Joint Learning Neuronal Skeleton and Brain Circuit Topology with Permutation Invariant Encoders for Neuron Classification

Minghui Liao[1, 2, 3], **Guojia Wan**[1, 2, 3 *], **Bo Du**[1, 2, 3 *]

[1]National Engineering Research Center for Multimedia Software, School of Computer Science, Wuhan University, China
[2]Institute of Artificial Intelligence, Wuhan University, China
[3]Hubei Key Laboratory of Multimedia and Network Communication Engineering, Wuhan University, China
minghui@whu.edu.cn, guojia@whu.edu.cn, dubo@whu.edu.cn

Abstract

Determining the types of neurons within a nervous system plays a significant role in the analysis of brain connectomics and the investigation of neurological diseases. However, the efficiency of utilizing anatomical, physiological, or molecular characteristics of neurons is relatively low and costly. With the advancements in electron microscopy imaging and analysis techniques for brain tissue, we are able to obtain whole-brain connectome consisting neuronal high-resolution morphology and connectivity information. However, few models are built based on such data for automated neuron classification. In this paper, we propose NeuNet, a framework that combines morphological information of neurons obtained from skeleton and topological information between neurons obtained from neural circuit. Specifically, NeuNet consists of three components, namely Skeleton Encoder, Connectome Encoder, and Readout Layer. Skeleton Encoder integrates the local information of neurons in a bottom-up manner, with a one-dimensional convolution in neural skeleton's point data; Connectome Encoder uses a graph neural network to capture the topological information of neural circuit; finally, Readout Layer fuses the above two information and outputs classification results. We reprocess and release two new datasets for neuron classification task from volume electron microscopy(VEM) images of human brain cortex and *Drosophila* brain. Experiments on these two datasets demonstrated the effectiveness of our model with accuracy of 0.9169 and 0.9363, respectively. Code and data are available at: *https://github.com/WHUminghui/NeuNet*

Introduction

Identifying various neuron types in a nervous system is a fundamental task in numerous neuroscience studies. Assigning neuron type is a necessary step to understand the anatomical and functional properties of a brain circuit (Armañanzas and Ascoli 2015). By identifying and classifying various types of neurons, researchers can determine which neurons are most susceptible to causing some neurodegenerative diseases, such as Alzheimer's(Buckner et al. 2005) and Amyotrophic Lateral Sclerosis (ALS). However, due to the vast number and minuscule size of neurons, anatomical-based neuron classification methods are exceedingly time-consuming and labor-intensive.

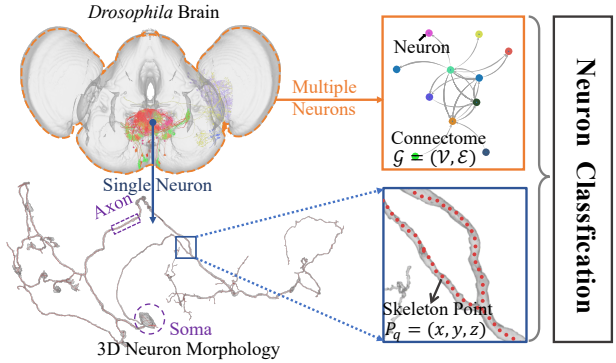


Figure 1: Neuron connectivity (orange) and morphology (blue) in a *Drosophila* (fruit fly) brain. For each neuron, its connectivity can be represented by a graph. Its morphology can be described by a skeleton that is a set of 3D points. Both of them can reveal neuron's functions and roles.

Recently, volume electron microscopy (VEM) imaging and analysis techniques (Januszewski et al. 2018) contribute the availability of whole-brain three-dimensional(3D) reconstruction with cellular resolution, providing a new opportunity to understand the neuronal properties and functions. Specifically, the 3D images of brain tissue generated by VEM contain a wealth of information about neuron size, location, morphology, and connectivity. We show a neuron in Figure 1 from *Drosophila* connectome (Scheffer et al. 2020). We can observe that the morphology of the complex nerve fibers (including axons and dendrites) that spans multiple brain regions. The red centerline within the fiber voxel is referred as the skeleton of the neuron, which can be depicted as a set filling with 3D points (in XYZ coordinates), $P = \{p_q | q = 1, ..., m\}$. A large number of such neurons are interconnected, forming the connectome of *Drosophila* brain (Scheffer et al. 2020). The connectome can be depicted by a complex network, which is formalized as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. In this network, nodes represent the neurons and edges represent the synaptic connections between them. Apparently, both of these two data exhibit permutation invariance. That is, permuting or rearranging the order of the ele-

*corresponding author

ments in these data does not change the results or the information conveyed by the data.

The morphology, or shape, of a neuron can provide significant insights into its function within the nervous system. Traditional methods based on feature statistics (Gillette and Ascoli 2015; Kanari et al. 2019; Zhu et al. 2022) usually conduct analysis on the optical data. However, the resolution of neuronal morphological data obtained from optical microscopes is relatively low, which cannot provide neuronal connectivity information. Importantly, the connectivity pattern (Fulcher and Fornito 2016) of neuron can also reveal the properties about its function.

To date, jointly learning the morphology and connectivity remains unexplored. In this paper, we propose a unified, efficient framework called NeuNet, a neuron classification framework that incorporates morphological information of neuron skeleton and topological information of connectome. NeuNet consists of a skeleton-level feature extractor and a connectome-level feature extractor, which well respect permutation invariance of points and topology connectivity in the input. After extracting the morphological features from neuron skeleton data and the topological information from the connectome, a fusion module combines the above two features to finally obtain the representation of the neuron. The main contributions of NeuNet are as follows:

- NeuNet can learn neuronal representations of entire brain circuit in a high-throughput and faster manner and use them for downstream tasks.
- Our model incorporates the skeleton's morphological information and the connectome's topological information of neurons, both of which are in a permutation-invariant manner, and it is in line with the actual characteristics of neuron VEM data.
- We reprocessed and released two 3D brain reconstruction data. We introduced baseline models from other tasks, and upon comparative evaluation, our model manifested superior classification performance, successfully capturing the latent feature representations of neurons.

