LETTER

# Dendritic Learning-Based Feature Fusion for Deep Networks

**Yaotong SONG**[†], **Zhipeng LIU**[†], **Zhiming ZHANG**[†], **Jun TANG**[††], **Zhenyu LEI**[†], ***Nonmembers***,
***and*** **Shangce GAO**[†a)], ***Member***

**SUMMARY** Deep networks are undergoing rapid development. However, as the depth of networks increases, the issue of how to fuse features from different layers becomes increasingly prominent. To address this challenge, we creatively propose a cross-layer feature fusion module based on neural dendrites, termed dendritic learning-based feature fusion (DFF). Compared to other fusion methods, DFF demonstrates superior biological interpretability due to the nonlinear capabilities of dendritic neurons. By integrating the classic ResNet architecture with DFF, we devise the ResNeFt. Benefiting from the unique structure and nonlinear processing capabilities of dendritic neurons, the fused features of ResNeFt exhibit enhanced representational power. Its effectiveness and superiority have been validated on multiple medical datasets.

***key words:*** *convolutional network, neural networks, dendritic neuron, feature fusion*

## 1. Introduction

Image classification serves as a crucial cornerstone in the domain of computer vision and holds significant importance in various applications including but not limited to object recognition, object detection, image retrieval, image quality assessment, and visual search engines [1]. It plays a fundamental role in comprehending image data, promoting innovation and progress in multiple fields. With the advent of large-scale datasets and the improvement in computational power, convolutional neural networks (CNNs) utilize a multilayer structure of convolutional and pooling operations to effectively capture both local and global features of images. This process gradually extracts more abstract and high-level features across different layers. The multilayer structure enables CNNs to provide a layered representation of images, thereby better capturing the information within images and enhancing the accuracy of image classification tasks.

Deep CNN remains the problems of gradient disappearance and network degradation. ResNet [2] effectively addresses these problems by introducing the residual connections. This innovation enables it can train deeper network structures while achieving remarkable performance. Despite recent emergence of numerous large-scale models such as ViT [3]. ResNet still has the advantages of easy implementation. This grants ResNet scalability and efficiency in practical applications. Due to its outstanding performance and simple network architecture, ResNet continues to be widely adopted.

Traditional ResNet models do not fully exploit the utilization of features. To address this issue, researchers have proposed various ResNet-based variants, such as ResNeXt [4], Res2Net [5], and ResNetSt [6], which have achieved state-of-the-art performance at that time. These improved variants of ResNet adopt different approaches to fully extract features at each block. However, these methods primarily focus on feature acquisition at individual network block layers, often lacking sufficient utilization of multi-level features.

Several methods have been proposed for fusion inter-level features. For example, Residual Steps Network (RSN) [7] maintains spatial information in a high-resolution sub-network while gradually incorporating semantic information from low-resolution sub-networksaggregates. In FFA-Net [8], it built upon attention mechanism for integrating features at different levels, the feature attention module, assigning higher weights to important features. This structure facilitates the preservation of shallow-level information and its propagation to deeper layers. These methods primarily focus on the feature representation of the last layer of the model, but these signs do not prove that the last layer serves as the ultimate representation for any task. In fact, the fusion of features across different layers has become a focal point of researchers' attention. In 2018, Fisher Yu et al. introduced the DLA [9], which successfully integrates features from different layers of the network, achieving deep aggregation of semantic and spatial information, thereby comprehensively capturing feature information. This study compellingly demonstrates that, like the width and depth of a network, feature fusion is also an important dimension in network architecture. Although the DLA achieves significant performance improvements, we note that its fusion approach still lacks in terms of biological interpretability.

Biologically inspired neurons play a crucial role in shaping neural networks. Recently, a biological neuron's approach to nonlinear feature processing, known as the dendritic neuron model (DNM) [10]. By using the characteristics of DNM's multiple dendrites, the network can more comprehensively utilize interactions between features from different layers, thereby enhancing the network's performance

Manuscript received March 12, 2024.
Manuscript revised May 14, 2024.
Manuscript publicized August 7, 2024.

