Task-Related Feature Enhancement Network for Neuronal Morphology Classification

Chunli Sun
cls118@mail.ustc.edu.cn
Feng Zhao*
fzhao956@ustc.edu.cn

MoE Key Laboratory of Brain-inspired Intelligent Perception and Cognition, University of Science and Technology of China, Hefei, China

Abstract

Analyzing the morphology of pyramidal cells (PCs) is essential to understanding brain activity and disease mechanisms. Existing methods describe the morphology of PCs in a task-agnostic manner, leading to insufficient representations for various analysis tasks. This paper presents a Task-related Feature Enhancement Network (TFENet) to discern subtle morphological differences and identify the PCs in a task-related manner. The TFENet first extracts task-common features via a shared backbone across tasks and then generates task-specific features for each task individually. The task-specific features are refined using a Region Feature Enhancement Module (RFEM) based on the morphology-aware regions. Furthermore, a Global Category-guided Fusion Module (GCFM) adaptively combines the task-specific and task-common features, yielding a distinctive morphology descriptor. Extensive experiments demonstrate the effectiveness of our method, achieving 90.34% and 74.15% accuracy for the species and brain region analysis tasks, respectively, outperforming the task-agnostic methods.

1 Introduction

Pyramidal cells (PCs), the principal excitation neuron type in the cerebral cortex, make up approximately two-thirds of all neurons in the mammalian cortex [19, 28]. These cells are distributed throughout various brain regions in mammals, including the cerebral cortex, hippocampus, and amygdala [20]. Thus, they are associated with a variety of cognition activities of the brain. To better understand the neural bases of such complex activities, it is essential to understand and analyze the morphologies of PCs.

PCs consist of subtypes distributed in multiple brain regions and possess variable morphology [8, 21, 24]. Moreover, the morphology of PC is also highly variable between species [5, 28]. The morphological difference makes brain region and species analysis of PCs more challenging. Recently, many methods and tools have been proposed to effectively characterize the shape of PCs. Given the tree-like structure of neurons, many efforts focus on computing heuristic measurements based on predefined metrics [3, 27]. These measurements include the total length and width, the order of the branch, and the surface of soma [26]. Many studies design features to accurately represent the morphology of neurons

*Corresponding author.

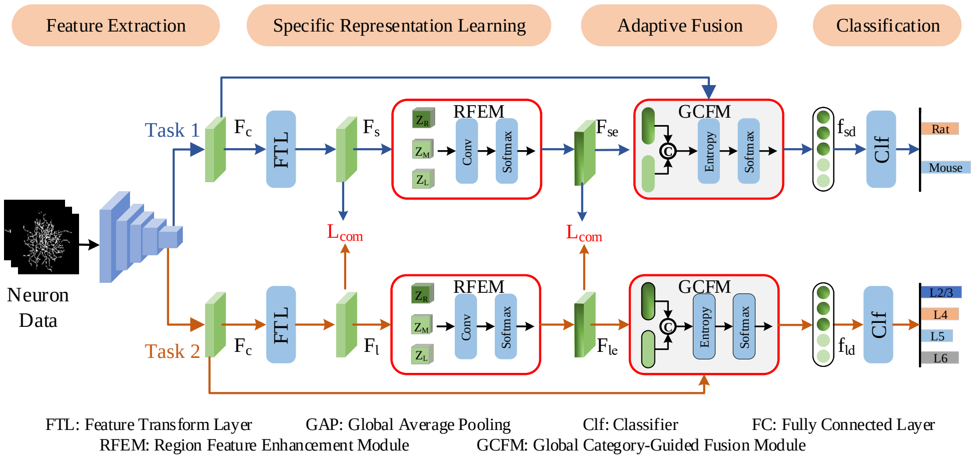
Feature Extraction
Specific Representation Learning
Adaptive Fusion
Classification
Task 1
Task 2
Neuron Data
Fc
FTL
Fs
Fl
RFEM
ZR
ZM
ZL
Conv
Softmax
Lcom
Fse
Fle
GCFM
C
Entropy
fsd
fld
Clf
Rat
Mouse
L2/3
L4
L5
L6
FTL: Feature Transform Layer
GAP: Global Average Pooling
Clf: Classifier
FC: Fully Connected Layer
RFEM: Region Feature Enhancement Module
GCFM: Global Category-Guided Fusion Module

Figure 1: Framework of the proposed Task-related Feature Enhancement Network (TFENet). TFENet introduces the Region Feature Enhancement Module (RFEM) to mine the salient task-specific features and the Global Category-guided Fusion Module (GCFM) to adaptively fuse the task-common (F_c) and task-specific features (F_{se} or F_{le}) to generate a more distinct morphological descriptor (F_{sd} or F_{ld}).