Related Work

Neurons exhibit diverse properties in terms of connectivity, morphology, and functionality, and the classification of neurons is the cornerstone for exploring the structure and functional mechanisms of brain circuit(Wagner, Regev, and Yosef 2016). Since the functions of neuron are closely related to its morphology, several methods have been proposed for comparing and classifying neurons from light-microscopy (Markram et al. 2015; Costa et al. 2016a; Peng et al. 2015; Sümbül et al. 2014).

With the development of electron microscope imaging speed (Eberle et al. 2015), electron microscopy can provide higher resolution and more informative photographs of brain tissue sections. Taking advantage of these innovations, the brain circuit reconstruction have been made to map the connectivity in *Drosophila* optic and antennal lobes (Berck et al. 2016; Takemura et al. 2013), the mouse retina, thalamus, and cortex (Greene, Kim, and Seung 2016; Lee et al. 2016; Morgan et al. 2016). However, there are relatively few works (Costa et al. 2016b; Schubert et al. 2019; Zhu et al. 2022) that use neural networks for feature extraction and classification of the above-described neuron data, and none of which take into account the topological information of brain circuit.

Methodology

Notations and Problem Definition

Neuron Data. We designed NeuNet to integrate morphological features about the skeleton and topological features about connectome. A skeleton-level neuron data is represented as a set of points $P = \{p|p = (x, y, z)\}$, where each point p is a corrdinate vector.

For modeling connectivity, we regard each neuron as a node. Let synapse connections from pre-synaptic cells to post-synaptic cells be represented as edges, where the number of synapses between two neurons is as connection strength. We then constructed a graph to represent the connectome. We denote the graph with n nodes (neurons) by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ represents its node set and $\mathcal{E}$ represents the set of edges. The adjacency matrix of $\mathcal{G}$ is denoted by $\mathbf{A} \in R^{n\times n}$ with its (i, j)-th entry $A[i, j] = e$ indicating the strength of the connection between node i and node j, and e is numerically equal to the number of synapses connected between neuron i and neuron j.

Permutation Invariance. Our input include a point set and a graph. The point set is inherently unordered. That is, permuting the input order of points within this point set does not affect the skeleton morphology of the neuron. As for the input graph, it is a 2-tuple $(\mathcal{V}, \mathcal{E})$ that consists of two sets: a node set $\mathcal{V}$ and an edge set $\mathcal{E}$. Likewise, it should follow permutation invariance as well.

Due to its permutation invariance, any learners working with such data should be invariant under transformations. For example, changing the input order of points, nodes and edges should not modify its neuron classification results.

Neuron Classification. In the task of neuron classification, we are given skeleton-level point set P, the connectome level graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, and the neuron labels denoted by one-hot vector Y. Each labeled node $i \in \mathcal{V}$ is associated with a one-hot vector $y_i \subset \{0, 1\}^C$ which encodes its ground-truth class, where C is the number of predefined classes. We needed to learn a function $f_\theta(i|(\mathcal{G}, P))$ parameterized by θ which can predict the correct class for a given unlabeled node i.

Network Architecture

NeuNet consists of three sub-modules (Figure 2): *Skeleton Encoder*, *Connectome Encoder*, and *Readout Layer*. Skeleton Encoder is used to extract the morphological information of neuron; Connectome Encoder is used to extract the topological information of the neural circuit; Readout Layer fuses the skeleton-level and connectome-level representation, and output a $1 \times C$ classification score vector with a Multilayer Perceptron(MLP).