[†] Faculty of Engineering, University of Toyama, Toyama-shi, 930–8555 Japan.
[††] Wicresoft Co Ltd, 13810 SE Eastgate Way, Bellevue, WA 98005, USA.
a) E-mail: gaosc@eng.u-toyama.ac.jp

DOI: 10.1587/transinf.2024EDL8021

and representational capacity. Moreover, this approach better emulates the feature processing mechanism of biological neurons.

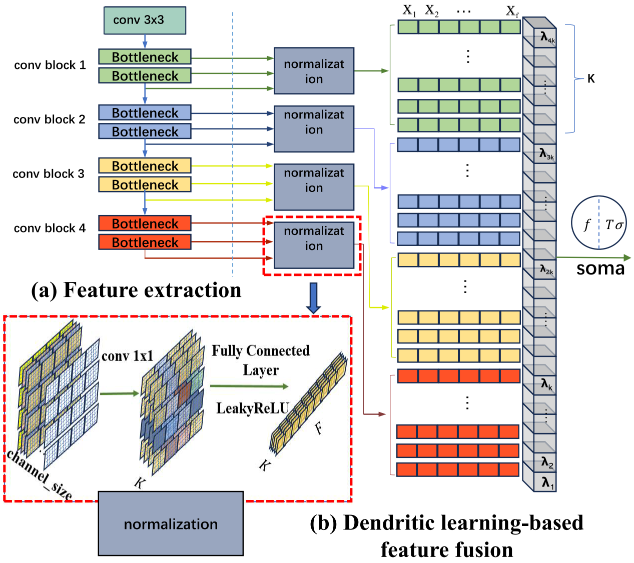
Motivated by the aforementioned discussions, this letter presents ResNeFt, a novel approach that extends the ResNet structure by incorporating the structural characteristics of dendritic neurons for feature fusion. The main contributions of this work are as follows: 1) Aggregating inter-level features to enhance the reuse of cross-level features in the network. 2) Introducing the structure of dendritic neurons and leveraging their nonlinear processing capabilities to improve the feature fusion. 3) Experimental evaluation of the proposed ResNeFt on multiple datasets within the MedMnist benchmark to validate its effectiveness and superiority.

## 2. Methodology

ResNeFt consists of two components: feature extraction and dendritic learning-based feature fusion (DFF). In the process of DFF, ResNeFt draws inspiration from the structure of dendritic neurons, enabling the network to more precisely balance the weights of different hierarchical features during fusion. This design not only enhances network performance but also brings the entire network architecture closer to real biological neurons.

### 2.1 Feature Extraction

In our work, we employ ResNet as the basic network for feature extraction. ResNet is primarily composed of four convolutional blocks. Each convolutional block consists of multiple bottlenecks. The Bottleneck structure constitutes the core component of the ResNet network and is responsible for the feature extraction process in images. The feature extraction is depicted in the left half of the dashed line of Fig. 1 (a). To ensure that each extracted feature map directly reflects the feature extraction results at the current depth of the network block. We choose to extract feature maps from each bottleneck without residual connections, as well as from the feature maps at the current depth of the convolutional block for feature fusion. Through meticulously designed feature extraction, we hope that the extracted features can reflect the information captured by the current depth module. This process provides a solid foundation for subsequent DFF processes, allowing for a comprehensive integration of the captured information.