based on the principles of topology and geometry, such as Sholl analysis [18] and topological morphology descriptor [10]. Besides, to achieve a more precise comparison of two or more neuronal structures, some studies decompose the whole neuron tree into multiple segments and then employ heuristic measurements to quantify the difference between neurons [2, 11]. Later, several learning-based techniques emerged that use deep neural networks to precisely describe neuronal morphology and significantly improve the analysis performance [13, 14, 16, 17, 30, 31]. For instance, Li *et al*. [14] leverage a stacked convolutional auto-encoder to learn neuronal morphologies. Additionally, a generative adversarial network is designed to analyze the morphological features of neurons [15]. While achieving improved results, these methods perform the analysis task in a task-agnostic manner, identifying neurons with the same feature representations for all tasks. Consequently, they fail to capture the subtle morphological differences for different analysis tasks or neuronal types. Since neurons exhibit variations in shape, size, and branching pattern, it is crucial to extract fine-grained and exclusive morphological features for the species and brain region analysis to fully capture these subtle distinctions.

In this paper, we propose a Task-related Feature Enhancement Network (TFENet) that effectively describes the morphology of PCs in a task-related manner and performs multiple analysis tasks of PCs simultaneously. As shown in our previous work [25], there are common features for species and brain region analysis tasks. Furthermore, the specific features should be explored and enhanced to improve the performance of different tasks. Specifically, we first obtain the task-common feature using a shared backbone and then capture the task-specific features for each analysis task. Moreover, we introduce a Region Feature Enhancement Module (RFEM) to mine the salient features from the key morphology-aware regions to capture the distinctive task-specific features. To fully characterize the morphology of PCs, we design a Global Category-guided Fusion Module (GCFM) to determine the importance of the task-common and task-specific features for the final decision and adaptively fuse them to generate a distinguishing morphology descriptor. Experimental results demonstrate that our TFENet can effectively characterize the subtle morphological difference for PCs and improve the performance of the species and brain region analysis tasks.

The main contributions of our method can be outlined as follows:

1. We propose a TFENet to perform fine-grained classification of PCs in a task-related manner. TFENet explicitly extracts fine-grained morphology features and conducts multiple analysis tasks simultaneously.

2. We introduce an RFEM to capture salient task-specific features from the morphology-aware regions, and a GCFM to learn the importance of the task-common and task-specific features and adaptively fuse them into a more discriminative descriptor.

3. Our method achieves promising analysis performance in identifying PCs according to brain layer and species. Notable, we achieve an overall accuracy of 90.34% when identifying the species of PCs.

2 Methodology

As illustrated in Fig. 1, the TFENet mainly includes feature extraction, specific representation learning, feature fusion, and classification modules. It extracts morphological features in a task-related manner and performs the analysis on the species and brain region to which the neuron belongs simultaneously. Specifically, the TFENet explicitly learns the task-common features (F_c) using the shared backbone from the neuron data. Since handling 3D data directly in a unified network is challenging, we project the 3D neuron data into three 2D view images like in [14, 15, 25]. These view images are concatenated along the channel and fed into the TFENet to extract the features. Then, the distinguishing specific representations for different tasks are captured through the introduced RFEM (shown in Fig. 2). Finally, to generate a distinct feature descriptor, our TFENet combines the task-common and task-specific features adaptively through GCFM (shown in Fig. 3). In this way, the acquired morphological features contribute to the final decision-making. Finally, the fused feature descriptor is fed into the classifier to predict the species or brain regions of PCs.

2.1 Region Feature Enhancement Module (RFEM)

We observe that most regions of the neuron image are background, and only a few pixel areas contain neuron trees. Therefore, we introduce the RFEM to learn the morphological characteristics of neurons efficiently from the morphology-aware regions and eliminate the influence of background.