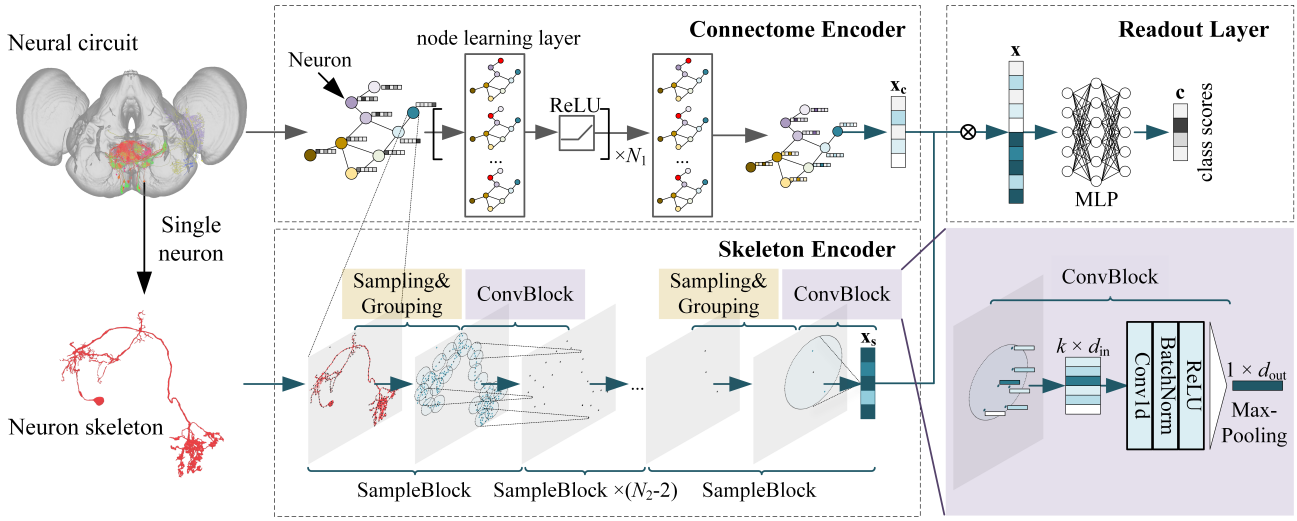


Figure 2: Architecture of proposed NeuNet. On the one hand, the skeleton data of neurons are input into Skeleton Encoder, and Sampling operation samples the points in the skeleton's point set using FPS. Grouping operation divides the attached points into groups with the sampled points as the center, and ConvBlock extracts the features on these group's points and fuses them using max pooling to obtain the group's global features. The above process is repeated until the skeleton's global features are obtained. On the other hand, GNN-based Connectome Encoder mines the topological features of the neural circuit through the information interaction between nodes and neighboring nodes. Finally, Readout Layer fuses the above two features and outputs the classification results with an MLP.

Skeleton Encoder

Neurons possess intricate branching patterns, encompassing fundamental local structures such as axons, dendrites, and curvatures. These complex branching morphologies often provide insights into the classification and functional roles of neuron (Udvary et al. 2022). However, inferring the functions and classifications of neurons directly from their 3D structures remains a challenging task. Therefore, there arises a necessity to develop morphology feature extractor for neuron morphology, which is able to obtain global representation of the neuron skeleton concerning its local structure.

Considering that neuron skeleton data is an unordered point set, we designed Skeleton Encoder in bottom-up manner consisting of multiple SampleBlocks, which can extract consistent feature representations from varying input orders of skeleton data pertaining to the same neuron.The SampleBlock abstraction level is made of three key operations (Figure 2): *Sampling*, *Grouping*, and *Conv1D*.

Sampling. The key component of CNN is the convolution kernel, which extracts local features of an image by performing Hadamard product with the certain region (kernel size) of pixels, and acquires a larger range of semantic features by multiple convolutional operations. Drawing inspiration from this concept, we aim to construct local feature extractor working on different regions of a neuron skeleton. Initially, it is imperative to determine the regions with which this feature extractor is intended to interact. Given input points $\{p_1, p_2, ..., p_m\}$, we employ iterative farthest point sampling(FPS) (Li et al. 2022) to select a subset of points $\{p_{i_1}, p_{i_2}, ..., p_{i_s}\}$ as the centroids of these regions. This selection ensures that the sampled points are the farthest apart(in European distance in this study). In contrast to random sampling, FPS can offers a wider coverage, thereby establishing a more comprehensive receptive field among the given points. This facilitates the acquisition of global morphological information from the input points.

Grouping. The subsequent task involves delineating the respective extent of these regions around the central points. For a given set of points $P = \{p_1, p_2, ..., p_m\}$ and points $S = \{p_{i_1}, p_{i_2}, ..., p_{i_s}\}$ selected by Sampling, we take S as the centroids and find k points in P that are most adjacent to the centroid respectively to form s group. Considering that the density and distribution of points in P are not uniform, we use ball query(Qi et al. 2017) to find k points that are within a radius of the centroid point to guarantee a fixed regional scale. The input to the Grouping operator is a point set of size m, and the output are groups of point sets of size $s \cdot k$, where each group corresponds to a local region and k is the number of points in the neighborhood of centroid points.

Conv1D. For the points of the groups generated by Grouping operation, they retain an inherent unordered nature. In order to extract consistent features from groups' points irrespective of the order of input points, we employ one-dimensional convolution for capturing local morphology features within the groups. The input skeleton feature $\mathbf{X}_{\text{in}} \in R^{m \times d_{\text{in}}}$ is fed into Conv1D and is transformed into the next latent skeleton feature $\mathbf{X}_{\text{out}} \in R^{m \times d_{\text{out}}}$ as $\mathbf{X}_{\text{out}} = \text{ReLU}(\mathbf{W} \star \mathbf{X}_{\text{in}} + \mathbf{b})$, where $\mathbf{W} \in R^{\kappa \times d_{\text{in}} \times d_{\text{out}}}$ is the kernel and κ is the kernel size. $\mathbf{b}$ denotes bias. $\star$ is the valid matrix multiplication. When the initial input is the point set P, the corresponding skeleton feature is $\mathbf{X}_{xyz} \in R^{m \times 3}$. Otherwise, d_{in} and d_{out} are intermediate latent feature dimensions

Method	Acc	LC	KC	SMP	AVLP	SLP	CL	PS	LHAV	PLP	LHPV
MLP	0.1796	0.1763	0.1652	0.1798	0.2032	0.1863	0.1765	0.1956	0.1652	0.1956	0.1698
PointNet++ (Qi et al. 2017)	0.8627	0.8621	0.8962	0.7584	0.8546	0.8654	0.8756	0.8745	0.8522	0.8921	0.8541
GCNII (Chen et al. 2020)	0.5266	0.4956	0.5361	0.5411	0.5323	0.5362	0.5412	0.5212	0.5421	0.4812	0.4932
DGCNN (Wang et al. 2019)	0.8560	0.8425	0.8632	0.8410	0.8651	0.8510	0.8710	0.8632	0.8365	0.8863	0.8360
AGNN (Thekumparampil et al. 2018)	0.5327	0.5621	0.5214	0.5412	0.5630	0.5489	0.5321	0.5423	0.5621	0.5421	0.5213
Point-Transformer (Zhao et al. 2021)	0.8473	0.8516	0.8326	0.8426	0.8321	0.8561	0.8502	0.8632	0.8412	0.8564	0.8214
NeuNet(ours)	**0.9169**	**0.9721**	**1.0000**	**0.9291**	**0.9259**	**0.9369**	**0.9200**	**0.9000**	**0.8929**	**0.9245**	**0.8813**

Table 1: Overall classification accuracy on the HemiBrain dataset and within-class accuracy for 10 of its classes.

