A Lightweight Multi-Dendritic Pyramidal Neuron Model with Neural Plasticity on Image Recognition

Yu Zhang, Pengxing Cai, Yanan Sun, *Member, IEEE*, Zhiming Zhang, Zhenyu Lei, and Shangce Gao, *Senior Member, IEEE*,

***Abstract*—Simulating the method of neurons in the human brain that process signals is crucial for constructing a neural network with biological interpretability. However, existing deep neural networks simplify the function of a single neuron without considering dendritic plasticity. In this paper, we present a multi-dendrite pyramidal neuron model (MDPN) for image classification, which mimics the multilevel dendritic structure of a nerve cell. Unlike the traditional feedforward network model, MDPN discards premature linear summation integration and employs a nonlinear dendritic computation such that improving the neuroplasticity. To model a lightweight and effective classification system, we emphasized the importance of single neuron and redefined the function of each subcomponent. Experimental results verify the effectiveness and robustness of our proposed MDPN in classifying 16 standardized image datasets with different characteristics. Compared to other state-of-the-art and well-known networks, MDPN is superior in terms of classification accuracy. The MDPN code is available on GitHub: https://github.com/zy-251026/MDPN.**

***Impact Statement*—Most current deep neural networks consist of the basic framework of convolution and multilayer perceptron, which lack biological plasticity. Designing a novel network framework that more closely resembles the neurons of the human brain has been one of the top priorities for artificial intelligence. In this paper, we present a multi-dendritic shallow network architecture called MDPN based on the biological framework of pyramidal neuron to solve the image classification problem. In comparison with current network models, MDPN has the following advantages:**

1) **MDPN mimics the biological structure of a multipolar neuron to model a five-layer lightweight network, including pre-synaptic layer, dendritic layer, membrane layer, soma layer, and axon layer. To reflect the biological plasticity of the model, we propose for the first time a repeated information input as well as multiple nonlinear dendritic computations. This information processing increases the width of the model and solves the problem of diminishing feature reuse.**
2) **Compared to traditional deep neural networks, we emphasize the importance of nonlinear dendritic computation in the process of feature extraction. The repeated dendritic computation mechanism enhances the model's expandability and nonlinearity. Additionally, MDPN proposes, for the first time, an integration function based on multiplication. This integration operation occurs after the activation function is called. The multiplication method can obtain feature information that cannot be obtained by the addition method in normalized data. This is consistent with the results of biological experiments.**

***Index Terms*—Neural network, machine learning, image classification, dendritic plasticity, dendritic computation.**

I. INTRODUCTION

Neurons refer to the basic units of the nervous system in biology that integrate and transmit received information in the form of chemical and electrical signals. Diverse morphologies of neurons differ in shape and function [1], [2]. They are basically composed of dendrites, cell bodies, synapses, and axons. A post-synaptic neuron receives neurotransmitters from the synapses of pre-synaptic neurons. In different cases, neurotransmitters can be regarded as excitatory or inhibitory, which causes the post-synaptic neuron to be more likely or less likely to generate a potential, respectively. Billions of neurons in the human brain construct a complex network to accomplish massive tasks, such as pattern recognition and logical deduction, among others. Therefore, simulating neuron computation and network architecture is key for intellectualization. In 1943, McCulloch and Pitts presented a simplified neuron model, which integrates multiple signal inputs with weights and mimics excitatory and inhibitory processes in neurons by means of a threshold [3]. Building upon the McCulloch-Pitts model, Rosenblatt proposed a perceptron model to solve binary linear classification problems [4]. However, the perceptron model suffers from nonlinear problems due to its one-layer architecture. In 1989, Hornik *et al.* constructed a standard feedforward network with the multiple layer perceptron (MLP) to approximate objective functions [5]. Since then, artificial neural networks have gradually become mainstream in the field of machine learning, and have been widely applied to various classification problems [6]–[8], disease prediction problems [9]–[11], time series forecasting [12]–[14], self-driving technology [15], real-time image fusion [16], drug combination prediction [17], [18], and so on. Considering the crucial role that pyramidal neurons play in the visual system in terms of perception and recognition, we primarily discuss the deep network model for image classification problems in this paper.

Convolution neural network (CNN) is a well-known feedforward network inspired by the neocognitron, it is mainly

This research was partially supported by the Japan Society for the Promotion of Science (JSPS) KAKENHI under Grant JP22H03643 and JP19K22891, Japan Science and Technology Agency (JST) Support for Pioneering Research Initiated by the Next Generation (SPRING) under Grant JPMJSP2145, and JST through the Establishment of University Fellowships towards the Creation of Science Technology Innovation under Grant JPMJFS2115. (Corresponding authors: Zhenyu Lei; Shangce Gao.)

Yu Zhang, Pengxing Cai, Zhiming Zhang, Zhenyu Lei, and Shangce Gao are with the Faculty of Engineering, University of Toyama, Toyama-shi, 930-8555, Japan. (E-mail:d2272005@ems.u-toyama.ac.jp; d2172005@ems.u-toyama.ac.jp; d2272007@ems.u-toyama.ac.jp; leizg@eng.u-toyama.ac.jp; gaosc@eng.u-toyama.ac.jp)

Yanan Sun is with Department of Artificial Intelligence, College of Computer Science, Sichuan University, Chengdu, China (e-mail: ysun@scu.edu.cn)

composed of multiple convolutional layers, pooling layers and fully connected layers. The architecture extracts higher-dimensional features more effectively than the traditional MLP model. Therefore, CNNs have rapidly developed and been successfully applied to image recognition in the last 30 years. LeCun *et al.* established a 7-layer CNN model named LeNet-5 for small-scale handwriting recognition [19]. With the data size and hardware technology gradually increase, large neural networks are proposed to process super-scale image recognition. AlexNet won the championship of ImageNet Large Scale Visual Recognition Challenge in 2012, it incorporates local response normalization and overlapping pooling into 11-layers CNN [20]. In order to further improve performance, Szegedy et al. presented inception module to replace the linear convolution layer of original CNN, and constructed a 22-layers network called GoogLeNet [21]. However, these deep neural networks with massive layers suffer from the problem of exploding or vanishing gradient. In 2015, ResNet employed residual mapping to modify the original features to solve the degradation problem, ensuring that the gradient can be updated in deep layers [22]. Although the performance of existing networks is already difficult to further improve, the uninterrupted pursuit of designing a CNN-like architecture with high accuracy has persisted over the last five years [23]–[25]. For example, the ConvNext model achieves competitive performance with large-scale models in various vision tasks, while maintaining the simplicity and efficiency of standard CNN [26]. Shen *et al.* presents a novel mathematical framework for designing high-performance CNN models without requiring GPUs or training data, named DeepMAD [27]. Recently, a novel method AMP-net is proposed, which effectively captures appearance and motion features using external memories and spatial-temporal fusion [28].

State-of-the art models attempt to design networks that can be applied to solve various pattern recognization problems. Nevertheless, most of these models suffer from poor biological interpretability or unsatisfactory performance. According to numerous physiological experimental results, dendritic computation is the key to information transmission between dendrites and synaptic inputs in neurons [29]–[31]. The mainstream network architectures are sequentially constructed from linear affine transformations and summation integration based on nonlinear activation functions. However, observed real-world neurons demonstrates that dendrites employ nonlinear operators before sum integration at the soma [32], [33]. Based on some of the above mentioned flaws and shortcomings of existing neural networks, in this work, we propose a novel lightweight network called MDPN with a simple biophysical theory to forcefully embody neuroplasticity based on the ingenious architecture. To validate the performance of MDPN, several advanced and standardized image datasets with different characteristics are selected for the classification task. The experimental results demonstrate excellent performance and a robust generalization capacity in comparison with other state-of-the-art network models on diverse classification datasets.

The contributions of this work can be described as follows:

1) In this paper, we mimic the biological structure of a multipolar neuron to model a five-layer lightweight network called MDPN, which performs remarkably well in handling diverse image classification problems. MDPN not only achieves better accuracy than other state-of-the-art algorithms, but also occupies the smallest memory space for multiple image datasets.

2) MDPN reconstructs the morphology of a neuron, which is innovatively divided into five layers containing presynaptic (input) layer, dendritic layer, soma layer, axon layer, and postsynaptic (output) layer. It is worth noting that MDPN maintains a shallow depth while preserving width through the repetition of dendritic input, making it suitable for constructing large-scale networks in subsequent applications.

3) Regarding on the synaptic nonlinearity and dendritic plasticity, we design a specific nonlinear operator and an autocephalous dendritic mechanism to replace the linear affine transformation and post-mature integration operation of the traditional feed-forward network for the first time.

4) The proposed network model exhibits high interpretability, simulating information transmission through neural connections via an innovative mechanism. It encapsulates the biophysical characteristics of the nervous system, thereby enhancing our comprehension of artificial neural networks.

The remainder of this paper is outlined as follows. Section II describes the related work, Section III gives the description of MDPN. Section IV summarizes the experimental result and provides illustration for 16 image classification problems. Section V presents the parameter analysis and performance discussion. Section VI concludes this work.