Specifically, we first utilize a Feature Transform Layer (FTL), consisting of two convolutional layers with a kernel size of 3×3 and a batch normalization layer to obtain the basic task-specific features (F_s and F_l for different tasks). Then, the RFEM is introduced to identify and focus on salient information within a feature map. Previous research [22, 29] has proved that in CNNs, various feature channels are associated with distinct areas and parts of an image. Consequently, we perform morphology-aware region feature enhancement based on channel attention. Drawing inspiration from [12], the RFEM divides the feature map F_s into $2n$ sub-feature maps ($Z = \{Z_1, Z_2, \cdots, Z_{2n}\}$) along the channels. As shown in Fig. 2,

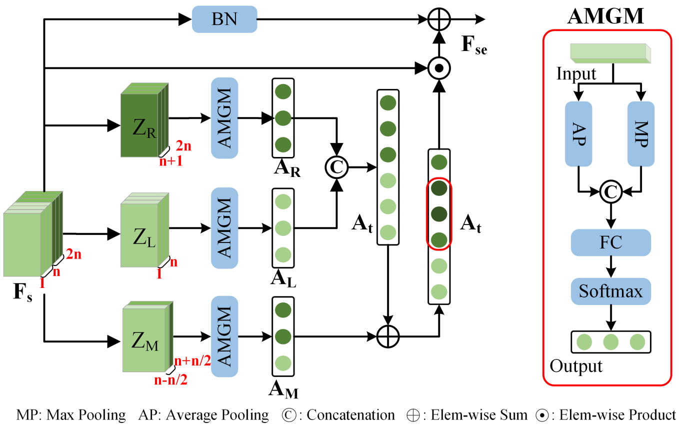

Figure 2: The framework of Region Feature Enhance Module (RFEM). It first divides the input (F_s) into three groups. Then, it computes attention maps using the Attention Map Generation Module (AMGM) for each group and obtains the attention map (A_t). Finally, the enhanced task-specific feature map (F_{se}) is acquired by assigning the A_t to the input F_s.

these sub-feature maps are further divided into three groups, namely $Z_L = \{Z_1, Z_2, \cdots, Z_n\}$, $Z_R = \{Z_{n+1}, Z_{n+2}, \cdots, Z_{2n}\}$, and $Z_M = \left\{Z_{n-\frac{n}{2}}, Z_{n-\frac{n}{2}+1}, \cdots, Z_{n+\frac{n}{2}}\right\}$. Note that $\frac{n}{2}$ is rounded down. Then, the Attention Map Generation Module (AMGM) is employed to compute the attention map. Specifically, a max pooling layer and an average pooling layer are first employed to obtain global features. Subsequently, their outputs are concatenated along the channel. Next, we apply a fully connected layer followed by a softmax layer to compute the attention maps as,

$$A_i = softmax(fc([max(Z_i); avg(Z_i)])). \tag{1}$$

where $i \in \{L, R, M\}$ and $[\cdot]$ is the concatenation operation. The f_c is employed to reduce the dimension of its output to be the same as F_s. Given that the critical morphological region of the neuron tree is generally located in the center, we emphasize the information of the A_M to generate the final attention map A_t as follows:

$$\begin{aligned} &A_t = [A_L; A_R], \\ &A_t[Z_{n-\frac{n}{2}}, Z_{n-\frac{n}{2}+1}, \cdots, Z_{n+\frac{n}{2}}] += A_M. \end{aligned} \tag{2}$$

Finally, the enhanced task-specific feature map F_{se} is acquired by assigning the weights A_t to the feature map F_s as:

$$F_{se} = A_t \odot F_s + BN(F_s), \tag{3}$$

where BN is the batch normalization. By leveraging the broadcasting mechanism, the A_t is first expanded along the last two dimensions and then element-wise multiplied with F_s. The obtained F_{se} can capture the salient features from morphology-aware regions and would be enhanced task-specific features for the species analysis task. Similarly, we can obtain the enhanced task-specific feature F_{le} for the brain region analysis task.

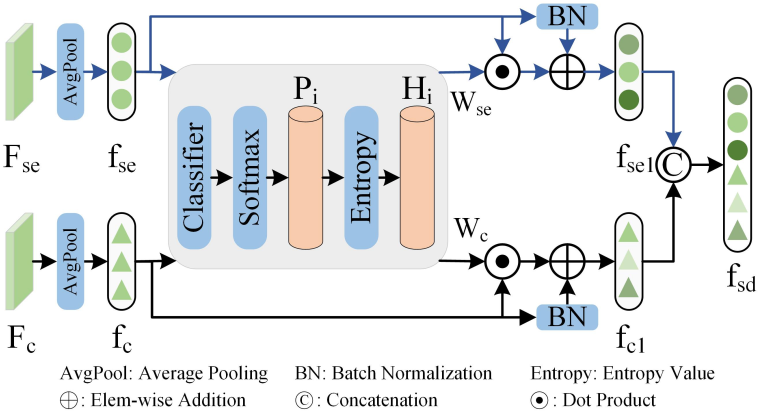

Figure 3: Structure of Global Category-guided Fusion Module (GCFM). GCFM first computes the weights (W_{se} and W_c) of different samples based on the global task-specific (f_{se}) and task-common features (f_c). Consequently, the weighted features (f_{se1} and f_{c1}) are obtained by assigning the weights to the global features. Finally, the weighted features (f_{se1} and f_{c1}) are further fused to generate a more distinguishing descriptor (f_{sd}).