Method	Acc	L1	L2	L3	L4	L5	L6
MLP	0.3239	0.3364	0.3562	0.3235	0.3102	0.3321	0.3014
PointNet++	0.6242	0.6025	0.6821	0.6214	0.6025	0.6412	0.6123
GCNII	0.6141	0.6231	0.6512	0.5831	0.6451	0.5821	0.6612
DGCNN	0.6104	0.6120	0.6521	0.6023	0.6325	0.6230	0.5984
AGNN	0.6334	0.6231	0.6213	0.6423	0.5921	0.6215	0.6432
Point-Transformer	0.5910	0.6012	0.5821	0.6123	0.6215	0.6023	0.5920
NeuNet	**0.9363**	**0.8113**	**0.9729**	**0.9134**	**0.9306**	**0.9273**	**0.9247**

Table 2: Overall classification accuracy on the H01 dataset and within-class accuracy on its 6 layers. L1, L2, ..., L6 are the 6 types of cells at different layers of the human cerebral cortex.

as hyperparameters. m is the number of points. For the j-th Conv1D layer, $m^{(j)}$ is the number of points. When applying sampling and grouping operations, $m^{(j-1)}$ of the last layer becomes $m^{(j)} = s \cdot k$.

Here, we set the kernel with $\kappa = 1$ and stride=1. In this way, Eq. () becomes a feature learner that is invariant to permutation of a point set along the 'm' axis.

After each Conv1D layer, we leverage MaxPooling as a downsampling operation and reduce the spatial dimensions. Subsequently, by stacking N_2 {Sampling, Grouping, ConvBlock, MaxPooling} layers, the input $\mathbf{X}_{xyz} \in R^{|P|\times 3}$ are projecting into a morphology representation $\mathbf{x}_s \in R^{1\times d}$ layer by layer. The value of N_2 was set to 3 in experiments.

Connectome Encoder

A brain circuit is inherently a graph, with neurons interconnected through synaptic or chemical connections. The connectivity of such graph forms the basis for biological functions and behaviors. For neuron classification, the connectivity pattern can reveal the functions and roles of neurons. To establish a mapping between learning connection patterns and neuron types, we design Connectome Encoder as a graph encoder to learn the topological features of this graph. Connectome Encoder iteratively stacks N_1 node learning layers to project graph structure into low-dimensional features. The l-th layer of Connectome Encoder is a function $\mathcal{F} : (\mathbf{A}, \mathbf{X}^{(l)} \in R^{|\mathcal{V}|\times d_{(l)}}) \to \mathbf{X}^{(l+1)} \in R^{|\mathcal{V}|\times d_{(l+1)}}$ as

$$\mathbf{X}^{(l+1)} = \mathrm{ReLU}\left(((1-\alpha)\hat{\mathbf{P}}\mathbf{X}^{(l)} + \alpha\mathbf{X}^{(\mathbf{0})})((1-\beta)\mathbf{I} + \beta\mathbf{\Theta})\right), \tag{1}$$

where $\hat{\mathbf{P}} = \hat{\mathbf{D}}^{-1/2}\hat{\mathbf{A}}\hat{\mathbf{D}}^{-1/2}$ is a degree normalized adjacency matrix with addinng self-loops that $\hat{\mathbf{A}} = \mathbf{A} + \mathbf{I}$. $\hat{\mathbf{D}}$ is a diagonal degree matrix, where $\hat{D}_{ii} = \sum_j \hat{A}_{ij}$. $\mathbf{X}^{(0)} \in R^{|\mathcal{V}|\times d_0}$ is the initial feature matrix from a random initialization. $\mathbf{\Theta} \in R^{d_l \times d_{(l+1)}}$ is a learnable weight matrix.

Here, α models the strength of the initial residual connection, while β models the strength of the identity mapping. The final feature $\mathbf{X}_\mathrm{c}$ obtained after N_1 layers of propagation encapsulates the topological information of the contextual surroundings in which the neuron resides. Then, the row vector $\mathbf{x}_\mathrm{c} \in \mathbf{X}_\mathrm{c}$ represent a neuron's topological low-dimensional feature representation. In our experiment, the values of α, β and N_1, as mentioned above, were set to 0.1, 0.5, 64 respectively.

Since a graph is a topological structure, its data structure also possesses permutation invariance. This means that permuting the node orders of the graph will not change the learned representation vector of the nodes.

Readout Layer

After Skeleton Encoder generates feature representations $\mathbf{x}_s$ about neuron morphology and Connectome Encoder generates feature representations $\mathbf{x}_\mathrm{c}$ about the topological information, we concatenate them to form the final neuron feature representations $\mathbf{x}$, $\mathrm{x}=\mathrm{x}_s \otimes \mathbf{x}_c$. Then, we use an MLP as a classifier to predict neuron label: $\mathrm{c}=\mathrm{MLP}(\mathrm{x})$, where $\mathbf{c}$ denotes the prediction results.

Finally, we use cross entropy as a classification loss to train our model: $\mathrm{L}=-\sum_i^{|C|}[y_i \log c_i + (1-y_i)\log(1-c_i)]$, where $y_i \in \{0, 1\}$ is the groud truth for i-th type. $|C|$ is the number of neuron types.