II. RELATED WORK

In this section, the related work on a novel network model based on biological theory are introduced. In fact, large-scale neural network models suffer from enormous resource costs and properties resuting in the difficulty of modification. In recent years, many researchers have focused on novel mechanisms with brief and clear architectures based on biological inspiration or basic theories from other fields. Inspired by the Hodgkin-Huxley model, a simple spiking model was proposed to simulate behaviour patterns of cortical neurons [34], which performs efficiently with few computations and produces rich firing patterns. Lin *et al.* presented an all-optical diffractive deep neural network architecture [35], it is physically formed by multiple layers of diffractive surfaces to perform various learning tasks. Gao *et al.* proposed a dendritic neuron model (DNM) by considering synaptic nonlinearity, inspired by the excitatory and inhibitory connections of neurons [36]. Kell *et al.* proposed a task-optimized neural network based on human auditory behavior [37]. Jones *et al.* designed a trainable neuron model with dendritic trees to solve some simple image classification problems [38]. According to the cable theory and dendritic plasticity, Beniaguev *et al.* designed a single cortical neuron model that modifies input patterns and dendritic computation inspired by biological simulation [39]. Giannari *et al.* employed the Hodgkin–Huxley model to propose a dynamic network based on three different types of neurons: inhibitory fast-spiking neurons, excitatory regular-spiking with adaptation neurons, and excitatory intrinsic bursting neurons [40]. Recently, a learnable and lightweight model called neural

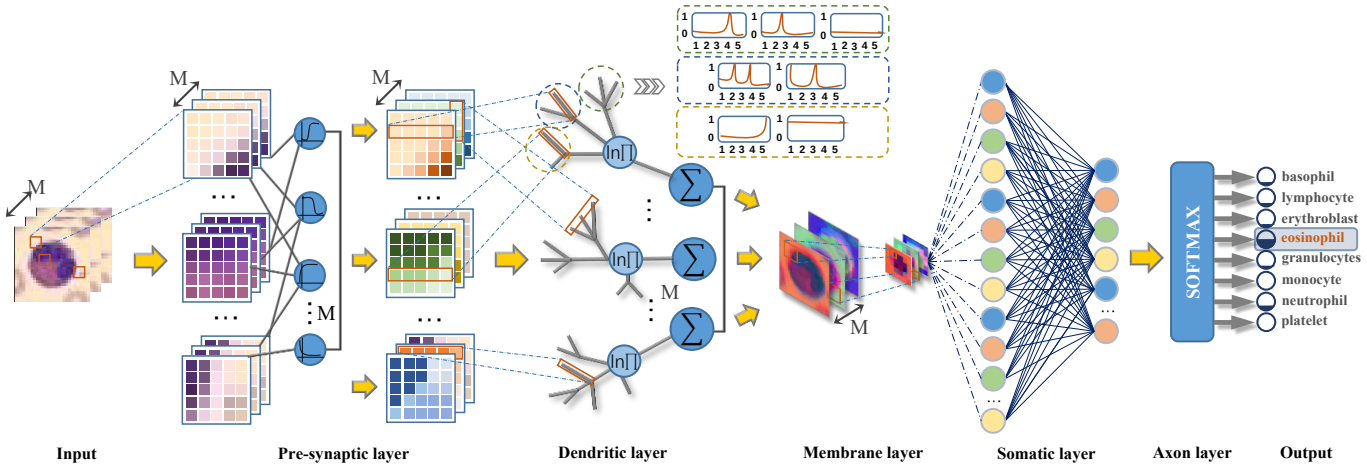

Fig. 1. The illustration of MDPN.

logic circuits was constructed based on neuroplasticity of the human brain [41].

III. MULTI-DENDRITE PYRAMIDAL NEURON MODEL

Pyramidal neurons have been observed in the cortical structures of all mammals and represent the most numerous excitatory cells. They possess multiple dendrites, each with plenty of branches, and play a crucial role in cognitive functions [42], [43]. The intrinsic firing characteristic of pyramidal neurons exhibit diversity. Influenced by variations in dendritic structure and channel distribution, pyramidal neurons can perform specific functions. Therefore, we propose a sequential network architecture named MDPN for image classification tasks, considering novel dendritic computation and neuroplasticity. According to the biological architecture of the pyramidal cell, we divide MDPN into five layers from the top to down [44]. It's worth mentioning that we simplify the classification of dendrites due to the dendritic plasticity of MDPN via parameters. Apical, oblique, and basal dendrite are not modeled separately.

In MDPN, a Pre-synaptic layer employs a sigmoidal function based on the sliding window mechanism for receiving input data. A Dendritic layer comprises a multiplicative operator and a logarithmic function to integrate the information from the Pre-synaptic layer. A Membrane layer uses a max pooling to downsample the high-dimensional data to reduce dimensionality of dataset. Dense connection operations are employed to map features of the recieved information to low dimensions at the somatic layer. An Axon layer finally outputs the classification results based on a softmax function. Its illustration and the original structure of MDPN are displayed in Fig. 1. In Fig. 1, we choose an image from the Bloodmnist dataset as an example, which extracts features through pre-synaptic layer and dendritic layer, and reduces the dimensionality in the membrane layer. Finally the fully-connected layer provides the final classification result.

A. Pre-synaptic layer

Most connections between neurons occur at synapses on dendrites. The post-synaptic neuron receives input information from the pre-synaptic one in the form of neurotransmitters. When neurotransmitter molecules from pre-synaptic neuron traverses the synaptic cleft and adhere to the membrane receptor of the post-synaptic neuron, an excitatory or inhibitory signal is confirmed to either facilitate or impede the propagation of neural impulses. Due to the unknown information processing received from pre-synaptic neurons within dendrites and the different states of excitability or inhibition, we implement dendritic plasticity through parameters. In precise terms, adjustments in weights and thresholds lead to modifications in the strength of synaptic connections. The establishment or dissolution of synaptic connection is directly influenced by the non-zero or all-zero connection, each specific parameter combination is elaborated below. In a pyramidal neuron, synaptic inputs from pre-synaptic neurons appear in the corresponding dendritic domains. Hence, we segregate synapses carrying distinct information into dendrites located in different regions. For a given input matrix $X^k \in R^{U\times V}$ (k = 1, 2, ..., M) from the kth pre-synaptic neuron to the dendritic input of the post-synaptic neuron, each $L \times L$ submatrix $X^k_{i,j}$ located in row i (i = 1, 2, ..., $U-L+1$), column j (j = 1, 2, ..., $V-L+1$) needs to used in dendritic computation. Obviously, this input method is equivalent to a sliding window mechanism with a stride of one. According to previous research [45], [46], we consider the existence of nonlinear transformation between synaptic connections. Therefore, the proposed nonlinear operator is formulated based on an inverse tangent function and two variables. The inverse tangent function has a smoother variation compared to the regular sigmoid function. The output range is then positioned between 0 and 1 by a normalization function. The corresponding output matrix $P^k_{i,j}$ can be formulated as:

$$P^k_{i,j} = \frac{\pi + 2tan^{-1}(\lambda(w^k \odot X^k_{i,j} - b^k))}{2\pi} \tag{1}$$

where λ is a positive integer to adjust the mapping space of output. w^k and b^k respectively denote the weight and threshold

matrices, which are generated based on the Gaussian random numbers. According to different combination of w and b, Eq. (1) presents four distinct situations as shown in Fig. 1: excitatory connection ($w>b>0$) shows a positive correlation with the input; on the contrary; inhibitory connection ($w<b<0$) is negatively correlated with any input; all-one connection ($w>0>b$ or $0>w>b$) where the output approaches 1 regardless of the input; all-zero connection ($w<b$) invariably export a number that is approximately 0. Each combination of w and b represents a specific morphology of a synapse located in a certain position. It's worth noting that each input data is normalized, meaning we are exclusively focused on the mapping space of the function within the range of [0,1].

B. Dendritic layer

Dendrite serves as the primary location for receiving and processing incoming information within a neuron. It perceives and integrates input signals from the pre-synaptic neurons. Excitatory input signals are defined by its inherent capacity to induce firing responses, indicating an increased likelihood of neuronal activation. On the other hand, inhibitory input signals exert inhibitory influences, dampening the occurrence of firing events.

A single neuron is capable of harboring multiple dendritic arborizations, allowing for the simultaneous reception of thousands to millions of input signals from a diverse array of neurons. The combined influence of excitatory and inhibitory inputs received by the dendritic branches collectively controls the neuron's propensity for firing, determining whether it will receive sufficient stimulation to fire.

Traditional linear summation suffers from poor information and the vanishing gradient problem when there is an equal number of all-zero and all-one connections within the dendritic layer. Meanwhile, it has been demonstrated that the nonlinear operator is involved in integrating sensory and motor information during behavior [47]. Therefore, a product operator with a logarithmic function is proposed to intergate the information of each dendritic. Then, the outputs of multiple dendritics are accumulated by a summation operators. Notly, as the base of the logarithmic function increases, the impact of the product of each dendrite on the overall integration becomes smaller. Therefore, we choose the compromising natural logarithm function instead of binary or common logarithm to achieve the most stable performance of the model. In this operator, multiplication provides nonlinearity and different feature information in comparison with summation, the natural logarithmic function amplifies (or reduces) the difference between two adjacent small (or large) values. The sum operation completes the final integration. The output matrix D^k in the dendritic layer can be formulated as:

$$D^k_{i,j} = \sum_{u=1}^{L} \ln \prod_{v=1}^{L} (P^k_{i,j}(u,v) + \gamma) \tag{2}$$

where u and v denote the index position of rows and columns in each submatrix. γ is a positive constant with range of $[0, 1]$ to avoid a minuscule value calculated by logarithmic function after product. Our proposed mechanism in dendritic layer can not only nonlinearly integrate the input information from the pre-synaptic layer, but also distinguish the approximate values under normalization and avoid the influence of outlier on the final results, so that the whole model will not be suffered from the problem of gradient vanishing or exploding.