2.2 Global Category-guided Fusion Module (GCFM)

To improve the performance of our method for accurate representation of the morphology of PCs, we combine the task-common and task-specific features to generate a distinguishing morphological descriptor for morphology analysis tasks. This module aims to amplify the effect of task-specific features while preserving the beneficial influence of task-common features on the final decision-making.

For the species analysis task, to better utilize task-common and task-specific features, we predict the weight W_{se} and W_c for F_{se} and F_c through GCFM and then reflect this for the final morphology descriptor. To make accurate predictions, we generate a distinguishing descriptor using the sample importance measured by task-common and task-specific features. Specifically, we first extract the global features (f_{se} and f_c) by employing average pooling on F_{se} and F_c, respectively.

Subsequently, for the f_{se}, the prediction output is generated through a classifier (G) and a softmax layer, denoted as $P_i(x) = softmax(G(x))$. The G comprises two fully connected layers and the output of these fully connected layers is 200 and the total number of classes (C_{se}) relevant to the task, respectively. Consequently, P_i represents a one-dimensional vector with a length equivalent to C_{se}. Following this, we obtain the weight (H_i) of each sample based on the task-specific features by calculating the entropy of P_i as:

$$H_i = \kappa\left(-\sum_{j}^{C_{se}} P_i log(P_i)\right), \tag{4}$$

where $\kappa(x) = \frac{1}{|x|}$ is a function to convert the estimated entropy to a value, indicating the importance of the sample based on task-specific features. As a result, the set of all H_i computed from f_{se} makes up the weights (W_{se}) allocated to all samples for species analysis. Note that the entropy of the prediction output is a probability across several classes, representing the recognition confidence of the input. A greater output confidence level denotes that the prediction for a sample is more accurate. Hence, we attribute this importance to the weighted descriptor f_{se1} by assigning the weight to the input as,

$$\begin{aligned} f_{se1} &= W_{se} \odot f_{se} + BN(f_{se}), \\ W_{se} &= \{H_i\}. \end{aligned} \tag{5}$$

Similarly, we obtain a feature f_{c1} from the task-common feature via the above operations. Finally, we obtain the final morphology descriptor f_{sd} by concatenating f_{se1} and f_{c1}. Besides, we can obtain the fused features f_{ld} for the brain region analysis task.

2.3 Training Loss

For each analysis task, we utilize the cross-entropy loss ($\mathcal{L}_{ce}$) computed from the prediction and true label as the classification loss to train the proposed TFENet. Besides, we use the cosine similarity ($\mathcal{L}_{com}$) to measure the similarities between different task-specific features. The overall loss function is defined as,

$$\begin{aligned} \mathcal{L}_{total} &= \sum_{i=1}^{2} \mathcal{L}_{cei} + \lambda(\mathcal{L}_{com}(F_{se}, F_{le}) + \mathcal{L}_{com}(F_s, F_l)), \\ \mathcal{L}_{com}(f_1, f_2) &= max(\frac{f_1 \cdot f_2}{\|f_1\|_2 \cdot \|f_2\|_2}, 0). \end{aligned} \tag{6}$$

3 Experimental Results

3.1 Datasets and Evaluation Metrics

Dataset: To assess the effectiveness of our method, we randomly download 5951 PCs from the public NeuroMorpho.org dataset [1]. These cells originate from the somatosensory neocortex and medial prefrontal neocortex of rats and mice, located in L2/3, L4, L5, and L6. For the species analysis task, this dataset comprises 2,630 rat cells and 3,321 mouse cells. For the brain region analysis task, there are 2,350 L2/3 cells, 723 L4 cells, 2,261 L5 cells, and 617 L6 cells. For clarity, we refer to these two tasks as Task 1 and Task 2, respectively.

Evaluation metrics: We assess our approach using overall accuracy (OA), average accuracy (AA), and F_1-score. The values of these evaluation metrics range from 0 to 1. The larger the value, the better the performance of our method. To enhance clarity, we convert these values to percentages.

