electrode, and tth sample, N is the number of trails, E is the number of electrodes, and T is the number of samples. For temporal analysis, the signals were segmented, and the discrete energy of each segment with 25 samples was computed. For spatial analysis, the energy across all segments of a given electrode were summed to obtain its overall energy contribution.

From temporal analysis, the raw trials [Fig. 10(a) and (c)] exhibit energy fluctuations across segments. In subject A1 [Fig. 10(a)], the temporal heatmap shows sporadic high-energy bursts localized to specific temporal segments, suggesting transient neural synchronization during MI. After NFL processing [Fig. 10(b)], these transient peaks are attenuated, yielding a more uniform energy distribution across segments. A similar trend is observed in A2, where the raw trials display pronounced energy oscillations, while processed trials stabilize temporal dynamics. This implies that NFL processing suppresses noise-induced temporal variability, enhancing the consistency of energy patterns.



(a) (b) (c) (d)

Fig. 10. Distribution of MI energy periods and topological maps: (a) and (c) represent the raw MI trials' results for A1 and A2, respectively; (b) and (d) correspond to the MI trials for A1 and A2 processed by NFL. Red boxes highlight several channels with concentrated energy.

Spatially, raw trials demonstrate diffuse energy dispersion across electrodes, with weak focalization highlighted by scattered red areas in the heatmaps. For instance, in Fig. 10(a), energy contributions from electrodes C3 and Cz remain suboptimal, whereas in post-processed trials [Fig. 10(b)], the NFL method amplifies energy concentration at sensorimotor electrodes. The spatial heatmaps further confirms that processed trials achieve higher spatial specificity, with most of energy localized to task-relevant electrodes compared to raw trials.

VI. DISCUSSION

In our study, an innovative lightweight GNN, NexusNet, has been proposed. Different from conventional GNNs, NexusNet integrates Spatial Nexus, Route Nexus, Neighbor Nexus, and Centrality Nexus across electrodes. Specifically, the Spatial Nexus is responsible for learning structural information from electrodes, offering global insights. The Route Nexus learns the shortest paths between electrodes, capturing both endpoints and intermediates to model multinode cooperation. Meanwhile, the Neighbor Nexus and Centrality Nexus emphasize localized information, enriching the graph structure with detailed features. This streamlined architecture offers distinct advantages for IoT applications, where efficiency and real-time responsiveness are paramount. By decoding MI with minimal computational overhead, the lightweight structure achieves high performance while minimizing the demand for extensive resources.

To validate the performance of NexusNet, extensive experiments were conducted, achieving state-of-the-art results on both Dataset I (78.78%), Dataset II (87.21%), and Dataset III (94.12%). Despite being a lightweight network (3.44K parameters and 0.88M FLOPs), NexusNet outperformed Transformers, demonstrating the effectiveness of NFL. Confusion matrix analyzes and ablation studies were also performed. The results of the confusion matrix clearly showed that NexusNet achieved balanced classification outcomes, and ablation studies confirmed the indispensable value of every Nexus in the construction of the brain connectome. Furthermore, it was validated that, with long-range MI trials, NexusNet could gather more information about the brain connectome, enabling accurate Nexus reconstruction and significantly improving performance.

To further reveal the intrinsic meaning of our work, Neighbor Nexus, Spatial Nexus, and Route Nexus were displayed. The results indicated that when faced with challenging subjects, NexusNet tended to rely on Route Nexus to activate more peripheral regions to extract distinctive features. This observation was further validated in the energy maps. It was clear that with the modeling of different nexuses, certain peripheral regions regained importance. To have a comprehensive understanding of Spatial Nexus and Route Nexus, their embeddings were analyzed across nine subjects. The average Spatial Nexus showed strong connections within adjacent brain regions. The average Route Nexus revealed fewer intraregion connections but stronger inter-region links. Lastly, t-SNE visualized the features extracted by NexusNet, confirming that the reconstruction of the connectome reduced the ambiguity in the features.

Despite the advancements we have made, several limitations can be noticed. First, it has been confirmed through ablation studies that there is overlapping information between the Spatial Nexus and Route Nexus. Ensuring mutual independence in Nexuses is crucial for improving brain connectivity modeling and requires further explorations. Second, a key challenge lies in the existence of any common Nexus patterns. Should a shared Nexus be obtained, a pretrained model can be applied without calibration. Third, NexusNet has only been tested on MI trials, with its performance on other EEG paradigms remaining to be explored. Future work will concentrate on tackling these challenges to improve the applicability and practical effectiveness of NexusNet.