C. Membrane layer

The membrane is the outer boundary of a neuron, separating it from the surrounding environment. It maintains the structural integrity of the neuron and controls the movement of substances. The neuronal membrane exhibits electrophysiological properties, allowing it to generate and transmit electrical signals. There is an ionic concentration difference between the interior and exterior of the membrane, which establishes a resting potential. When it receives neural stimulation, ionic channels on the membrane can open or close, leading to the generation of a nerve impulse based on the alteration of membrane potential. Based on the chemical characteristic mentioned above, we incorporate max pooling into the Membrane layer to filter out redundant information and retain a minor but significant amount of features. The formulation can be represented as:

$$E^k = \mathcal{M}_{\mathcal{R}} \left(D^k \right) \tag{3}$$

where p is the length of subdomains in input matrix D^k. Max pooling ($\mathcal{M}_{\mathcal{R}}$) reduces the spatial dimensions of input data by retaining the maximum value within each subdomain $\mathcal{R}$. This crucial process effectively decreases the number of parameters within the posterior soma layer, which serves to mitigate overfitting and enhance the generalization capability of the whole model.

D. Somatic layer

Soma plays a crucial role in information processing within neurons, as it receives signals from multiple dendrites. After the filtration by the membrane layer, the composite and prominent features are transmitted to the soma, where final integration takes place.

To achieve the integration of information and activation of nervous impulses, we incorporate a fully-connected operation and a rectified linear unit (ReLU) function into the somatic layer. The fully-connected operation allows the somatic layer to receive inputs from all the preceding layers in the network, enabling the integration of information from multiple sources. On the other hand, ReLU serves as the activation function, retaining nonlinearity by filtering out negative inputs and preserving positive inputs. The formulation can be expressed as follows:

$$S = ReLU \left(\begin{bmatrix} W^S_{11} & \cdots & W^S_{1n} \\ W^S_{21} & \cdots & W^S_{2n} \\ \vdots & \ddots & \vdots \\ W^S_{m1} & \cdots & W^S_{mn} \end{bmatrix} \begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_n \end{bmatrix} + \begin{bmatrix} B^S_1 \\ B^S_2 \\ \vdots \\ B^S_m \end{bmatrix} \right) \tag{4}$$

where n represents the length of features obtained by unfolding output matrices E of the membrane layer row by row and column by column. Parameter m is the number of

output features. W^S and B^S denote the weight and threshold, respectively. The primary purpose of utilizing fully-connected layers is to enable the whole model to effectively handle various classification tasks, extracting comprehensive feature information from the previous layer. ReLU is selected as the activation function is based on the considerations of avoiding the vanishing gradient problem and accelerating training speed. Additionally, ReLU is closer to the biological neural activation mechanism than the sigmoidal function.

E. Axon layer

In a neuron, the axon primarily serves as a pathway for transmitting information to the next neuron. Once a neural impulse is generated in the soma, it travels through the axon in the form of an electrical signal until reaching the terminal. Finally, neurotransmitters are released and transmitted to the synapses of post-synaptic neuron. Therefore, we employ a softmax activation function to map the output to values between 0 and 1:

$$O_z = \frac{e^{A_z}}{\sum_{k=1}^{l} e^{A_k}}, z = 1, 2, ..., l \tag{5}$$

where

$$A = \left(\begin{bmatrix} W_{11}^O & \cdots & W_{1m}^O \\ W_{21}^O & \cdots & W_{2m}^O \\ \vdots & \ddots & \vdots \\ W_{r1}^O & \cdots & W_{rm}^O \end{bmatrix} \begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_m \end{bmatrix} + \begin{bmatrix} B_1^O \\ B_2^O \\ \vdots \\ B_l^O \end{bmatrix} \right)^T \tag{6}$$

where W^O and B^O are the wights and thresholds in axon layer, and r denotes the number of types for classification problems.

F. Learning rule

Since the entire architecture of the presented MDPN model can be regarded as a feed-forward network, a gradient descent-based learning rule is employed to realize the optimization process. In this work, we adopt the Adam algorithm to update all the parameters to be optimized. Because the key of this mechanism is the nonlinear operation and integration function, we derive the updating formula for weight and threshold parameters in the pre-synaptic layer, which can be enumerated as:

$$\begin{aligned} w_{t+1} &= w_t - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} \\ b_{t+1} &= b_t - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} \end{aligned} \tag{7}$$

where ϵ is a constant to mantain numerical stability, α represents the learning rate, and the bias-corrected moments $\hat{v}_t$ and $\hat{m}_t$ are formulated as:

$$\begin{aligned} \hat{v}_t &= \frac{v_t}{1 - \beta_1^t} \\ \hat{m}_t &= \frac{m_t}{1 - \beta_2^t} \end{aligned} \tag{8}$$

where

$$\begin{aligned} v_t &= \beta_1 v_{t-1} + (1 - \beta_1) \nabla \mathcal{L} \odot \nabla \mathcal{L} \\ m_t &= \beta_2 m_{t-1} + (1 - \beta_2) \nabla \mathcal{L} \end{aligned} \tag{9}$$

where β_1 and β_2 are the decay rates, both v_t and m_t are initilized to 0 in the first iteration. $\nabla \mathcal{L}$ is the gradient operator for the loss function $\mathcal{L}$. Due to the utilization of the softmax function in multi-class classification problems, we employ cross-entropy function to evaluate the performance of the model, which is formulated as:

$$\mathcal{L} = -\frac{1}{N} \sum_{i=1}^{N} T_i ln O_i \tag{10}$$

where N is the number of samples. O and T respectively represent the predicted value and true value. Compared to other commonly used loss functions, cross-entropy loss can dynamically adjust the weight update rate based on the tendency of error variation. This avoids the gradient vanishing problem that arises when the activation function enters the saturation region. Based on the chain rule, the partial derivatives of $\mathcal{L}$ with respect to w and b can be calculated by:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial w_{r,s}^k} &= \frac{\partial \mathcal{L}}{\partial A_a} \frac{\partial A_a}{\partial S_p} \frac{\partial S_p}{\partial E_q} \frac{\partial E_q}{\partial D_{u,v}^k} \frac{\partial D_{u,v}^k}{\partial P_{u,v}^k(r,s)} \frac{\partial P_{u,v}^k(r,s)}{\partial w_{r,s}^k} \\ \frac{\partial \mathcal{L}}{\partial b_{r,s}^k} &= \frac{\partial \mathcal{L}}{\partial A_a} \frac{\partial A_a}{\partial S_p} \frac{\partial S_p}{\partial E_q} \frac{\partial E_q}{\partial D_{u,v}^k} \frac{\partial D_{u,v}^k}{\partial P_{u,v}^k(r,s)} \frac{\partial P_{u,v}^k(r,s)}{\partial b_{r,s}^k} \end{aligned} \tag{11}$$

According to Eq. (5) and Eq. (9), we can get that:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial A_a} &= \frac{\partial \mathcal{L}}{\partial O_a} \frac{\partial \mathcal{L}}{\partial O_a} \\ &= -\frac{T_a}{N O_a} \frac{e^{A_a} (\sum_{i=1}^{r} e^{A_i} - e^{A_a})}{(\sum_{i=1}^{r} e^{A_i})^2} \\ &= -\frac{T_a (\sum_{i=1}^{r} e^{A_i} - e^{A_a})}{N \sum_{i=1}^{r} e^{A_i}} \end{aligned} \tag{12}$$

The fully connection in the somatic layer is represented as an affine transformation of the matrix, the $\frac{\partial A_a}{\partial S_p}$ can be formulated as:

$$\frac{\partial A_a}{\partial S_p} = W_{a,p}^O \tag{13}$$

Comparing Eq. (4) with Eq. (6), the difference between these two formulas is whether or not the rectified linear unit is employed, the piecewise function can be found that:

$$\frac{\partial S_p}{\partial E_q} = \begin{cases} 0, if \ (\sum_{i=1}^{n} W_{p,i}^S + B_p^S) < 0 \\ W_{p,q}^S, \quad otherwise \end{cases} \tag{14}$$

Continuously, according to the pooling operation of Eq. (3), we are able to get that:

$$\frac{\partial E_q}{\partial D_{u,v}^k} = \begin{cases} 1, if \ D_{u,v}^k = \max(\mathcal{R}) \\ 0, otherwise \end{cases} \tag{15}$$

In our work, we employ a novel integration operator instead of a sum operation. Different from the gradient of the traditional integration unit, which is equal to 1, according to the definition

TABLE I
EXPERIMENTAL RESULTS OF MODEL FOR CLASSIFICATION DATASETS.