3.2 Implementation Details

We implement TFENet using the PyTorch framework. The TFENet adopts ResNet50 pretrained on ImageNet [9] as the backbone and is trained on a single 3090 GPU. We set n as 2 in RFEM and the coefficient λ of the loss function as 0.8. We utilize the Adam optimizer with a learning rate of 0.001 over 80 epochs and set the batch size to 16. Besides, we employ 10-fold cross-validation to train our TFENet. Specifically, the dataset is divided into 10 small data sets, where the 9 small data sets serve as the training dataset and the remaining one small dataset serves as the test dataset. During training and testing, the training set and its corresponding test set are selected in turn. Accordingly, we conduct ten validation experiments and report the average results of these ten experiments.

Table 1: Comparison of different methods. The best results are highlighted in bold.

Method	Feature	Task 1			Task 2		
		F_1	OA	AA	F_1	OA	AA
LCCDNN [16]	Task-agnostic	86.14 ± 0.68	86.57 ± 0.44	84.27 ± 1.70	67.62 ± 1.43	67.06 ± 0.79	62.60 ± 2.69
DRNN [17]	Task-agnostic	71.94 ± 5.07	74.12 ± 3.27	69.78 ± 4.74	63.05 ± 7.32	63.33 ± 9.72	58.12 ± 1.59
TreeMoCo [4]	Task-agnostic	85.28 ± 3.10	85.58 ± 2.90	82.27 ± 4.17	70.56 ± 2.49	71.88 ± 2.22	61.72 ± 2.88
Baseline	Task-agnostic	85.24 ± 0.46	85.22 ± 0.48	85.13 ± 0.29	67.85 ± 0.43	68.60 ± 0.12	58.78 ± 0.79
Sun *et al.* [25]	Task-related	87.48 ± 0.49	87.44 ± 0.48	87.60 ± 0.52	72.28 ± 0.59	72.46 ± 0.60	**69.81 ± 0.23**
TFENet (Ours)	Task-related	**90.32 ± 0.79**	**90.34 ± 0.81**	**89.99 ± 0.72**	**73.69 ± 0.52**	**74.15 ± 0.57**	69.59 ± 0.69

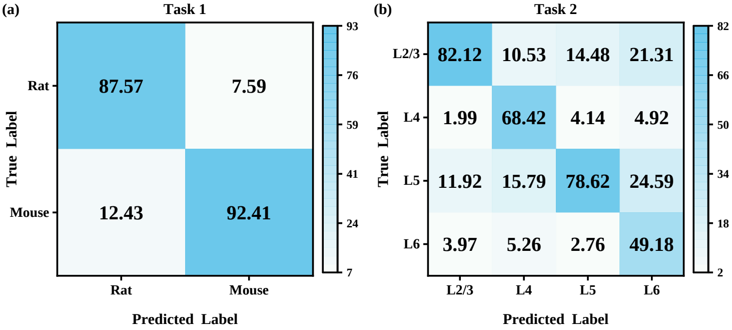

Figure 4: Confusion metrics of our method on Task 1 (a) and Task 2 (b).

3.3 Performance Evaluation

We compare our method with the task-agnostic analysis methods. Besides, we make a comparison with the method that performs the analysis task in a single-task manner without considering the task-common and task-specific features, denoted as baseline. As shown in Table 1, compared to the baseline, our method makes an improvement of 5.08% and 5.12 in F_1 and OA for the species analysis task. Moreover, our method yields gains of 5.84% and 5.55% in F_1 and OA when analyzing the brain region of PCs. Furthermore, our method has a relatively large gain in F_1 and OA compared to the task-related method [25]. This is because our method considers both task-common and task-specific features simultaneously. These results demonstrate that our method can extract subtle morphological differences of species or brain layers of PCs and effectively characterize them in a task-related manner. Although the model complexity of our method is relatively larger than that of the comparison methods [16, 17], our method is able to learn richer features and achieve better performance. Overall, our method significantly enhances morphology feature learning, improving the performance of these two analysis tasks.

In addition, we utilize the confusion matrix to explicitly demonstrate the effectiveness of our method (as shown in Fig. 4). As illustrated in Fig. 4(a), our TFENet accurately identifies the rat and mouse PCs with a high OA of 87.57% and 92.41%, respectively. Moreover, our approach also achieves satisfactory results when performing the brain region analysis task. Note that our method only yields an OA of only 49.38% for identifying L6 PCs. This is because there are only 617 L6 PCs compared to 2,350 L2/3 PCs. Besides, the sparsity of neuron data may introduce noise during model training. In certain cases, particularly when learning categories with few examples, our model might not learn the fine morphological details needed to identify these categories correctly. We will enhance the capacity of our model through data augmentation and model simplification in future work.

Table 2: Performance of the proposed TFENet with different modules.

