Rethinking the Role of Activation Functions in Deep Convolutional Neural Networks for Image Classification

Qinghe Zheng, Mingqiang Yang, Xinyu Tian, Xiaochen Wang and Deqiang Wang

Abstract—**Deep convolutional neural network used for image classification is an important part of deep learning and has great significance in the field of computer vision. Moreover, it helps humans to simulate the human brain more realistically, pointing out the direction for the development of artificial intelligence. In fact, the rapid development and its application of deep neural networks are due to the improvements of various activation functions. The activation function is one of the most critical parts of the neural networks, which provides the possibility of strong nonlinear fitting ability of the deep neural network. In this paper, we analyze how the activation function affects the deep neural network, and then analyzes and summarizes the development status and the performance of different activation functions. Based on these, we designed a new activation function to improve the classification performance of neural networks. Finally, we perform extensive classification experiments on the MNIST, CIFAR10/100, and ImageNet datasets, and compare various popular activation functions to provide a reference for the selection of activation functions when designing deep neural network models. Deep convolutional neural networks, including the four models AlexNet, VGGNet, GoogLeNet, and Network in Network (NIN), are used to observe the role of the activation function in training and testing phase. The experimental results show that the constructed deep convolutional neural networks based on the improved activation function not only have a faster convergence rate, but also can improve the image classification accuracy more effectively.**

Index Terms—**deep learning, image classification, activation function, generalization, overfitting**

I. Introduction

In recent years, deep learning model is the most remarkable direction in the field of machine learning. It interprets data, such as images, voice and text, by imitating the working mechanism of human brain. It is widely used in automatic driving [1] [2] [3], medical diagnosis [4] [5] [6], speech recognition [7] [8] [9], machine translation [10] [11] and some other fields. Its concept originates from the research and development of artificial neural networks. Back propagation algorithm [12] makes deep learning model no longer remote, and eventually brings the revival of deep learning research based on statistical model [13].

Activation functions are the core of deep neural network structure. It is just a node added to the output of the neural network, also known as a transfer function. It can also connect two layers of neural network models. It is used to determine whether the neural network output is yes or no, mapping the output values between 0 and 1 or between -1 and 1 (depending on the activation functions between different two layers). At present, the popular and commonly used activation functions include Sigmoid function [14], Tanh function [15], ReLU function [16], Leaky ReLU function [17], etc. However, the gradient vanishing problem usually occurs in the backward transferring of sigmoid function, which greatly reduces the training speed and convergence results.

The ReLU function can effectively alleviate the gradient vanishing problem. It can be used to train the deep neural network in the supervised manner without relying on the unsupervised layer-by-layer pre-training, which significantly improves the performance of the deep convolutional neural network. Krizhevsky and others [18] tested the commonly used activation functions ReLU, sigmoid and tanh functions, and proved that the performance of the ReLU function is better than the sigmoid function.

However, ReLU also has fatal shortcomings. Firstly, the output of ReLU function is greater than 0, so that the output is not zero mean, that is, mean shift (bias shift), which easily leads the neurons of the latter layers to get the signals of non-zero mean output of the upper layer as input, making the calculation of network parameter difficult. Secondly, as the training progresses, part of the input will fall into the hard saturation region of ReLU function, resulting in the result of corresponding weight cannot be updated. Mean shift and neuron death affect the convergence and optimizing speed of deep neural networks.

The main function of activation functions in deep neural networks is to provide the ability of non-linear modeling of networks. If only linear convolution and full connection operations are included in the network, it can only express linear mapping. Even if the depth of the network is increased, it is still linear mapping, which makes it difficult to effectively tune the data of non-linear distribution in the real environment. When the activation function is applied to deep convolutional neural networks, it mainly has an impact on network training in the forward and backward process.

Manuscript received March 13, 2019; revised November 19, 2019. This work was supported by the Fundamental Research Funds of Shandong University (Grant 2018JC040), National Natural Science Foundation of China (Grant 61571275), and National Key Research and Development Program of China (Grant 2018YFC0831503).

Qinghe Zheng, Mingqiang Yang (corresponding author) and Deqiang Wang are with School of Information Science and Engineering, Shandong University, Jimo, Qingdao 266237, Shandong, China (e-mail: yangmq@ -sdu.edu.cn).

Xinyu Tian is with College of Mechanical and Electrical Engineering, Shandong Management University, Changqing, Jinan 250357, Shandong, China.

Xiaochen Wang is with Shanghai MicroPort Medical (Group) Co., Ltd., Shanghai, 201203, China.