Model	Bloodmnist	Breastmnist	Chestmnist	Dermamnist	OCTmnist	OrganAmnist	OrganCmnist	OrganSmnist
	Accuracy	Accuracy	Accuracy	Accuracy	Accuracy	Accuracy	Accuracy	Accuracy
Alexnet	91.991	86.538	63.959	73.167	64.000	87.935	87.119	73.825
Bi-LSTM	82.987	75.000	63.946	71.072	48.000	61.081	67.066	52.022
BLS	85.267	80.769	62.074	71.820	60.100	71.470	72.642	55.680
DD	88.249	83.974	63.768	71.970	62.300	74.137	74.141	55.193
DNM	87.606	79.487	49.681	68.130	43.200	76.173	75.714	60.460
DSAN	85.618	81.410	59.002	70.474	70.900	88.356	79.161	72.318
SDENet	94.475	82.692	51.152	73.217	69.300	84.486	84.857	70.053
MDPN	**94.651**	**87.179**	**64.022**	**75.561**	**71.100**	**88.615**	**88.377**	**74.289**
Pathmnist	Pneumoniamnist	Retinamnist	Tissuemnist	EmnistE	EmnistL	Fer2013	Fashionmnist	Size
Accuracy	Accuracy	Accuracy	Accuracy	Accuracy	Accuracy	Accuracy	Accuracy	
81.922	87.981	52.500	57.341	96.610	93.804	25.367	91.157	92/97M
77.799	82.692	54.000	49.454	16.917	12.094	34.950	82.514	8.5/8.5M
65.139	87.179	54.250	53.665	98.903	90.681	39.233	88.614	3.7/4.4M
63.162	90.064	55.000	50.334	98.510	87.775	39.750	88.443	1.3/2.8M
56.128	84.936	45.750	47.551	10.063	72.167	32.817	84.314	1.8/4.3M
75.070	82.372	45.000	59.648	**99.613**	94.036	**53.033**	90.757	193/202M
81.017	83.974	43.750	54.091	99.510	**94.042**	46.833	90.771	7.6/7.6M
82.423	**92.147**	**55.750**	**59.833**	99.390	92.638	49.800	**91.371**	**0.9/2.7M**

of Eq. (2), we can find that:

$$\begin{aligned}\frac{\partial D^k_{u,v}}{\partial P^k_{u,v}(r,s)} &= \frac{\prod_{u=1,u\neq r}^{L}\prod_{v=1,v\neq s}^{L}(P^k_{i,j}(u,v)+\gamma)}{\prod_{u=1}^{L}\prod_{v=1}^{L}(P^k_{i,j}(u,v)+\gamma)} \\ &= \frac{\prod_{u=1}^{L}\prod_{v=1}^{L}(P^k_{i,j}(u,v)+\gamma)}{\prod_{u=1}^{L}\prod_{v=1}^{L}(P^k_{i,j}(u,v)+\gamma)(P^k_{i,j}(r,s)+\gamma)} \\ &= \frac{1}{P^k_{i,j}(r,s)+\gamma}\end{aligned} \tag{16}$$

Finally, we derive the gradient with respect to the weight parameters, which can be given by:

$$\frac{\partial P^k_{u,v}(r,s)}{\partial w^k_{r,s}} = \frac{\lambda X^k_{i,j}(r,s)}{\pi(1+\lambda^2(w^k_{r,s}X^k_{i,j}(r,s)-b^k_{r,s})^2)} \tag{17}$$

Similarly, we can get that:

$$\frac{\partial P^k_{u,v}(r,s)}{\partial b^k_{r,s}} = -\frac{\lambda}{\pi(1+\lambda^2(w^k_{r,s}X^k_{i,j}(r,s)-b^k_{r,s})^2)} \tag{18}$$

G. Biological plausibility of the MDPN

At present, the development of large-scale neural networks has led to the acceptance of a common view that neural networks mainly rely on the connections between numerous neurons, and a single neuron has a small effect. However, previous research shows that a single neuron can compelete some simple tasks [41], [48], and neurons in the brain receive a large number of signals, ultimately only one axon within a neuron will generate the final output signal [29]. Therefore, constructing a neuron model with neuroplasticity and biological interpretability is still essential for brain science and real artificial intelligence [49], [50].

Compared to traditional neural networks based on linear dendritic computation and sum integration operators, the design of MPDN is closer to biological principles. Firstly, the repetitive inputs of a data are proven to have a positive influence on the whole training process of the model [51]. Therefore, we consider the biological signals recieved by the current neuron from different pre-synaptic neurons as repetitions of input data, which are assigned to different dendrites, each dendrite ensures plasticity through parameter variations. Secondly, the observed physiology of neurons reveals a characteristic that dendrites produce nonlinearity between synapse and pre-synaptic neuron before the information integration [52], [53]. In MDPN, the Pre-synaptic layer employs an inchoate nonlinear function to simulate the information transmission process between synapses. Finally, the repetitive inputs generate excitation or inhibition through different dendrites, which are partially integrated into the soma. Most traditional integration operations consist of individual summations. Nevertheless, we present a multiplication-like integration function to process the information from the Dendritic layer. This is because that multiplication-like formulas can enhance the performance of the network model [54], [55], and it has been demonstrated in experiments with the drosophila melanogaster ON motion vision circuit [56].

IV. EXPERIMENTAL SETUP AND RESULTS

A. Experimental setup

In the experiment, 16 standardized image datasets involving peripheral blood cells (Bloodmnist) [57], breast cancer (Breastmnist), chest disease (Chestmnist), pigmented skin lesions (Dermamnist), optical coherence tomography (Octmnist), liver tumor in axial view (OrganAmnist), liver tumor in coronal view (OrganCmnist), liver tumor in sagittal view (OrganSmnist), colorectal cancer (Pathmnist), pediatric pneumonia (Pneumoniamnist), diabetic retinopathy (Retinamnist), human kidney cortex cells (Tissuemnist), extended handwritten digits (EmnistE) [58], handwritten letters (EmnistL), fashion products (Fashionmnist), and facial expressions (Fer2013) [59] datasets are used to test the performance of our proposed neuron model. The table in Supplementary Material lists the detailed parameters of the chosen datasets. It is clear that each dataset represents a specific classification task with different types of imgaes and labels, which can verify the universality and robustness of MDPN.

B. Performance comparison

To prove the effectiveness of our proposed classifier, we select six classical or state-of-the-art network models as competitors, including single dendritic neuron model (DNM) which considers the dendritic plasticity [36], attention-based bidirectional long short-term memory network (Bi-LSTM) [60], broad learning system for multilayer perceptron-based method (BLS) [61], deep subdomain adaptation network (DSAN) [62], the white-box MLP-based dendrite net (DD) [63], Alexnet [63], and novel single-source domain expansion network (SDENet) [64]. It is worth noting that our proposed model consists of only five layers. However, the data transmitted repetitively from the presynaptic layer to the dendritic layer in between is much higher than the depth. In theoretical terms, a single MDPN model is closer to the broad learning module. Therefore, we choose a classical broad learning model BLS as a comparative model. The other competitors are also lightweight network models based on either biological theories or novel architectures. All the experiments are developed by python software, and they are all implemented on Ubuntu 20.04.6 LTS with dual 2.10 GHz Intel(R) Xeon(R) silver 4110 CPU and conducted on a single NVIDIA RTX A6000 GPU. The classification results of MDPN and other competitors are listed in Table I. In this experiment, each model is trained 5 times on each dataset separately with 50 epochs. The batch size and initial learning rate are set as 64 and 1E-3, respectively.

In Table I, the accuracy value represents the optimal overall accuracy of each model from 5 independent experiments for all datasets. The bolded digit denotes the best classification accuracy in comparison with other competitors. The model size indicates the average occupied memory on grayscale or RGB image datasets, which is a crucial indicator for lightweight networks to achieve efficient computation in resource-constrained environments. From the experimental results in Table I, MDPN achieves the highest accuracy on 13 datasets using far fewer model parameters than competing products. Take the Dermamnist and Pneumoniamnist datasets as examples, MDPN improves 2.344%-7.431% and 2.083%-9.775% overall accuracy in comparison with other algorithms, respectively. It is clear that MDPN outperforms other state-of-the-ssrt network models on most datasets, while maintaining the smallest model size for any dataset. To reflect the classification results of each class, we illustrate the confusion matrices of each model on Bloodmnist in Fig. 2, where each column represents a predicted category, and the sum of each column indicates the number of images predicted as that category. Each row represents the true label, and the total count of each row is the number of instances with that category. In the confusion matrix of each model, we present representative images for each class in Bloodmnnist, it can be observed that the first and fourth classes exhibit some similarity in shape. As a result, apart from our proposed MDPN and SDENet, all competitors show very poor classification accuracy. Additionally, the second and fifth classes also share some similarities. Because of scarce samples of the fifth class, the accuracy of the second class generally tends to surpass that of the fifth class. However, our proposed MDPN not only ensures high-level accuracy for the second class but also optimally classifies the other. The performance of MDPN for other classes is also outstanding, with the classification accuracy for the eighth class reaching 99.9%. From Fig. 2, we can draw the conclusion that MDPN outperforms state-of-the-art models in terms of effectiveness and robustness.

V. Discussion

The above experimental results verify the superiority of MDPN compared to classical and state-of-the-art models on general MNIST-like datasets in terms of classification accuracy. Our proposed brand-new mechanism and integration function hold promise for the establishment of large-scale networks with credible interpretability. In order to testify the feasibility of MDPN, we take the mechanism of traditional MLP-based networks into consideration for additional comparison. The analysis of parameters is intended to achieve the optimal performance of MDPN.

A. Analysis of transformation in pre-synaptic layer

The traditional MLP-based neural network mainly consists of numerous neurons, each of which executes a simple affine transformation operation in the first step. However, the linear affine transformation may potentially leave out some information, and the generated large number is difficult to employ in multiplication integration before the nonlinear activation function. Therefore, we employ an equivalent number of parameters to nonlinearly map the original input to a standardized range. In this section, we test all the datasets mentioned above between MDPN with affine transformation and nonlinear function under the same conditions. In this experiment, the affine transformation can be formulated under the same situation as:

$$P_{i,j}^{k} = w^{k} \odot X + b^{k} \tag{19}$$

The experimental results can be listed in Table II, where the Wilcoxon signed-ranks test is employed to calculate significant differences between the MDPN with two different transformation modes. The p-value is a crucial indicator used to determine the significance of statistical results, indicating the probability that the control model significantly performs better than the competitor with a probability of 1-p. Therefore, a lower p-value signifies a higher statistical significance of the difference, and a p-value less than or equal to a predefined threshold value is generally considered to be statistically significant. R+ represents the summation of ranks for multiple classification results where the control model performs better than the competitive model, and R− is the summation of ranks for the opposite.