RFEM	GCFM	Task 1		Task 2	
		F_1	OA	F_1	OA
✗	✗	86.52	86.47	69.41	69.32
✓	✗	88.64	88.65	72.17	72.46
✗	✓	88.43	88.41	70.97	71.38
✓	✓	90.32	90.34	73.69	74.15

Table 3: Performance of the proposed TFENet with different backbones.

Backbone	Task 1		Task 2	
	F_1	OA	F_1	OA
VGG16	80.65	80.68	54.78	54.11
ResNet18	88.66	88.65	72.36	72.71
ResNet50	90.32	90.34	74.16	73.69
MobileNet	66.90	66.91	48.79	41.29

3.4 Ablation Studies

Influence of different modules: Our TFENet enhances task-specific features through RFEM and adaptively fuses task-common and task-specific features through GCFM to obtain more distinguishing descriptors. Here, we assess the impact of RFEM and GCFM on the TFENet.

As shown in Table 2, if we do not use the RFEM and GCFM, our approach only yields an OA of 86.47% and 69.32% on Task 1 and Task 2, respectively. When we only utilize RFEM to enhance the task-specific features, our method achieves OA of 88.65% and 72.46% on Task 1 and Task 2, respectively. Besides, when we only introduce GCFM to fuse the task-common and task-specific features, our method yields OA of 88.41% and 71.38% on Task 1 and Task 2, respectively, which are higher than that without two modules. By utilizing both RFEM and GCFM together, our method significantly improves the performance of the species and brain region analysis tasks. This proves that the utilization of both RFEM and GCFM is advantageous in extracting distinguishing morphology features and enhancing overall performance.

Effect of the number of sub-feature maps in RFEM: In RFEM, we use a parameter n to divide the feature map into multiple sub-feature maps. Here, we investigate the effect of varying n on the performance of our TFENet, and the results are presented in Fig. 5. When the feature map is divided very finely (*e.g.*, n equals 5, 6, and 7), each sub-feature map contains relatively little morphology information, and the connections between branches are ignored. As a result, our approach fails to accurately capture the salient morphological features of PCs and the relationship between branches, resulting in slight performance degradation. Our method achieves satisfactory performance when n is set to 1, 2, or 3. In these cases, each sub-feature map not only contains a certain number of morphological features of PCs but also retains the connection relationship between branches, thereby improving the performance of our method. As shown in Fig. 5, the OA and F_1 are optimal on Task 1 and Task 2 when n is set as 2. Consequently, we set n to 2 in other experiments.

Evaluation on different backbones: Here, we conduct experiments to evaluate our method with different feature extractors, such as MobileNetV3-Small [7], VGG16 [23], and ResNet18 [6]. As shown in Table 3, our method performs best when applying ResNet50 as the backbone, outperforming that based on other backbones. When applying the VGG16

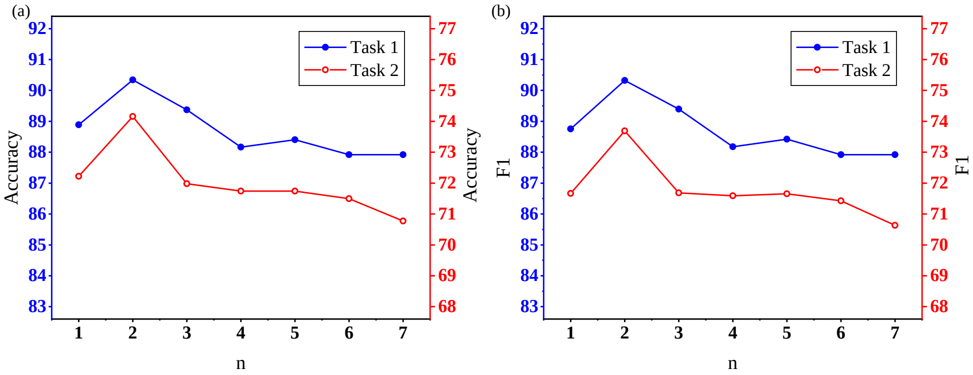

Figure 5: OA (a) and F_1-score (b) of the proposed TFENet with the different number of sub-feature maps.

Table 4: Generalization evaluation of our TFENet evaluated on another dataset.