VII. CONCLUSION

Our study introduces NexusNet, a novel GNN designed for MI classification by leveraging the brain connectome. The proposed network diverges from traditional methods by incorporating multinode cooperation, which is critical for understanding the collaborative nature of brain activity during cognitive tasks. The experimental results over two public datasets yield promising outcomes, with NexusNet outperforming other baselines. Visual analyses ensure the interpretability of NexusNet. This work not only constructs a new brain connectivity graph but also lays the groundwork for the integration of MI-based BCIs with IoT systems. By enabling efficient and real-time decoding, NexusNet has the potential to enhance IoT applications, allowing for intuitive control of smart devices and improving the accessibility of assistive technologies. Both theoretical and experimental support for the further application of GNNs in MI classification are provided, with prospects for future IoT-driven BCI solutions.

REFERENCES

[1] I. Daly, S. J. Nasuto, and K. Warwick, "Single tap identification for fast BCI control," *Cogn. Neurodyn.*, vol. 5, no. 1, pp. 21–30, 2011.

[2] J. Jin, H. Zhang, I. Daly, X. Wang, and A. Cichocki, "An improved P300 pattern in BCI to catch user's attention," *J. Neural Eng.*, vol. 14, no. 3, 2017, Art. no. 36001.

[3] J. R. Wolpaw, J. D. R. Millan, and N. F. Ramsey, "Brain-computer interfaces: Definitions and principles," *Handbook Clin. Neurol.*, vol. 168, pp. 15–23, Mar. 2020.

[4] T. Sunny, T. Aparna, P. Neethu, J. Venkateswaran, V. Vishnupriya, and P. Vyas, "Robotic arm with brain–computer interfacing," *Procedia Technol.*, vol. 24, pp. 1089–1096, Jul. 2016.

[5] Z. Liu, L. Wang, S. Xu, and K. Lu, "A multiwavelet-based sparse time-varying autoregressive modeling for motor imagery EEG classification," *Comput. Biol. Med.*, vol. 155, Mar. 2023, Art. no. 106196.

[6] C. Tang, D. Jiang, L. Dang, and B. Chen, "EEG decoding based on normalized mutual information for motor imagery brain–computer interfaces," *IEEE Trans. Cogn. Develop. Syst.*, vol. 16, no. 6, pp. 1997–2007, Dec. 2024.

[7] G. Liu, R. Zhang, L. Tian, and W. Zhou, "Fine-grained spatial-frequency-time framework for motor imagery brain-computer interface," *IEEE J. Biomed. Health Inf.*, early access, Jan. 29, 2025, doi: 10.1109/JBHI.2025.3536212.

[8] Y. Shi, A. Jiang, J. Zhong, M. Li, and Y. Zhu, "Multiclass classification framework of motor imagery EEG by Riemannian geometry networks," *IEEE J. Biomed. Health Inform.*, vol. 29, no. 2, pp. 935–947, Feb. 2025.

[9] S. Sartipi and M. Cetin, "Subject-independent deep architecture for EEG-based motor imagery classification," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 32, pp. 718–727, 2024.

[10] G. Liang, D. Cao, J. Wang, Z. Zhang, and Y. Wu, "EISATC-fusion: Inception self-attention temporal convolutional network fusion for motor imagery EEG decoding," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 32, pp. 1535–1545, 2024.

[11] H. Wang, H. Yu, and H. Wang, "EEG_GENet: A feature-level graph embedding method for motor imagery classification based on EEG signals," *Biocybern. Biomed. Eng.*, vol. 42, no. 3, pp. 1023–1040, 2022.

[12] J. Wang, Y. Luo, L. Wang, L. Zhang, and X. Kang, "MTSAN-MI: Multiscale temporal-spatial convolutional self-attention network for motor imagery classification," in *Proc. Int. Conf. Neural Inf. Process*, 2023, pp. 338–349.

[13] C. Ju and C. Guan, "Graph neural networks on SPD manifolds for motor imagery classification: A perspective from the time–frequency analysis," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 12, pp. 17701–17715, Dec. 2024.

[14] J. Jia, B. Zhang, H. Lv, Z. Xu, S. Hu, and H. Li, "CR-GCN: Channel-relationships-based graph convolutional network for EEG emotion recognition," *Brain Sci.*, vol. 12, no. 8, p. 987, 2022.

[15] A. Demir, T. Koike-Akino, Y. Wang, M. Haruna, and D. Erdogmus, "EEG-GNN: Graph neural networks for classification of electroencephalogram (EEG) signals," in *Proc. 43rd Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, 2021, pp. 1061–1067.