In Table II, it is clearly observed that MDPN with our proposed nonlinear processing performs better than MDPN_L with the traditional linear function on each dataset except for chestmnist in terms of both mean and standard deviation of overall accuracy. When the predefined significance level is set to 0.05, it is obvious that our proposed model based on nonlinear dendritic computation outperforms the linear

This article has been accepted for publication in IEEE Transactions on Artificial Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAI.2024.3379968

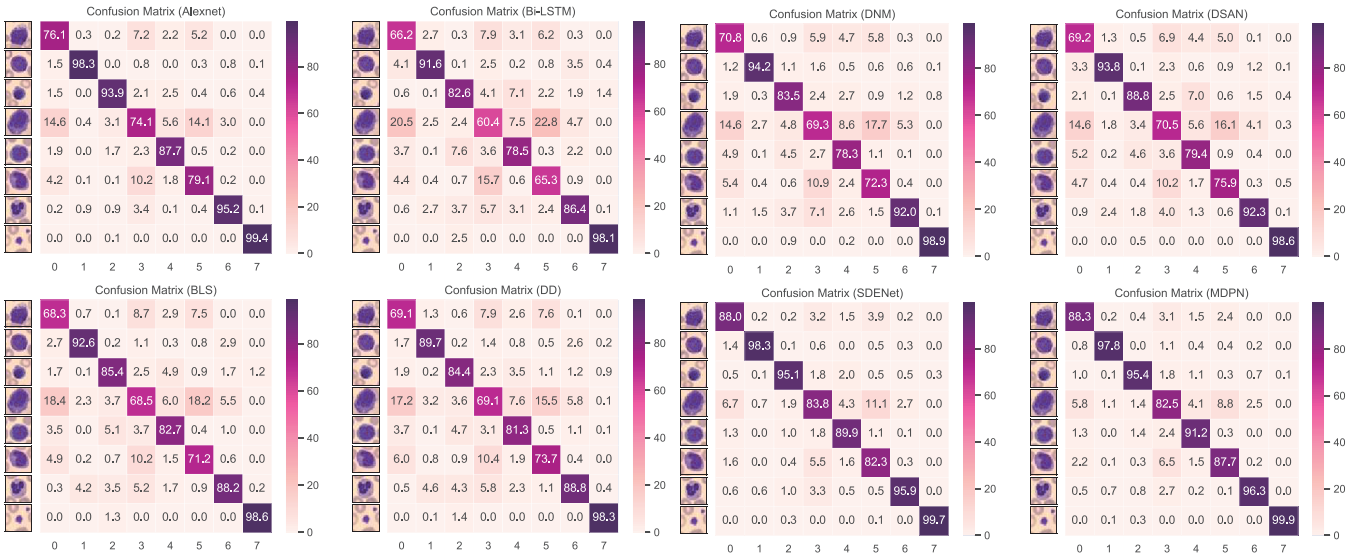

Fig. 2. Confusion matrix on Bloodmnist.

TABLE II
STATISTICAL COMPARISON BETWEEN MDPN WITH LINEAR AND OUR PROPOSED PROCESSING.

Dataset	MDPN	MDPN_L	R+	R-	p-value
	Mean±Std	Mean±Std			
Bloodmnist	94.24±0.12	62.54±28.74	15	0	0.0312
Breastmnist	86.41±1.05	78.33±2.98	15	0	0.0312
Chestmnist	63.98±0.03	64.00±0.03	3	12	0.9062
Dermamnist	75.10±0.34	71.23±1.46	15	0	0.0312
OCTmnist	70.78±0.21	51.50±15.87	15	0	0.0312
OrganAmnist	88.28±0.22	83.71±4.26	15	0	0.0312
OrganCmnist	87.84±0.31	81.75±5.31	15	0	0.0312
OrganSmnist	73.68±0.36	62.10±12.65	15	0	0.0312
Pathmnist	81.33±1.01	41.76±33.68	15	0	0.0312
Pneumoniamnist	91.18±0.72	78.46±14.58	15	0	0.0312
Retinamnist	55.10±1.09	51.95±5.02	10	5	0.3125
Tissuemnist	59.71±0.10	52.81±11.59	15	0	0.0312
EmnistE	99.35±0.02	99.35±0.04	10	5	0.3125
EmnistL	92.45±0.16	78.08±30.95	15	0	0.0312
Fer2013	48.80±0.59	27.45±4.46	15	0	0.0312
Fashionmnist	91.30±0.05	90.78±0.24	15	0	0.0312

transformation significantly on 13 datasets according to the Wilcoxon signed-ranks test. This testifies to the effectiveness and robustness of our proposed dendritic computation.

B. Analysis of integration operator in dendritic layer

Choosing the right integration method is crucial to accurately summarize the input data, which can make influence on final decisions and results. Integration operation is usually used to summarize the data based on summation after mapping function in traditional feedforward neural network, which can be formulated based on Eq. 2 as:

$$D_{i,j}^{k} = \sum_{u=1}^{L}\sum_{v=1}^{L} P_{i,j}^{k}(u,v) \tag{20}$$

However, MDPN model early employs a nonlinear transformation to adjust the input values between 0 and 1. For specific input types, we select the product operation to mimic a probability of compound events regarding neuron firing. Due to the explosive reduction of the value caused by the multiplication of multiple decimals, we introduce a parameter and logarithmic function to distinguish the integrated information. Finally, a summation formula is applied to integrate the information.

TABLE III
STATISTICAL COMPARISON BETWEEN MDPN WITH SUMMATION AND OUR PROPOSED OPERATION.

Dataset	MDPN	MDPN_S	R+	R-	p-value
	Mean±Std	Meanpm1Std			
Bloodmnist	94.24±0.12	31.79±27.55	15	0	0.0312
Breastmnist	86.41±1.05	73.97±2.00	15	0	0.0312
chestmnist	63.98±0.03	63.96±0.01	7	8	0.5938
dermamnist	75.10±0.34	69.25±1.51	15	0	0.0312
OCTmnist	70.78±0.21	58.82±8.04	15	0	0.0312
OrganAmnist	88.28±0.22	62.24±30.06	15	0	0.0312
OrganCmnist	87.84±0.31	67.02±16.07	15	0	0.0312
OrganSmnist	73.68±0.36	64.91±4.35	15	0	0.0312
Pathmnist	81.33±1.01	17.17±0.00	15	0	0.0312
Pneumoniamnist	91.18±0.72	78.26±14.39	15	0	0.0312
Retinamnist	55.10±1.09	51.15±4.89	13.5	1.5	0.0938
Tissuemnist	59.71±0.10	49.53±2.02	15	0	0.0312
EmnistE	99.35±0.02	10.15±0.00	10	5	0.3125
EmnistL	92.45±0.16	40.72±20.76	15	0	0.0312
Fer2013	48.80±0.59	25.65±1.70	15	0	0.0312
Fashionmnist	91.30±0.05	10.35±0.00	15	0	0.0312

Table III exhibits the result of comparison between traditional summation and our proposed integration mechanism. We can observe that MDPN with multiplication and logarithmic function outperforms the summation on each classification dataset, it is worth noting that a low accuracy with a standard deviation of 0 indicates that the model does not converge. The possible reasons for the model not converging may include insufficient accumulated information or the generation of poor gradients. According to statistical result of wilcoxon signed-ranks test, we can find that original MDPN performs better/tied/worse than MDPN_S on 13/2/1 datasets. Experimental results demonstrate that our proposed integration operator is more promising than traditional summation in synthesizing the

probability-like values, which further enhance the performance of classifier.

C. Analysis of parameters

The unique mechanism allows our proposed MDPN to achieve acceptable performance on small-scale image classification tasks with a minimal model size. In addition to some predefined parameters, integration weight γ, the number of dendrite M, scale of synapse input L also affect the performance of MDPN. Since the dendritic input from presynaptic layer is within the range of $[0, 1]$, using a single multiplication operation may result in excessively small integrated data, making it difficult to distinguish between images belonging to different categories. Therefore, the decision of weight parameter γ is crucial for accurately classifying the images. Fig. 3 shows the line chart of mean accuaracy when

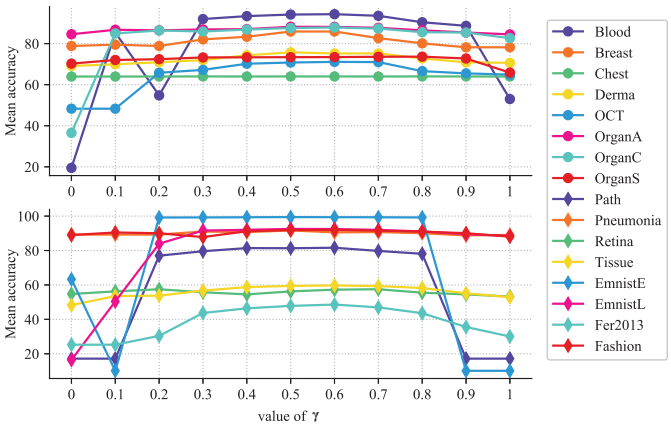

Fig. 3. The overall accuracy of MDPN with different γ.