(a) Feature extraction

(b) Dendritic learning-based feature fusion

**Fig. 1** The structure of ResNeFt.

### 2.2 Dendritic Learning-Based Feature Fusion

Dendritic neurons consist of dendritic structures and soma body. The dendritic structure receives and processes feature signals, while the soma body aggregates the computational results from various dendrites and outputs the integrated result. This structure enables dendritic neurons to efficiently process complex feature information and generate precise outputs. Feature maps extracted at different depths of the network exhibit diverse sizes and channel dimensions. Consequently, normalization is necessary to standardize both the feature size and channel dimensions. As show in Fig. 1 normalization, we use a 1 × 1 convolutional kernel with a stride of 1 to aggregate spatial information, while simultaneously unifying inter-group features to the same channel number $K$ ($K$ = 10). The features extracted from images by the network have a two-dimensional structure. In our process, each two-dimensional feature is mapped to a corresponding one-dimensional feature of size $F$ ($F = 512$) and activate using the LeakyReLU. Unlike the ReLU activation function, LeakyReLU doesn't set the value to 0 when $x < 0$; instead, it replaces the value with a small numeric. LeakyReLU provides non-linear mapping and offers advantages over ReLU by avoiding the issue of "neuron death". The formula for the normalization process as follows:

$$\begin{aligned} Y[K, w, h] &= \sum_{c}\sum_{w}\sum_{h} X[c, w, h] \times W_1[K, c, 1, 1], \\ T[K, F] &= Y[K, wh] \times W_2[wh, F], \end{aligned} \tag{1}$$

where $X$, $Y$ and $T$ represents the input, result after convolution and normal, $c$ represents the number of channels, while $w$ and $h$ respectively denote the width and height of the feature map. $W_1$ and $W_2$ represents the parameters of the convolutional kernel and the linear. The $[c, w, h]$ denotes that the shape of the input data matrix $X$, and the $X \times W$ represents the multiplication between the input matrix and the weight matrix. Ultimately, we have obtained $4K$ one-dimensional features, each length of $F$.

Then we utilize the structural framework of dendritic neural for the process of feature fusion. Its inherent nonlinear characteristics allow for the processing of more complex information, surpassing the linear integration capabilities of traditional neuron models. Furthermore, the parallel processing of multiple dendritic branches enhances computational performance. The structure of dendritic learning-based feature fusion is illustrated in Fig. 1. These features

**Table 1** Accuracy and F1-score for four MedMNIST datasets.

| Methods | PathMNIST | | | BloodMNIST | | | DermaMNIST | | | RetinaMNIST | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Acc(%) | F1(%) | T(s) | Acc(%) | F1(%) | T(s) | Acc(%) | F1(%) | T(s) | Acc(%) | F1(%) | T(s) |
| DD | 64.64±1.54 | 59.28±2.02 | 1978 | 80.87±0.25 | 78.54±0.33 | 223 | 69.26±0.72 | 28.69±6.82 | 127 | 52.88±1.63 | 33.53±0.91 | 39 |
| BLS | 60.6±0.15 | 49.4±3.14 | 2490 | 85.97±0.03 | 84.06±0.12 | 305 | 72.45±0.03 | 35.46±0.02 | 166 | 54.37±0.13 | 32.38±0.69 | 26 |
| Vit | 73.53±0.3 | 66.9±0.2 | 2099 | 88.38±0.01 | 86.88±0.11 | 239 | 71.52±0.05 | 40.08±0.7 | 141 | 54.37±0.37 | 34.85±1.34 | 24 |
| MobileNetV3 | 79.45±0.15 | 74.0±0.42 | 3387 | 90.0±0.3 | 88.68±0.15 | 439 | 70.3±0.23 | 37.11±1.8 | 272 | 50.75±0.0 | 27.62±2.77 | 39 |
| ShuffleNetV2 | 87.0±0.27 | 81.84±0.62 | 4218 | 93.19±0.15 | 92.13±0.22 | 546 | 72.02±0.05 | 41.43±4.84 | 289 | 50.37±0.37 | 28.26±3.11 | 55 |
| ResNet | 91.6±0.18 | 87.59±0.24 | 11573 | 95.56±0.06 | 95.14±0.1 | 1654 | 74.78±0.08 | 50.84±0.34 | 973 | 53.62±1.12 | 33.78±1.16 | 111 |
| ResNeFt | **91.8±0.37** | **88.24±0.7** | 18087 | **96.88±0.09** | **96.61±0.09** | 2496 | **76.48±0.34** | **54.48±0.89** | 1321 | **54.88±0.52** | **37.0±2.18** | 220 |