Experiments

Data Description

HemiBrain. HemiBrain(Scheffer et al. 2020) is a dense reconstruction of a portion of the central brain of *Drosophila*. We accessed the raw data through NeuPrint(Clements et al. 2020) and reprocessed it into 19,384 skeletons(point sets) of neurons which can be classified into 191 different classes. Regarding these neurons as nodes and

Figure 3: HemiBrain neuron skeletons of KC (Kenyon Cells), LHAV(Lateral Horn Anterior Ventral cell cluster), MC(Medulla Columnar), SLP(Superior lateral protocerebrum), and WED(Wedge). The first column are neurons (blue) from the training set, the rest of the columns are NeuNet's prediction cases on test, and the gray background is *Drosophila* brain. NeuNet learned the morphology of the neurons, including the "L" shaped structure of the KC class.

the synapses between neurons as connecting edges, these neurons are connected by 3,307,248 edges.

H01. H01(Shapson-Coe et al. 2021) is from a human cerebral cortex through steps such as slicing, imaging, segmentation, and synaptic detection. We obtained the raw data and reprocessed it into 15,554 neurons connected by 28,246 edges, which are from 6 layers of the cortex. The neurons of HemiBrain and H01 datasets will be randomly partitioned into training, validation, and test dataset in an 8:1:1 ratio.

Results

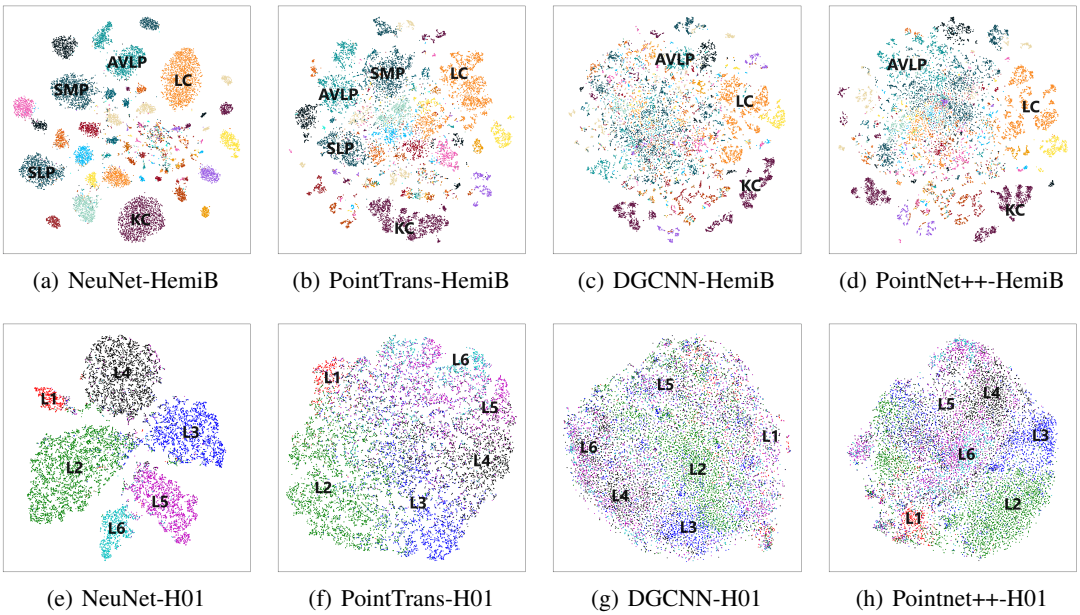
Neuron Classification. On the HemiBrain and H01 dataset, we compare our method with MLP, PointNet++ (Qi et al. 2017), GCNII (Chen et al. 2020), DGCNN (Wang et al. 2019), AGNN (Thekumparampil et al. 2018) and Point-Transformer (Zhao et al. 2021). The classification results are listed in the Table 1 and Table 2. The numerical values reported in this study refer to either overall accuracy or within-class accuracy. Overall accuracy represents the proportion of correctly classified samples out of the total number of samples. While within-class accuracy pertains to the proportion of correctly classified samples within a specific category out of the total number of samples in that category. As Table 1 suggests, our model outperforms other methods on the HemiBrain dataset. And it is ahead of the second-place model PointNet++ by 6.46% and ahead of the last-place MLP by 411.41%. As shown in Table 2, our model also significantly outperforms the other models on the H01 dataset, with an overall accuracy of 48.70% higher than the second place.

We draw skeletons of neuron randomly selected from the HemiBrain dataset. As shown in Figure 3,NeuNet is able to learn the local and global structure of neurons' skeletons. For example, NeuNet learns an 'L'-type structure with complex bifurcation structures at both ends in KC; NeuNet recognizes the complex dendritic structure followed by a long axonal structure as a key characteristic in LHAV.

Representation Visualization. We produce t-SNE (Van der Maaten and Hinton 2008) visualizations on $\mathbf{x}$ learned by various methods. As shown in Figure 4, our model can transform the complex neuronal data into clustered groups, demonstrating the strong classification ability of the learned representations. However, this ability is not observed in DGCNN and PointNet++. Fuzzy grouping is also observed in Point-Transformer, with the scatter within the groups being more dispersed and overlapping occurring between the groups. Additionally, we can find that the representation of sub-classes also tend to cluster together, even if the subclass labeling information is not used during the training phase.

Neuron Retrieval. The representational capacity of NeuNet for high-dimensional neuron data can be applied to neuron retrieval. On the HemiBrain dataset, We use $\mathbf{x}$ to compute the Euclidean distance between all neurons. From the test set, we randomly selecte four neuron representations and identify the nearest neighboring neurons. The experimental results are shown in Figure 5. It is noteworthy that the subclasses of the top10 neurons are consistent with the query neuron (subclass information of the retrieved neurons is not shown in the figure), despite the fact that subclass information was not utilized during the training phase.