$\gamma = \{0, 0.1, 0.2, ..., 1\}$, where the horizontal coordinate denotes the value of γ, and the vertical coordinate represents the mean accuracy. It is clearly observed that whether the value of γ is too closer to 0 or 1, MDPN can not effectively classify parts of the dataset during training. When the value γ is equal to 0.6, MDPN performs excellent on every dataset, due to the fact that its integrated value lies in a suitable interval.

The scale of dendrite is also a significant factor impacting the size and performance of model, including the number of dendrites M and the number of synapses L in each dendrite. In Table IV, we list the mean value and standard deviation of MDPN with M={10, 20, 30, ..., 80} dendrites on all the grayscale image datasets, and each RGB image dataset is triple the corresponding value of M.

It is obeserved that as M increases, the overall size of the model grows linearly, but it doesn't positively impact the classification performance. Each MDPN with a different value of M might perform superior than other variants on a specific dataset, depending on the complexity and scale of the problem. Overall, as images carry more features, the model requires a greater number of parameters, resulting in a larger M. Conversely, the opposite holds true. At the same time, the insensitivity to hyperparameters demonstrates the robustness and generalizability of MDPN. To obtain a comprehensive parameter selection, we gather accuracy data of all datasets into a single sequence of samples. In comparison with Wilcoxon signed-rank test, Friedman test is more suitable for the situation of multiple samples. Therefore, the Friedman test is employed with a significance level set to 0.05 for generating an unadjusted p-value. In addition to the unadjusted p-value, the other adjusted p-values are obtained through three post hoc analysis procedures, containing Bonferroni correction, Sidak correction, Bonferroni–holm method, respectively. All the p-values serve different purposes, and each p-value below 0.05 indicates that the control model is superior to the competitor. In Table V, we can conclude that the model size of MDPN becomes proportionally larger as M increases. MDPN achieves the optimal performance when M is equal to 50.

The number of synapses L in each dendrite determines the sharpness and detail of the input image during the process. To illustrate the different characteristics of MDPN when L = {1, 3, 5, 7, 9}, we extract the image after the dendritic layer of the trained model.

Fig. 5 exhibits the obtained image after our proposed nonlinear function of microscopic peripheral blood cells, facial expressions, fashion products, and so on. The images in the black box are the original images, and the subsequent five images are the extracted feature maps under different L. It is observed that as the synaptic size gradually increases, the images extracted from the dendritic layer become more blurred, making it easier to accomplish classification tasks. However, blurry images may also lead to the omission of some important features, resulting in classification errors. Therefore, two box-and-whisker plots are illustrated to verify the comprehensive performance by multiple indicators on Bloodmnist, OCTmnist, OrganAmnist, OrganCmnist, OrganSmnist, Tissuemnist, Fer2013, and Fashionmnist datasets. In Fig. 4, the black circle represents an outlier, the top and bottom lines represent the highest and lowest accuracy, respectively. The upper and lower sides of the polygon are quantiles, and the red line represents the median. It is observed that MDPN performs at the same level with different scale $L * L$ of dendrite, the selection of L may result in slight performance differences when encountering different datasets. In general, MDPN exhibits the most stability on all datasets when L is set to 5.

VI. Conclusion

In this study, we present a novel lightweight neuron model called MDPN, designed specifically to address standardized image classification tasks. Unlike conventional neuron models and their variants, which rely on the premature activation function and a summation integration operator, this architecture deviates from the established physiological understanding of the nervous system. To bridge this gap, we formulate a 5-layer network architecture inspired by the pyramidal neuron structure and grounded in established physiological principles, thereby augmenting the performance of the foundational model.

To substantiate the effectiveness and resilience of the proposed neuron model, we conducted experiments on 12 standardized biomedical image datasets, along with three extended

TABLE IV
CLASSIFICATION ACCURACY OF MDPN WITH DIFFERENT M.

M	Bloodmnist		Breastmnist		Chestmnist		Dermamnist		OCTmnist		OrganA
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean
10	92.8208	0.3952	85.0000	1.0726	63.9549	0.0232	**75.2718**	0.3772	69.0200	1.3682	85.8702
20	93.6451	0.1832	85.3846	1.6593	63.9522	**0.0116**	75.1920	0.4399	70.2200	1.4550	87.2539
30	93.7796	0.1759	84.8718	2.1066	63.9736	0.0346	75.1022	0.6041	69.6800	1.2834	87.5959
40	93.8497	0.1724	84.8718	**0.9722**	63.9602	0.0139	74.8928	0.3005	70.0800	0.9985	87.8097
50	94.2414	**0.1291**	**86.4103**	1.0533	**63.9861**	0.0320	75.1022	0.3408	70.7800	**0.2168**	88.2833
60	94.1654	0.3152	85.1282	1.3897	63.9700	0.0462	74.6633	0.2639	**70.8400**	0.6804	87.7298
70	**94.3408**	0.2468	85.6410	1.3293	63.9638	0.0480	74.6933	**0.1713**	69.7200	1.3123	88.1179
80	94.2414	0.3645	83.8462	6.2216	63.9522	0.0201	74.8229	0.2350	70.4200	0.9576	**88.3069**
-mnist	OrganCmnist		OrganSmnist		Pathmnist		Pneumoniamnist		Retinamnist		Tissue
Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean
0.5777	86.2071	0.4416	70.9435	0.6143	79.9582	1.0489	89.5192	0.7568	55.5000	1.3578	58.6413
0.3942	86.8916	0.2793	72.3253	0.1685	79.8357	0.8684	89.4872	0.7818	**56.0500**	0.9585	59.2868
0.2016	87.3295	0.3286	72.9278	**0.1265**	80.7632	0.4499	90.7051	**0.2998**	54.8000	1.1096	59.4141
0.3393	87.3972	**0.1767**	72.6152	0.2550	79.9387	1.0644	90.7051	0.9548	55.5500	0.6471	59.6189
0.2282	**87.8423**	0.3137	**73.6799**	0.3617	**81.3315**	1.0155	**91.1859**	0.7256	55.1000	**0.4541**	59.7115
0.3866	87.6705	0.3923	72.7376	0.5571	79.8663	**0.2737**	90.4808	0.7818	54.4500	0.5420	59.5385
0.4461	87.7794	0.4339	73.1884	0.2853	80.5098	1.1301	90.3846	0.4389	54.4000	0.9287	**59.7276**
0.3670	87.7673	0.2144	73.4534	0.2566	80.2507	0.6908	90.5128	0.4727	54.7000	0.8732	59.5482
-mnist	EmnistE		EmnistL		Fer2013		Fashionmnist		Size		Rank
Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std			
0.1943	99.3060	0.0228	92.4203	0.0820	48.1600	0.4846	90.2000	0.1351	194K/560K		8
0.1445	99.3513	0.0212	**92.4898**	0.1041	48.7000	0.3333	90.5571	0.1961	378K/1.08M		7
0.1788	99.3733	0.0347	92.4261	0.0816	48.3500	**0.2841**	90.8143	0.2013	562K/1.62M		5
0.0637	99.3900	**0.0115**	92.4319	0.1000	48.4867	0.3741	91.0686	0.0650	746K/2.16M		4
0.1073	99.3580	0.0214	92.4536	0.1670	**48.8000**	0.5913	91.3086	**0.0540**	930K/2.70M		**1**
0.2735	99.3833	0.0361	92.5043	**0.0754**	46.9200	0.3499	91.1800	0.0568	1.08M/4.61M		6
0.2298	**99.3940**	0.0196	92.4855	0.2156	47.1300	0.5861	91.3571	0.1097	1.26M/5.15M		2
0.1919	99.3833	0.0122	92.2942	0.1595	47.0300	0.4604	**91.4086**	0.1376	1.44M/5.69M		3

TABLE V
THE FRIEDMAN TEST OF MDPN WITH DIFFERENT PARAMETER M.

M	Unadjusted p	p_{bonf}	p_{sidak}	p_{holm}	Size
10	4.75E-16	1.33E-14	1.24E-14	1.33E-14	0.19/0.55M
20	2.73E-08	7.65E-07	7.64E-07	7.10E-07	0.37/1.09M
30	3.80E-05	0.001066	0.001065	7.99E-04	0.55/1.63M
40	5.86E-04	0.016427	0.016298	0.010561	0.73/2.16M
50	-	-	-	-	0.91/2.70M
60	9.10E-05	0.002548	0.002545	0.001821	1.09/4.62M
70	0.039088	1	0.672562	0.488661	1.27/5.16M
80	0.006214	0.174005	0.160164	0.099431	1.45/5.70M

handwritten character datasets, as well as datasets pertaining to fashion products and facial expressions. Our results demonstrate that MDPN consistently outperforms state-of-the-art network models across a spectrum of classification datasets. Furthermore, it surpasses the majority of existing networks with regard to space-time complexity. The innovative architecture and computational paradigm of MDPN exhibit closer alignment with physiological principles compared to conventional neural networks, thus advancing our understanding of information processing mechanisms within the human brain. Unfortunately, due to its simple architecture, MDPN performs only averagely on high-resolution image classification tasks. However, our proposed dendritic computation operator and integration mechanism holds promise for achieving satisfactory classification results with a relatively compact model size in future network constructions based on MDPN.

REFERENCES

[1] R. Parekh and G. A. Ascoli, "Neuronal morphology goes digital: a research hub for cellular and system neuroscience," *Neuron*, vol. 77, no. 6, pp. 1017–1038, 2013.