are individually assigned to $4K$ dendrites for further processing. The implementation of dendritic learning is formulated as

$$O_j = \lambda_j \sum_{i=1}^{F} (W_i \cdot X_i + b_i), \tag{2}$$

where $O_j$ is the individual output of a dendrite, $\lambda_j$ is the adaptive weight, $X_i$ represents the $i$-th input feature, $W_i$ corresponds to its weight, and $b_i$ represents its bias. The computed output from dendrite are input into the soma body for the final classification.

In the soma body, the computational outputs from $4K$ dendrites are summed along the dimensions of dendrites. This aggregation process mimics the information integration function of dendrites in biological neurons, it can effectively fuse the feature information carried by different dendrites. Subsequently, we map the aggregated results to the final classification head, thus achieving the transformation from feature space to classification space. Due to the intrinsic characteristics of dendritic structure, the information processed on dendrites typically exhibits lower numerical levels. Therefore, directly applying a sigmoid function as the activation function is not beneficial for the ResNeFt's training. To enhance the ultimate classification results, we employ a modified sigmoid function as the activation function in the soma body, referred to as "$T\sigma$" [11]. The sigmoid function and its derivative are as follows:

$$\sigma(x) = \frac{1}{1 + e^{-x}}. \tag{3}$$

When the input data is small and near zero, it is evident that the derivative of the sigmoid function does not equal to $x$. $T\sigma$ functixon and its derivative are as follows:

$$T\sigma(x) = 4\sigma(x) - 2. \tag{4}$$

The reciprocal of the $T\sigma$ function at the point 0 is equal to $x$. This property allows for better preservation of the feature information.

## 3. Experimental Results

To demonstrate the effectiveness of our proposed method, We conduct an extensive comparison of network architectures, encompassing classic CNNs such as ResNet [2], MobileNet [12], and ShuffleNet [13]. Additionally, to explore performance differences among different network architectures, we employ the Vision Transformer (ViT) [3] based on the transformer architecture, the Dendrite Net (DD) [14] utilizing dendrite neuron structures and the Broad Learning System (BLS) [15], which without deep architecture.

**Table 2** The composition for four MedMNIST datasets.

| Dataset | Data Modality | Samples | Classes | Training / Validation / Test | Test Proportion(%) |
|---|---|---|---|---|---|
| PathMNIST | Colon Pathology | 107,180 | 9 | 89,996 / 10,004 / 7,180 | 6.66 |
| BloodMNIST | Blood Cell Microscope | 17,092 | 8 | 11,959 / 1,712 / 3,421 | 28.60 |
| DermaMNIST | Dermatoscope | 10,015 | 7 | 7,007 / 1,003 / 2,005 | 28.61 |
| RetinaMNIST | Fundus Camera | 1,600 | 5 | 1,080 / 120 / 400 | 25.00 |

The experiments are conducted on multiple datasets from MedMnist. MedMNIST is a publicly available dataset serving as a benchmark for machine learning and deep learning algorithms in the field of medical imaging. The purpose of MedMNIST is to provide a standardized and accessible dataset for researchers and practitioners in the medical image analysis domain. In our experiments, we select four color image datasets: PathMNIST, BloodMNIST, DermaMNIST, and RetinaMNIST. These datasets encompass four types of biomedical images and the image size is set to 28 × 28. The detailed composition of the datasets is shown in Table 2.