[16] M. Axer and K. Amunts, "Scale matters: The nested human connectome," *Science*, vol. 378, no. 6619, pp. 500–504, 2022.

[17] M. Thiebaut de Schotten and S. J. Forkel, "The emergent properties of the connected brain," *Sci.*, vol. 378, no. 6619, pp. 505–510, 2022.

[18] X. Li, L. Sun, M. Ling, and Y. Peng, "A survey of graph neural network based recommendation in social networks," *Neurocomputing*, vol. 549, Sep. 2023, Art. no. 126441.

[19] B. Sun, H. Zhang, Z. Wu, Y. Zhang, and T. Li, "Adaptive spatiotemporal graph convolutional networks for motor imagery classification," *IEEE Signal Process. Lett.*, vol. 28, pp. 219–223, 2021.

[20] Q. She, T. Chen, F. Fang, Y. Gao, and Y. Zhang, "Discriminative adversarial network based on spatial–temporal–graph fusion for motor imagery recognition," *IEEE Trans. Comput. Social Syst.*, vol. 12, no. 3, pp. 972–983, Jun. 2025.

[21] D. Zhang, K. Chen, D. Jian, and L. Yao, "Motor imagery classification via temporal attention cues of graph embedded EEG signals," *IEEE J. Biomed. Health Inform.*, vol. 24, no. 9, pp. 2570–2579, Sep. 2020.

[22] B. Vivek, A. Adarsh, J. Gubbi, K. Muralidharan, R. K. Ramakrishnan, and A. Pal, "ST-GNN for EEG motor imagery classification," in *Proc. IEEE-EMBS Int. Conf. Biomed. Health Inform. (BHI)*, 2022, pp. 01–04.

[23] S. Cai, H. Li, Q. Wu, J. Liu, and Y. Zhang, "Motor imagery decoding in the presence of distraction using graph sequence neural networks," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 1716–1726, 2022.

[24] R. W. Floyd, "Algorithm 97: Shortest path," *Commun. ACM*, vol. 5, no. 6, pp. 345–345, 1962.

[25] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance, "EEGNet: A compact convolutional neural network for EEG-based brain–computer interfaces," *J. Neural Eng.*, vol. 15, no. 5, 2018, Art. no. 56013.

[26] C. Brunner, R. Leeb, G. Müller-Putz, A. Schlögl, and G. Pfurtscheller, 2008, "BCI competition 2008–Graz data set A," Graz University of Technology, IEEE Dataport. [Online]. Available: https://ieee-dataport.org/documents/bci-competition-2008-graz-data-set

[27] R. Leeb, C. Brunner, G. Müller-Putz, A. Schlögl, and G. Pfurtscheller, *BCI Competition 2008–Graz Data Set B*, Graz Univ. Technol., Graz Austria, 2008.

[28] R. T. Schirrmeister et al., "Deep learning with convolutional neural networks for EEG decoding and visualization," *Human Brain Map.*, vol. 38, no. 11, pp. 5391–5420, 2017.

[29] K. K. Ang, Z. Y. Chin, C. Wang, C. Guan, and H. Zhang, "Filter bank common spatial pattern algorithm on BCI competition IV datasets 2A and 2B," *Front. Neurosci.*, vol. 6, p. 39, Mar. 2012.

[30] R. Caruana, S. Lawrence, and C. Giles, "Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 13, 2000, pp. 402–408.

[31] R. Mane et al., "FBCNet: A multi-view convolutional neural network for brain-computer interface," 2021, *arXiv:2104.01233*.

[32] X. Li, X. Tang, S. Qiu, X. Deng, H. Wang, and Y. Tian, "Subdomain adversarial network for motor imagery EEG classification using graph data," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 8, no. 1, pp. 327–336, Feb. 2024.

[33] Y. Hou et al., "GCNs-Net: A graph convolutional neural network approach for decoding time-resolved eeg motor imagery signals," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 6, pp. 7312–7323, Jun. 2024.

[34] K. Liu et al., "MSVTNet: Multi-scale vision transformer neural network for eeg-based motor imagery decoding," *IEEE J. Biomed. Health Inform.*, vol. 28, no. 12, pp. 7126–7137, Dec. 2024.

[35] Y. Song, Q. Zheng, B. Liu, and X. Gao, "EEG conformer: Convolutional transformer for EEG decoding and visualization," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 710–719, 2023.