From Figure 5, we can observe that the retrieved neurons exhibit a high degree of similarity to the query neuron, and



(a) NeuNet-HemiB (b) PointTrans-HemiB (c) DGCNN-HemiB (d) PointNet++-HemiB

(e) NeuNet-H01 (f) PointTrans-H01 (g) DGCNN-H01 (h) Pointnet++-H01

Figure 4: The visualizations using t-SNE (Van der Maaten and Hinton 2008) of neuron representation learned by various methods. Where each of color represents a specific class. Since there are up to 191 class of HemiBrain, different classes of point may be given the same color.

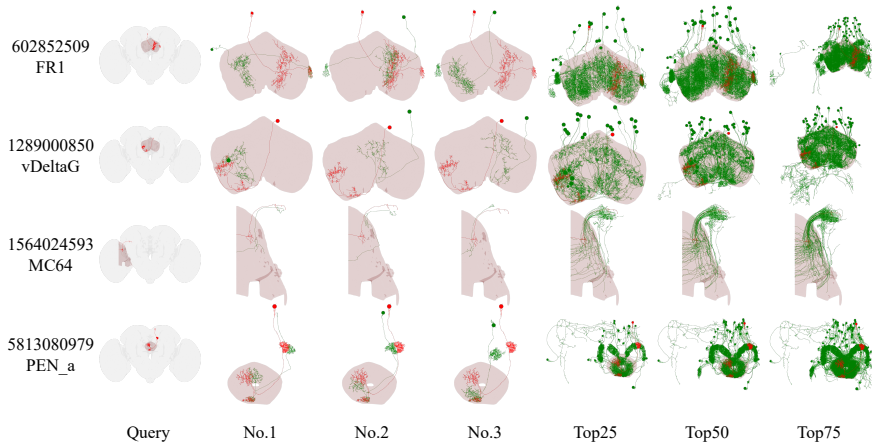


Figure 5: Query neurons(red) and retrieved neurons(green) using NeuNet. The gray-shaded background represents the *Drosophila* brain, and the red shaded background indicates the region with the highest synaptic connectivity for the query neuron. Query neuron IDs and finer-grained categories are provided by (Scheffer et al. 2020). The subcategories of the top 10 retrieved neurons remain consistent with the query neurons(not shown in this figure), without the utilization of finer-grained subcategory information in NeuNet.

the retrieved neurons share common morphologies and location, indicating the potential presence of functional units within the corresponding regions.

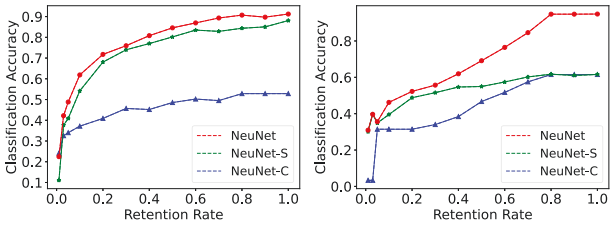
Ablation Study. We replace the way of fusing features in Readout Layer from concatenation to max pooling and average pooling to construct NeuNet(maxPool) and NeuNet(avgPool), respectively; we discard the brain circuit information and use only the representation generated by Skeleton Encoder to design the NeuNet-S; we remove skeleton information and employ solely the topological information generated by Connectome Encoder to design NeuNet-C. The experimental results of the aforementioned variants on the HemiBrain and H01 datasets are shown in Table 3 and Table 4, respectively. It can be seen that the ways of information fusion have little effect. When the NeuNet discards the skeleton information, its effectiveness decreases by 36.03% and 42.40% on the HemiBrain and H01 datasets, respectively. Similarly, when NeuNet discards the connectome information, the decreases on the HemiBrain and H01

Method	Acc	LC	KC	SMP	AVLP	SLP	CL	PS	LHAV	PLP	LHPV
NeuNet	**0.9169**	0.9721	**1.0000**	**0.9291**	0.9259	**0.9369**	0.9200	**0.9000**	0.8929	**0.9245**	0.8813
NeuNet(maxPool)	0.9149	0.9814	**1.0000**	**0.9291**	**0.9407**	0.9279	**0.9600**	0.8714	**0.9821**	0.8929	**0.9623**
NeuNet(avgPool)	0.9135	**0.9907**	**1.0000**	0.9220	0.9185	0.9279	0.9067	**0.9000**	0.9464	0.8571	0.9434
NeuNet-S	0.8612	0.8512	0.8325	0.8754	0.8695	0.8720	0.8426	0.8365	0.8714	0.8845	0.7965
NeuNet-C	0.5865	0.6021	0.6231	0.5841	0.5923	0.5814	0.5736	0.5596	0.6021	0.6254	0.6123

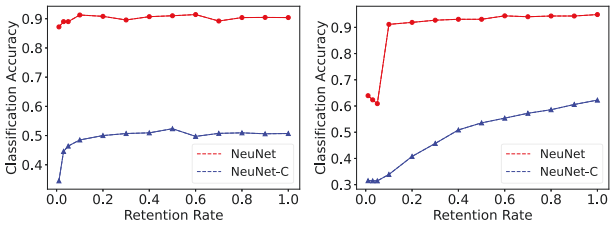
Table 3: Classification accuracy of various variants of the NeuNet on the HemiBrain dataset.