All experiments are conducted on an NVIDIA RTX 3090 GPU. All models train for 100 epochs and the batch size is set to 128. Adam optimizer is utilized with a learning rate (lr) of 1e-3. In ResNeFt, the lr for the Base Network is set to 1e-3, while the lr for the DFF module is set to 1e-5. To augment the data, we solely employed fundamental normalization. We adopted the cross-entropy loss function as the employed loss function.

We employ accuracy (Acc) and f1-score (F1) as evaluation metrics to assess each model's performance. Acc provides a direct measure of model performance, while F1-score considers both false positives and false negatives, offering a more comprehensive evaluation of classification performance. All models are trained using the training data set, and their performance is assessed using the validation set. Subsequently, the final model performance is examined on the test data set.

To mitigate the impact of randomness on the experiments and validate the stability of the proposed method, we perform five independent replicate experiments for all models. The results of all models on the test set are presented in Table 1. We show the mean, standard deviation and the average time (T) for training the models. It is apparent that the proposed ResNeft need more training time, potentially attributed to the DFF module in feature fusion. However, ResNeFt exhibits exceptional performance across all datasets, unequivocally demonstrating its superiority. Importantly, this outcome validates the effectiveness of the DFF

**Table 3** Discussion on the parameters $K$ and $F$ in DermaMNIST.

| K \ F | 64 | | 128 | | 512 | | 1024 | |
|---|---|---|---|---|---|---|---|---|
| | Acc | F1 | Acc | F1 | Acc | F1 | Acc | F1 |
| 5 | 0.7696 | 0.5408 | 0.7631 | 0.5436 | 0.7701 | 0.5495 | 0.7706 | 0.5530 |
| 10 | 0.7521 | 0.5217 | 0.7626 | 0.5734 | **0.7706** | **0.5584** | 0.7591 | 0.5288 |
| 15 | 0.7631 | 0.5331 | 0.7581 | 0.5240 | 0.7546 | 0.5108 | 0.7656 | 0.5614 |
| 20 | 0.7551 | 0.4991 | 0.7471 | 0.4516 | 0.7606 | 0.5546 | 0.7646 | 0.5378 |

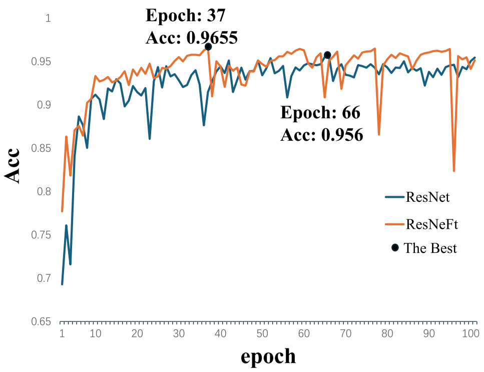


**Fig. 2** The average convergence curves of ResNet and ResNeFt on BloodMNIST.

method. This approach adeptly feature fusion from different network layers, significantly enhancing the representational capacity of features.

To better explore the performance of ResNeFt, we discuss the effect of hyperparameters $K$ and $F$ for our model, and the results of Acc and F1 are shown in Table 3. The table represents the parameter range of $F = \{64, 128, 512, 1024\}$ along the horizontal axis, and the parameter range of $K = \{5, 10, 15, 20\}$ along the vertical axis, it can be observed that the model achieves the best performance when $K = 10$ and $F = 512$. Based on our preliminary analysis, we speculate that the sizes of the hyperparameters $K$ and $F$ are related to the number and dimensions of the selected feature maps.

The dendritic structure can effectively adjust the features at different layers of the deep network, optimizing the parameters of various model components more efficiently. In Fig. 2, we show the average convergence curves of ResNeFt and ResNeFt on BloodMNIST. Evidently, the proposed with DFF enhances the convergence speed and performance of the ResNet, thereby reducing the training time required to achieve optimal performance.