Method	Feature	Task 1			Task 2		
		F_1	OA	AA	F_1	OA	AA
LCCDNN [16]	Task-agnostic	95.03 ± 1.17	95.34 ± 1.06	86.25 ± 3.13	79.92 ± 7.91	80.22 ± 7.17	79.51 ± 7.78
DRNN [17]	Task-agnostic	91.30 ± 0.23	91.35 ± 0.19	90.63 ± 1.53	82.44 ± 5.75	82.60 ± 5.42	82.07 ± 5.83
Baseline	Task-agnostic	97.46 ± 0.49	97.45 ± 0.48	95.89 ± 0.99	86.82 ± 1.63	86.79 ± 1.66	86.54 ± 1.48
Sun *et al.* [25]	Task-related	98.30 ± 0.42	98.32 ± 0.41	96.72 ± 1.25	88.27 ± 0.94	88.27 ± 0.95	88.31 ± 0.83
TFENet (Ours)	Task-related	**98.88** ± **0.09**	**98.89** ± **0.09**	**97.50** ± **0.08**	**88.84** ± **1.31**	**88.89** ± **1.39**	**88.61** ± **1.41**

and MobileNetV3-Small as the feature extractors, the performance of our method needs to be improved, especially on Task 2 (only with OA of 54.11% and 41.29%, respectively). Employing ResNet as the feature extractor, our approach effectively captures richer morphological features of PCs during the feature extraction phase. This effectively mitigates the limitations imposed by the sparsity of neuronal data to a significant degree. Therefore, our method utilizes the ResNet as the backbone in other experiments.

3.5 Generalization Evaluation

To further demonstrate the effectiveness of our method, we test it on another dataset, which comprises PCs obtained from monkeys and rats in Layer 2/3 and Layer 5. This dataset is randomly downloaded from the public NeuroMorpho.org dataset [1]. Specifically, this new dataset consists of 282 L2/3 PCs and 110 L5 PCs from monkeys, and 2350 L2/3 PCs and 1442 L5 PCs from rats. For the species analysis task (denoted as Task 1), there are 392 PCs from monkeys and 3792 PCs from rats. For the brain region analysis task (denoted as Task 2), there are 2632 L2/3 PCs and 1552 L5 PCs. Besides, we train our method from scratch and do not perform fine-tuning based on the above experiments. The parameters during training are set as described in the Implementation Details section.

As presented in Table 4, our approach performs well in two analysis tasks. Note that our method precisely identifies PCs from the rat and monkey with OA of 98.89% and PCs from different brain layers with OA of 88.89%. Compared with these task-agnostic-based methods [16, 17] and the task-related method [25], our method achieves the best performance both on Task 1 and Task 2. Furthermore, our method, performing two analysis tasks simultaneously can improve the performance of each task, outperforming the baseline, which performs each analytics task individually. These results prove that our method has good generalization and can be applied to more morphological analysis tasks of PCs.

4 Conclusions

In this paper, we propose a Task-related Feature Enhancement Network (TFENet) for pyramidal cell classification, which effectively describes the subtle morphological differences of pyramidal cells in a task-related manner. Our TFENet captures the salient task-specific features via mining and enhancing the key morphology region features through the Region Feature Enhancement Module. Using the Global Category-guided Fusion Module, our method generates distinguishing morphological descriptors by adaptively fusing the task-specific and task-common features based on the guidance of global features. Our TFENet achieves accuracies of 90.34% and 74.15% on the species and brain region analysis tasks, respectively, outperforming the task-agnostic methods. In future work, we will explore the potential of our approach to perform multiple analysis tasks simultaneously and further improve the performance of each analysis task.

Acknowledgments

This work was supported by the Anhui Provincial Natural Science Foundation under Grant 2108085UD12. We acknowledge the support of GPU cluster built by MCC Lab of Information Science and Technology Institution, USTC.

References

[1] Giorgio A Ascoli, Duncan E Donohue, and Maryam Halavi. Neuromorpho. org: a central resource for neuronal morphologies. *J. Neurosci.*, 27(35):9247–9251, 2007.

[2] Saurav Basu, Barry Condron, and Scott T Acton. Path2path: Hierarchical path-based analysis for neuron matching. In *Proc. IEEE Int. Symp. Biomed. Imaging*, pages 996–999, 2011.

[3] Tamal Batabyal, Andrea Vaccari, and Scott T Acton. Neurobfd: Size-independent automated classification of neurons using conditional distributions of morphological features. In *Proc. Int. Symp. Biomed. Imaging*, pages 912–915, 2018.

[4] Hanbo Chen et al. TreeMoCo: Contrastive neuron morphology representation learning. In *Proc. Adv. Neural Inf. Process. Syst.*, volume 35, pages 25060–25073, 2022.