Method	Acc	L1	L2	L3	L4	L5	L6
NeuNet	0.9363	0.8113	**0.9729**	0.9134	**0.9306**	0.9273	**0.9247**
NeuNet(maxPool)	0.9412	**0.9436**	0.9422	**0.9449**	0.9273	**0.9462**	0.8679
NeuNet(avgPool)	**0.9419**	0.9415	0.9624	**0.9449**	0.9273	**0.9462**	0.8679
NeuNet-S	0.6356	0.6541	0.6125	0.6023	0.6152	0.6201	0.6312
NeuNet-C	0.5123	0.5210	0.5023	0.5421	0.5011	0.5102	0.5321

Table 4: Classification accuracy of various variants of NeuNet on the H01 dataset.



(a) Node scale on HemiBrain (b) Node scale on H01



(c) Edge scale on HemiBrain (d) Edge scale on H01

Figure 6: The impact of varying the node and edge scale of the training set on classification accuracy.

datasets are 6.07% and 32.11%, respectively. This demonstrates that the two types of information complement each other, and NeuNet addresses the issue of the current methods lacking the utilization of brain circuit's topological information.

To investigate the performance of NeuNet under different scales of training samples and connectivity edge quantities, we conduct experiments on the training set while retaining varying proportions of neurons and connectivity edges. As shown in Figure 6, the model variants achieved 60% of their optimal performance with only 0.1 times the number of neurons used for training. Additionally, we can observe that the combination of skeleton and connectome information enhances the model's performance, which is particularly evident on the H01 dataset(Figure 6(b)).

Encoder-C is performed by message passing mechanism such that its time complexity is $\mathcal{O}(|\mathcal{V}| + |\mathcal{E}|)$ (Chiang, Wei-Lin et.al., 2019 KDD)-linear in the number of network edges $|\mathcal{E}|$ and nodes $|\mathcal{V}|$, where we omit hyperparameter term. Encoder-S is $\mathcal{O}(|P|)$-linear in the point amount of one neuron. Practical training times are shown in Table 5. The skeleton part has more data than connectome data such that Encoder-S dominates the run time. On HemiBrain, NeuNet is fast that one epoch takes about **77 s**, and inferring a single sample takes about 0.0044 **s.**(tested on RTX 2080Ti).

Scale		0.1	0.3	0.5	0.7	0.9
Encoder-C	H-brain	26.85	30.79	34.94	38.53	43.84
Encoder-S	H-brain	329.35	617.90	916.42	1579.35	1513.44
NeuNet	H-brain	356.88	649.78	952.43	1619.00	1558.36
Encoder-C	H01	8.03	9.37	11.75	13.49	15.33
Encoder-S	H01	271.48	503.12	723.85	946.36	1159.55
NeuNet	H01	280.57	512.99	736.70	961.91	1176.94

Table 5: Run time (Seconds) under different scales of training scales.

Conclusion

In this paper, we propose a framework called NeuNet specifically for the neuron classification task, which incorporates both the morphological information of neuronal skeleton and the topological information of brain circuit with permutation invariance. NeuNet's Skeleton Encoder mines the morphological information of neuronal sekleton in a bottom-up manner, while NeuNet's Connectome Encoder employs a GNN to extract the topological information of brain circuit. The two information are fused and utilized by the Readout Layer for classification. Furthermore, we reprocess and release two datasets for neuron classification task. On these datasets, NeuNet demonstrate superior performance compared to other baseline models.

Acknowledgements

This work is partially supported by National Key RD Program of China (2023YFC2705700), the National Natural Science Foundation of China (62206202, 62225113), China Postdoctoral Science Foundation, China (2022M712461), Artificial Intelligence Innovation Project of Wuhan Science and Technology Bureau (No. 2022010702040070), The Fundamental Research Funds for the Central Universities, China (2042022kf1043).

References

Armañanzas, R.; and Ascoli, G. A. 2015. Towards the automatic classification of neurons. *Trends in Neurosciences*, 38(5): 307–318.

Berck, M. E.; Khandelwal, A.; Claus, L.; Hernandez-Nunez, L.; Si, G.; Tabone, C. J.; Li, F.; Truman, J. W.; Fetter, R. D.; Louis, M.; et al. 2016. The wiring diagram of a glomerular olfactory system. *Elife*, 5.

Buckner, R. L.; Snyder, A. Z.; Shannon, B. J.; LaRossa, G.; Sachs, R.; Fotenos, A. F.; Sheline, Y. I.; Klunk, W. E.; Mathis, C. A.; Morris, J. C.; et al. 2005. Molecular, structural, and functional characterization of Alzheimer's disease: evidence for a relationship between default activity, amyloid, and memory. *Journal of Neuroscience*, 25(34): 7709–7717.

Chen, M.; Wei, Z.; Huang, Z.; Ding, B.; and Li, Y. 2020. Simple and deep graph convolutional networks. In *International Conference on Machine Learning (ICML)*, 1725–1735. PMLR.

Clements, J.; Dolafi, T.; Umayam, L.; Neubarth, N. L.; Berg, S.; Scheffer, L. K.; and Plaza, S. M. 2020. neuPrint: analysis tools for EM connectomics. *BioRxiv*.

Costa, M.; Manton, J. D.; Ostrovsky, A. D.; Prohaska, S.; and Jefferis, G. S. 2016a. NBLAST: rapid, sensitive comparison of neuronal structure and construction of neuron family databases. *Neuron*, 91(2): 293–311.

Costa, M.; Manton, J. D.; Ostrovsky, A. D.; Prohaska, S.; and Jefferis, G. S. 2016b. NBLAST: rapid, sensitive comparison of neuronal structure and construction of neuron family databases. *Neuron*, 91(2): 293–311.

Eberle, A.; Mikula, S.; Schalek, R.; Lichtman, J.; Tate, M. K.; and Zeidler, D. 2015. High-resolution, high-throughput imaging with a multibeam scanning electron microscope. *Journal of Microscopy*, 259(2): 114–120.