## 4. Conclusions

This letter addresses the issue of feature fusion at different layers and introduces a novel model called ResNeFt. The DFF module leverages the dendritic structure to fuse features across various depths of the network. This enables thorough feature utilization, accelerates the convergence speed of neural networks, and enhances biological interpretability by closely resembling the real dendritic neural structure of the human brain. Experimental results on multiple medical datasets demonstrate that ResNeFt, compared to other advanced networks, exhibits more effective performance. In future research, we hope to apply the DFF to a broader range of models, thereby universally enhancing the performance of other neural networks.

## Acknowledgments

This research was partially supported by the Japan Society for the Promotion of Science (JSPS) KAKENHI under Grant 23K24899.

## References

[1] W. Wang, Y. Yang, X. Wang, W. Wang, and J. Li, "Development of convolutional neural network and its application in image classification: A survey," Optical Engineering, vol.58, no.4, 040901, 2019.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," IEEE Conf. Comput. Vis. Pattern Recognit (CVPR), pp.770–778, 2016.

[3] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," Int. Conf. Learn. Represent (ICLR), 2021.

[4] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," IEEE Conf. Comput. Vis. Pattern Recognit (CVPR), pp. 5987–5995, 2017.

[5] S.-H. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, and P. Torr, "Res2Net: A new multi-scale backbone architecture," IEEE Trans. Pattern Anal. Mach. Intell., vol.43, no.2, pp.652–662, 2019.

[6] H. Zhang, C. Wu, Z. Zhang, Y. Zhu, H. Lin, Z. Zhang, Y. Sun, T. He, J. Mueller, R. Manmatha, M. Li, and A. Smola, "ResNeSt: Split-attention networks," IEEE Conf. Comput. Vis. Pattern Recognit (CVPR), pp.2735-2745, 2022.

[7] Y. Cai, Z. Wang, Z. Luo, B. Yin, A. Du, H. Wang, X. Zhang, X. Zhou, E. Zhou, and J. Sun, "Learning delicate local representations for multi-person pose estimation," Eur. Conf. Comput. Vis (ECCV), pp.455–472, Springer, 2020.

[8] X. Qin, Z. Wang, Y. Bai, X. Xie, and H. Jia, "FFA-Net: Feature fusion attention network for single image dehazing," AAAI Conf. Artif. Intell., vol.34, no.7, pp.11908–11915, 2020.

[9] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, "Deep layer aggregation," IEEE Conf. Comput. Vis. Pattern Recognit (CVPR), pp.2403–2412, 2018.

[10] S. Gao, M. Zhou, Z. Wang, D. Sugiyama, J. Cheng, J. Wang, and Y. Todo, "Fully complex-valued dendritic neuron model," IEEE Trans. Neural Networks Learn. Syst., vol.34, no.4, pp.2105–2118, 2023.

[11] N. Papernot, A. Thakurta, S. Song, S. Chien, and Ú. Erlingsson, "Tempered sigmoid activations for deep learning with differential privacy," AAAI Conf. Artif. Intell., vol.35, no.10, pp.9312–9321, 2021.

[12] A. Howard, M. Sandler, B. Chen, W. Wang, L.-C. Chen, M. Tan, G. Chu, V. Vasudevan, Y. Zhu, R. Pang, H. Adam, and Q. Le, "Searching for MobileNetV3," IEEE Int. Conf. Comput. Vis (ICCV), pp.1314–1324, 2019.

[13] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "ShuffleNet V2: Practical guidelines for efficient CNN architecture design," Eur. Conf. Comput. Vis. (ECCV), pp.122–138, 2018.

[14] G. Liu and J. Wang, "Dendrite Net: A white-box module for classification, regression, and system Identification," IEEE Trans. Cybern., vol.52, no.12, pp.13774–13787, 2022.

[15] C.L.P. Chen and Z. Liu, "Broad learning system: An effective and efficient incremental learning system without the need for deep architecture," IEEE Trans. Neural Networks Learn. Syst., vol.29, no.1, pp.10–24, 2018.