[5] Guy N Elston. Cortex, cognition and the cell: new insights into the pyramidal neuron and prefrontal function. *Cereb. Cortex*, 13(11):1124–1138, 2003.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, pages 770–778, 2016.

[7] Andrew Howard et al. Searching for mobilenetv3. In *Proc. IEEE Int. Conf. Comput. Vision*, pages 1314–1324, 2019.

[8] Bob Jacobs et al. Regional dendritic and spine variation in human cerebral cortex: a quantitative golgi study. *Cereb. Cortex*, 11(6):558–571, 2001.

[9] Deng Jia et al. ImageNet: A large-scale hierarchical image database. In *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, pages 248–255, 2009.

[10] Lida Kanari et al. A topological representation of branching neuronal morphologies. *Neuroinformatics*, 16(1):3–13, 2018.

[11] Reem Khalil, Sadok Kallel, Ahmad Farhat, and Pawel Dlotko. Topological sholl descriptors for neuronal clustering and classification. *PLOS Comput. Biol.*, 18(6): e1010229, 2022.

[12] Minjung Kim, MyeongAh Cho, and Sangyoun Lee. Feature disentanglement learning with switching and aggregation for video-based person re-identification. In *Proc. IEEE Winter Conf. Appl. Comput. Vision*, pages 1603–1612, 2023.

[13] Sophie C Laturnus and Philipp Berens. Morphvae: Generating neural morphologies from 3d-walks using a variational autoencoder with spherical latent space. In *Proc. Int. Conf. Mach. Learn.*, pages 6021–6031, 2021.

[14] Zhongyu Li et al. Large-scale exploration of neuronal morphologies using deep learning and augmented reality. *Neuroinformatics*, 16(3):339–349, 2018.

[15] Zhongyu Li et al. Towards computational analytics of 3d neuron images using deep adversarial learning. *Neurocomputing*, 438:323–333, 2021.

[16] Xianghong Lin and Jianyang Zheng. A neuronal morphology classification approach based on locally cumulative connected deep neural networks. *Appl. Sci.*, 9(18):3876, 2019.

[17] Xianghong Lin, Jianyang Zheng, Xiangwen Wang, and Huifang Ma. A neuronal morphology classification approach based on deep residual neural networks. In *Proc. Int. Conf. Neural Inf. Process.*, pages 336–348, 2018.

[18] José D López-Cabrera, Leonardo A Hernández-Pérez, Rubén Orozco-Morales, and Juan V Lorenzo-Ginori. New morphological features based on the sholl analysis for automatic classification of traced neurons. *J. Neurosci. Methods*, 343:108835, 2020.

[19] Henry Markram et al. Reconstruction and simulation of neocortical microcircuitry. *Cell*, 163(2):456–492, 2015.

[20] Alan Pestronk. Histology of the nervous system of man and vertebrates. *Neurology*, 48 (1):299–300, 1997.

[21] Concepción Rojo et al. Laminar differences in dendritic structure of pyramidal neurons in the juvenile rat somatosensory cortex. *Cereb. Cortex*, 26(6):2811–2822, 2016.

[22] Marcel Simon and Erik Rodner. Neural activation constellations: Unsupervised part model discovery with convolutional networks. In *Proc. IEEE Int. Conf. Comput. Vis.*, pages 1143–1151, 2015.

[23] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[24] Nelson Spruston. Pyramidal neurons: dendritic structure and synaptic integration. *Nat. Rev. Neurosci.*, 9(3):206–221, 2008.

[25] Chunli Sun, Qinghai Guo, Gang Yang, and Feng Zhao. Learning task-specific morphological representation for pyramidal cells via mutual information minimization. In *Predictive Intelligence in Medicine*, pages 134–145, 2023.

[26] Harry BM Uylings and Jaap Van Pelt. Measures for quantifying dendritic arborizations. *Netw.-Comput. Neural Syst.*, 13(3):397–414, 2002.

[27] Xavier Vasques, Laurent Vanel, Guillaume Villette, and Laura Cif. Morphological neuron classification using machine learning. *Front. Neuroanat.*, 10(102):1–12, 2016.

[28] Yun Wang et al. A simplified morphological classification scheme for pyramidal cells in six layers of primary somatosensory cortex of juvenile rats. *IBRO reports*, 5:74–90, 2018.

[29] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579*, 2015.

[30] Tianfang Zhu et al. Morphognn: Morphological embedding for single neuron with graph neural networks. *bioRxiv*, 2022.

[31] Tianfang Zhu et al. Data-driven morphological feature perception of single neuron with graph neural network. *IEEE Trans. Med. Imaging*, 42(10):3069–3079, 2023.