Fulcher, B. D.; and Fornito, A. 2016. A transcriptional signature of hub connectivity in the mouse connectome. *Proceedings of the National Academy of Sciences*, 113(5): 1435–1440.

Gillette, T. A.; and Ascoli, G. A. 2015. Topological characterization of neuronal arbor morphology via sequence representation: I-motif analysis. *BMC Bioinformatics*, 16: 1–15.

Greene, M. J.; Kim, J. S.; and Seung, H. S. 2016. Analogous convergence of sustained and transient inputs in parallel on and off pathways for retinal motion computation. *Cell Reports*, 14(8): 1892–1900.

Januszewski, M.; Kornfeld, J.; Li, P. H.; Pope, A.; Blakely, T.; Lindsey, L.; Maitin-Shepard, J.; Tyka, M.; Denk, W.; and Jain, V. 2018. High-precision automated reconstruction of neurons with flood-filling networks. *Nature Methods*, 15(8): 605–610.

Kanari, L.; Ramaswamy, S.; Shi, Y.; Morand, S.; Meystre, J.; Perin, R.; Abdellah, M.; Wang, Y.; Hess, K.; and Markram, H. 2019. Objective morphological classification of neocortical pyramidal cells. *Cerebral Cortex*, 29(4): 1719–1735.

Lee, W.-C. A.; Bonin, V.; Reed, M.; Graham, B. J.; Hood, G.; Glattfelder, K.; and Reid, R. C. 2016. Anatomy and function of an excitatory network in the visual cortex. *Nature*, 532(7599): 370–374.

Li, J.; Zhou, J.; Xiong, Y.; Chen, X.; and Chakrabarti, C. 2022. An adjustable farthest point sampling method for approximately-sorted point cloud data. In *IEEE Workshop on Signal Processing Systems (SiPS)*, 1–6. IEEE.

Markram, H.; Muller, E.; Ramaswamy, S.; Reimann, M. W.; Abdellah, M.; Sanchez, C. A.; Ailamaki, A.; Alonso-Nanclares, L.; Antille, N.; Arsever, S.; et al. 2015. Reconstruction and simulation of neocortical microcircuitry. *Cell*, 163(2): 456–492.

Morgan, J. L.; Berger, D. R.; Wetzel, A. W.; and Lichtman, J. W. 2016. The fuzzy logic of network connectivity in mouse visual thalamus. *Cell*, 165(1): 192–206.

Peng, H.; Hawrylycz, M.; Roskams, J.; Hill, S.; Spruston, N.; Meijering, E.; and Ascoli, G. A. 2015. BigNeuron: large-scale 3D neuron reconstruction from optical microscopy images. *Neuron*, 87(2): 252–256.

Qi, C. R.; Yi, L.; Su, H.; and Guibas, L. J. 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Conference on Neural Information Processing Systems (NeurIPS)*, volume 30.

Scheffer, L. K.; Xu, C. S.; Januszewski, M.; Lu, Z.; Takemura, S.-y.; Hayworth, K. J.; Huang, G. B.; Shinomiya, K.; Maitlin-Shepard, J.; Berg, S.; et al. 2020. A connectome and analysis of the adult Drosophila central brain. *Elife*, 9: e57443.

Schubert, P. J.; Dorkenwald, S.; Januszewski, M.; Jain, V.; and Kornfeld, J. 2019. Learning cellular morphology with neural networks. *Nature Communications*, 10(1): 1–12.

Shapson-Coe, A.; Januszewski, M.; Berger, D. R.; Pope, A.; Wu, Y.; Blakely, T.; Schalek, R. L.; Li, P. H.; Wang, S.; Maitin-Shepard, J.; et al. 2021. A connectomic study of a petascale fragment of human cerebral cortex. *BioRxiv*, 2021–05.

Sümbül, U.; Song, S.; McCulloch, K.; Becker, M.; Lin, B.; Sanes, J. R.; Masland, R. H.; and Seung, H. S. 2014. A genetic and computational approach to structurally classify neuronal types. *Nature Communications*, 5(1): 1–12.

Takemura, S.-y.; Bharioke, A.; Lu, Z.; Nern, A.; Vitaladevuni, S.; Rivlin, P. K.; Katz, W. T.; Olbris, D. J.; Plaza, S. M.; Winston, P.; et al. 2013. A visual motion detection circuit suggested by Drosophila connectomics. *Nature*, 500(7461): 175–181.

Thekumparampil, K. K.; Wang, C.; Oh, S.; and Li, L.-J. 2018. Attention-based graph neural network for semi-supervised learning. *arXiv preprint arXiv:1803.03735*.

Udvary, D.; Harth, P.; Macke, J. H.; Hege, H.-C.; de Kock, C. P.; Sakmann, B.; and Oberlaender, M. 2022. The impact of neuron morphology on cortical network architecture. *Cell Reports*, 39(2).

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(11).

Wagner, A.; Regev, A.; and Yosef, N. 2016. Revealing the vectors of cellular identity with single-cell genomics. *Nature Biotechnology*, 34(11): 1145–1160.

Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S. E.; Bronstein, M. M.; and Solomon, J. M. 2019. Dynamic graph cnn for learning on point clouds. *ACM Transactions On Graphics*, 38(5): 1–12.

Zhao, H.; Jiang, L.; Jia, J.; Torr, P. H.; and Koltun, V. 2021. Point transformer. In *IEEE International Conference on Computer Vision (ICCV)*, 16259–16268.

Zhu, T.; Yao, G.; Hu, D.; Xie, C.; Gong, H.; and Li, A. 2022. MorphoGNN: Morphological embedding for single neuron with graph neural networks. *bioRxiv*, 2022–05.