[2] D. Udvary, P. Harth, J. H. Macke, H.-C. Hege, C. P. de Kock, B. Sakmann, and M. Oberlaender, "The impact of neuron morphology on cortical network architecture," *Cell Reports*, vol. 39, no. 2, 2022.

[3] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The Bulletin of Mathematical Biophysics*, vol. 5, pp. 115–133, 1943.

[4] F. Rosenblatt, "The perceptron: a probabilistic model for information storage and organization in the brain." *Psychological Review*, vol. 65, no. 6, p. 386, 1958.

[5] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[6] X. Zheng, H. Sun, X. Lu, and W. Xie, "Rotation-invariant attention network for hyperspectral image classification," *IEEE Transactions on Image Processing*, vol. 31, pp. 4251–4265, 2022.

[7] O. El Ariss and K. Hu, "Resnet-based parkinson's disease classification," *IEEE Transactions on Artificial Intelligence*, vol. 4, no. 5, pp. 1258–1268, 2023.

[8] B. Arslan, S. Memiş, E. B. Sönmez, and O. Z. Batur, "Fine-grained food classification methods on the uec food-100 database," *IEEE Transactions on Artificial Intelligence*, vol. 3, no. 2, pp. 238–243, 2022.

[9] J. Yang, Z. Zeng, K. Wang, H. Zou, and L. Xie, "Garbagenet: A unified learning framework for robust garbage classification," *IEEE Transactions on Artificial Intelligence*, vol. 2, no. 4, pp. 372–380, 2021.

[10] L. Peng, N. Wang, N. Dvornek, X. Zhu, and X. Li, "Fedni: Federated graph learning with network inpainting for population-based disease prediction," *IEEE Transactions on Medical Imaging*, vol. 42, no. 7, pp. 2032–2043, 2023.

[11] M. B. Abubaker and B. Babayiğit, "Detection of cardiovascular diseases in ecg images using machine learning and deep learning methods," *IEEE Transactions on Artificial Intelligence*, vol. 4, no. 2, pp. 373–382, 2023.

[12] D. Cheng, F. Yang, S. Xiang, and J. Liu, "Financial time series forecasting with multi-modality graph neural network," *Pattern Recognition*, vol. 121, p. 108218, 2022.

[13] Z. Jia, X. Cai, G. Zheng, J. Wang, and Y. Lin, "Sleepprintnet: A multivariate multimodal neural network based on physiological time-series for automatic sleep staging," *IEEE Transactions on Artificial Intelligence*, vol. 1, no. 3, pp. 248–257, 2020.

[14] Y. Hu, X. Cheng, S. Wang, J. Chen, T. Zhao, and E. Dai, "Times series forecasting for urban building energy consumption based on graph convolutional network," *Applied Energy*, vol. 307, p. 118231, 2022.

This article has been accepted for publication in IEEE Transactions on Artificial Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAI.2024.3379968

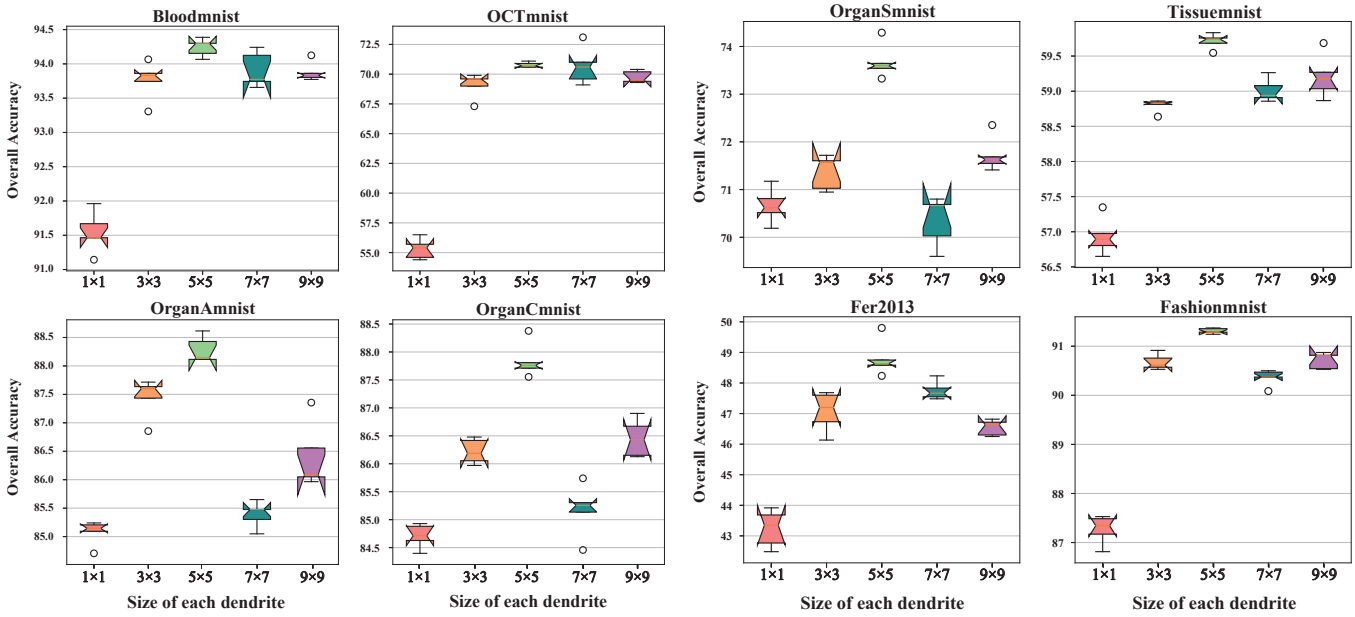

Fig. 4. The Box-and-whisker plots of MDPN.

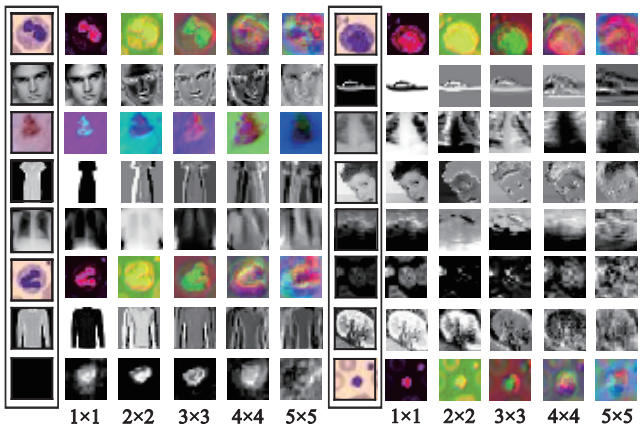

Fig. 5. The obtained image after nonlinear transformation of MDPN with different L.

[15] J. Ni, K. Shen, Y. Chen, W. Cao, and S. X. Yang, "An improved deep network-based scene classification method for self-driving cars," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–14, 2022.

[16] L. Tang, J. Yuan, and J. Ma, "Image fusion in the loop of high-level vision tasks: A semantic-aware real-time infrared and visible image fusion network," *Information Fusion*, vol. 82, pp. 28–42, 2022.

[17] W. Jin, J. M. Stokes, R. T. Eastman, Z. Itkin, A. V. Zakharov, J. J. Collins, T. S. Jaakkola, and R. Barzilay, "Deep learning identifies synergistic drug combinations for treating covid-19," *Proceedings of the National Academy of Sciences*, vol. 118, no. 39, p. e2105070118, 2021.

[18] F. Cheng, I. A. Kovács, and A.-L. Barabási, "Network-based prediction of drug combinations," *Nature Communications*, vol. 10, no. 1, p. 1197, 2019.

[19] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[21] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[23] C. Han, T. Ma, J. Huyan, X. Huang, and Y. Zhang, "Crackw-net: A novel pavement crack image segmentation convolutional neural network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 22 135–22 144, 2021.

[24] M. Böhle, M. Fritz, and B. Schiele, "Optimising for interpretability: Convolutional dynamic alignment networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[25] A. M. G. Allah, A. M. Sarhan, and N. M. Elshennawy, "Edge u-net: Brain tumor segmentation using mri based on deep u-net model with boundary information," *Expert Systems with Applications*, vol. 213, p. 118833, 2023.

[26] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A convnet for the 2020s," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11 976–11 986.

[27] X. Shen, Y. Wang, M. Lin, Y. Huang, H. Tang, X. Sun, and Y. Wang, "Deepmad: Mathematical architecture design for deep convolutional neural network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 6163–6173.

[28] Y. Liu, J. Liu, K. Yang, B. Ju, S. Liu, Y. Wang, D. Yang, P. Sun, and L. Song, "Amp-net: Appearance-motion prototype network assisted automatic video anomaly detection system," *IEEE Transactions on Industrial Informatics*, vol. 20, no. 2, pp. 2843–2855, 2024.

[29] M. London and M. Häusser, "Dendritic computation," *Annu. Rev. Neurosci.*, vol. 28, pp. 503–532, 2005.

[30] A. L. Vlasits, R. D. Morrie, A. Tran-Van-Minh, A. Bleckert, C. F. Gainer, D. A. DiGregorio, and M. B. Feller, "A role for synaptic input distribution in a dendritic computation of motion direction in the retina," *Neuron*, vol. 89, no. 6, pp. 1317–1330, 2016.

[31] P. Poirazi and A. Papoutsi, "Illuminating dendritic function with computational models," *Nature Reviews Neuroscience*, vol. 21, no. 6, pp. 303–321, 2020.

[32] H. Agmon-Snir, C. E. Carr, and J. Rinzel, "The role of dendrites in auditory coincidence detection," *Nature*, vol. 393, no. 6682, pp. 268–272, 1998.

[33] X. Li, J. Tang, Q. Zhang, B. Gao, J. J. Yang, S. Song, W. Wu, W. Zhang, P. Yao, N. Deng *et al.*, "Power-efficient neural network with artificial dendrites," *Nature Nanotechnology*, vol. 15, no. 9, pp. 776–782, 2020.

[34] E. Izhikevich, "Simple model of spiking neurons," *IEEE Transactions on Neural Networks*, vol. 14, no. 6, pp. 1569–1572, 2003.

[35] X. Lin, Y. Rivenson, N. T. Yardimci, M. Veli, Y. Luo, M. Jarrahi, and

A. Ozcan, "All-optical machine learning using diffractive deep neural networks," *Science*, vol. 361, no. 6406, pp. 1004–1008, 2018.

[36] S. Gao, M. Zhou, Y. Wang, J. Cheng, H. Yachi, and J. Wang, "Dendritic neuron model with effective learning algorithms for classification, approximation, and prediction," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 2, pp. 601–614, 2018.

[37] A. J. Kell, D. L. Yamins, E. N. Shook, S. V. Norman-Haignere, and J. H. McDermott, "A task-optimized neural network replicates human auditory behavior, predicts brain responses, and reveals a cortical processing hierarchy," *Neuron*, vol. 98, no. 3, pp. 630–644, 2018.

[38] I. S. Jones and K. P. Kording, "Might a single neuron solve interesting machine learning problems through successive computations on its dendritic tree?" *Neural Computation*, vol. 33, no. 6, pp. 1554–1571, 2021.

[39] D. Beniaguev, I. Segev, and M. London, "Single cortical neurons as deep artificial neural networks," *Neuron*, vol. 109, no. 17, pp. 2727–2739, 2021.

[40] A. Giannari and A. Astolfi, "Model design for networks of heterogeneous hodgkin–huxley neurons," *Neurocomputing*, vol. 496, pp. 147–157, 2022.

[41] H. T. Ünal and F. Başçiftçi, "Neural logic circuits: An evolutionary neural architecture that can learn and generalize," *Knowledge-Based Systems*, vol. 265, p. 110379, 2023.

[42] G. N. Elston, R. Benavides-Piccione, and J. DeFelipe, "The pyramidal cell in cognition: a comparative study in human and monkey," *The Journal of Neuroscience*, vol. 21, no. 17, p. RC163, 2001.

[43] A. Gidon, T. A. Zolnik, P. Fidzinski, F. Bolduan, A. Papoutsi, P. Poirazi, M. Holtkamp, I. Vida, and M. E. Larkum, "Dendritic action potentials and computation in human layer 2/3 cortical neurons," *Science*, vol. 367, no. 6473, pp. 83–87, 2020.

[44] N. Spruston, "Pyramidal neurons: dendritic structure and synaptic integration," *Nature Reviews Neuroscience*, vol. 9, no. 3, pp. 206–221, 2008.

[45] S. Li, N. Liu, X. Zhang, D. W. McLaughlin, D. Zhou, and D. Cai, "Dendritic computations captured by an effective point neuron model," *Proceedings of the National Academy of Sciences*, vol. 116, no. 30, pp. 15 244–15 252, 2019.

[46] T. Haga and T. Fukai, "Dendritic processing of spontaneous neuronal sequences for single-trial learning," *Scientific Reports*, vol. 8, no. 1, p. 15166, 2018.

[47] N.-l. Xu, M. T. Harnett, S. R. Williams, D. Huber, D. H. O'Connor, K. Svoboda, and J. C. Magee, "Nonlinear dendritic integration of sensory and motor input during an active sensing task," *Nature*, vol. 492, no. 7428, pp. 247–251, 2012.

[48] M. Hausser, N. Spruston, and G. J. Stuart, "Diversity and dynamics of dendritic signaling," *Science*, vol. 290, no. 5492, pp. 739–744, 2000.

[49] A. Destexhe and E. Marder, "Plasticity in single neuron and circuit computations," *Nature*, vol. 431, no. 7010, pp. 789–795, 2004.

[50] Y. Humeau and D. Choquet, "The next generation of approaches to investigate the link between synaptic plasticity and learning," *Nature Neuroscience*, vol. 22, no. 10, pp. 1536–1543, 2019.

[51] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[52] P. Poirazi, T. Brannon, and B. W. Mel, "Arithmetic of subthreshold synaptic summation in a model ca1 pyramidal cell," *Neuron*, vol. 37, no. 6, pp. 977–987, 2003.

[53] S. Chavlis and P. Poirazi, "Drawing inspiration from biological dendrites to empower artificial neural networks," *Current Opinion in Neurobiology*, vol. 70, pp. 1–10, 2021.

[54] C. Koch and T. Poggio, "Multiplying with synapses and neurons," in *Single neuron computation*. Elsevier, 1992, pp. 315–345.

[55] C. Koch and I. Segev, "The role of single neurons in information processing," *Nature Neuroscience*, vol. 3, no. 11, pp. 1171–1177, 2000.

[56] L. N. Groschner, J. G. Malis, B. Zuidinga, and A. Borst, "A biophysical account of multiplication by a single neuron," *Nature*, vol. 603, no. 7899, pp. 119–123, 2022.

[57] J. Yang, R. Shi, D. Wei, Z. Liu, L. Zhao, B. Ke, H. Pfister, and B. Ni, "Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification," *Scientific Data*, vol. 10, no. 1, p. 41, 2023.

[58] G. Cohen, S. Afshar, J. Tapson, and A. Van Schaik, "Emnist: Extending mnist to handwritten letters," in *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2017, pp. 2921–2926.

[59] L. Zahara, P. Musa, E. P. Wibowo, I. Karim, and S. B. Musa, "The facial emotion recognition (fer-2013) dataset for prediction system of micro-expressions face using the convolutional neural network (cnn) algorithm based raspberry pi," in *2020 Fifth International Conference on Informatics and Computing (ICIC)*. IEEE, 2020, pp. 1–9.

[60] S. Mei, X. Li, X. Liu, H. Cai, and Q. Du, "Hyperspectral image classification using attention-based bidirectional long short-term memory network," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–12, 2021.

[61] C. P. Chen and Z. Liu, "Broad learning system: An effective and efficient incremental learning system without the need for deep architecture," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 1, pp. 10–24, 2017.

[62] Y. Zhu, F. Zhuang, J. Wang, G. Ke, J. Chen, J. Bian, H. Xiong, and Q. He, "Deep subdomain adaptation network for image classification," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 4, pp. 1713–1722, 2020.

[63] G. Liu and J. Wang, "Dendrite net: A white-box module for classification, regression, and system identification," *IEEE Transactions on Cybernetics*, vol. 52, no. 12, pp. 13 774–13 787, 2021.

[64] Y. Zhang, W. Li, W. Sun, R. Tao, and Q. Du, "Single-source domain expansion network for cross-scene hyperspectral image classification," *IEEE Transactions on Image Processing*, vol. 32, pp. 1498–1512, 2023.

Yu Zhang received the M.S. degree in artificial intelligence from the University of Toyama, Toyama, Japan, in 2022. He is currently pursuing the Ph.D. degrees degree at University of Toyama, Toyama, Japan. His main research interests include machine learning, pattern recognition, and neuroscience.

Pengxing Cai received the M.S. degree in artificial intelligence from the University of Toyama, Toyama, Japan, in 2021. She is currently pursuing the Ph.D. degrees degree at University of Toyama, Toyama, Japan. Her current research interests include evolutionary algorithm, engineering optimization, and neural network.

Yanan Sun (Member, IEEE) received his Ph.D.degree in Computer Science from Sichuan University, Chengdu, China in 2017. He is currently a Professor with the College of Computer Science, Sichuan University, Chengdu, China. His current research interests include evolutionary computation, neural networks, and their applications on neural architecture search. He designed the indicator of "GPU Day", which has been widely used among the community of neural architecture search. He was ranked as "World's Top 2% Scientists 2021" collectively released by Stanford University and Springer. He serves as an Associate Editor for IEEE Transactions on Evolutionary Computation.

Zhiming Zhang received the M.S. degree in artificial intelligence from the University of Toyama, Toyama, Japan, in 2022. He is currently pursuing the Ph.D. degrees degree at University of Toyama, Toyama, Japan. His current research interests include computer vision, machine learning, neural networks and bioinformatics.

This article has been accepted for publication in IEEE Transactions on Artificial Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAI.2024.3379968

Zhenyu Lei received the Ph.D. degree in Science and Engineering from the University of Toyama, Toyama, Japan, in 2023. He is currently an Assistant Professor with the Faculty of Engineering, University of Toyama, Japan. His current research interests include evolutionary computation, machine learning, and neural network for real-world applications and optimization problems.

Shangce Gao (Senior Member, IEEE) received his Ph.D. degree in Innovative Life Science from University of Toyama, Toyama, Japan in 2011. He is currently a Professor with the Faculty of Engineering, University of Toyama, Japan. His current research interests include nature-inspired technologies, machine learning, and neural networks for real-world applications. He serves as an Associate Editor for many international journals such as IEEE Transactions on Neural Networks and Learning Systems, and IEEE/CAA Journal of Automatica Sinica.