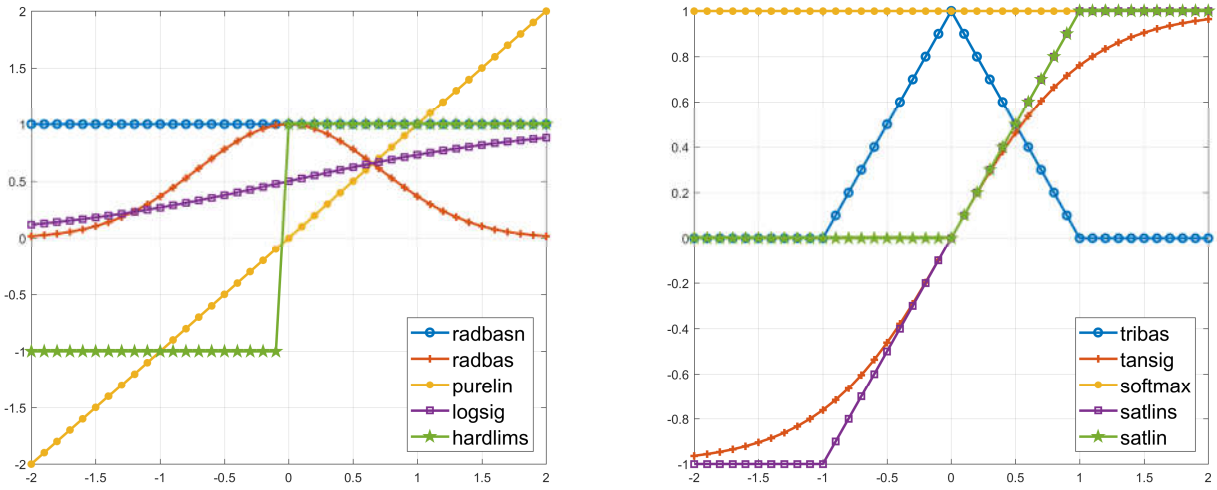

Fig. 1. Curves of some classic activation functions in the MATLAB platform.

In deep convolutional neural networks, activation function of hidden layer neurons is at the core position . Obviously, it is equivalent to the based kernel function [46] in signal decomposition. According to the theory of signal processing, the true optimal transformation is based on the specific signal characteristics; the basis function that matches it best is selected, and the signal decomposition is performed based on the preferred basis function. It is difficult to say that, for a certain kind of signal, the wavelet transform or the Fourier transform is good. Since the feedforward network design has the same model as the signal decomposition, optimizing the type of activation functions according to the actual problem is important for designing a high performance deep neural network. At present, there is almost no theoretical guidance for the selection of neural network activation functions. The activation functions of most specific applications are determined through a large number of experiments [47]. Therefore, it is very important to summarize the advantages and disadvantages of the popular activation functions for different situations and to lay the foundation for better application of the deep learning model in the future [42].

In this paper, in order to solve gradient vanishing problem of neural network model, we analyze the advantages, defects and intrinsic properties of several popular activation functions, and construct a piecewise function as new activation function based on ReLu and Swish functions. Finally, we use four deep convolutional neural networks (*i.e.* AlexNet, VGGNet, Goog- LeNet, and NIN) to conduct intensive experiments on four public datasets (*i.e.*, MNIST, CIFAR10, CIFAR100, and ImageNet) to compare the effects of various neuron activation functions on the convergence speed and image classification performance of deep CNN. Finally, experimental results show that the activation function type of the neuron is equivalent to the basis function in the signal decomposition. If it is not fully optimized, the generalization ability and performance of deep convolutional neural network cannot be obtained, especially the classification performance on the unseen test data. In addition, the proposed new activation function plays a very significant role in improving the training speed and reducing the error rate of the deep neural network.

The rest of the paper is organized as the follows. We first introduce and analyze the recent developments of activation functions in deep learning in Section II. Then we present multiple popular activation functions in deep CNN for image classification in Section III. Finally, experimental results and corresponding analysis are introduced in Section IV. Finally, we discuss what we learned, our conclusions, and the future works in Section V.

II. Related Works

In this part, we introduce the meaning of deep learning model, and its key parts, activation function, development. Then, we summarize the existing problems, difficulties, challenges and improvements made by researchers.

In 1980, Fukushima *et al.* [19] proposed the concept of convolutional neural network (CNN). In 1998, Lecun *et al.* [20] further realized and optimized the convolution neural network. In 2003, Simard *et al.* [21] simplified the neural network model. The above three typical deep learning models introduced above are from different teams. DBN comes from the Hinton team at the University of Toronto, SAE comes from the Bengio team at the University of Montreal, and CNN from the Lecun team at New York University. It can be seen that image classification systems based on deep convolution neural network are more and more widely used, and the research work of deep convolution neural network has been highly valued by researchers. But some of these headaches still do not have a better solution. For example, the deep learning model itself is complex and difficult to implement; the training algorithm of the deep learning model determines that the model is easy to diffuse gradients, and the model is not easy to converge, which requires a lot of time to debug; there is no complete general theory yet, so the design of neural network structures and training models requires a lot of practical skills and needs to constantly explore the best parameters and optimization algorithms and so on. In order to solve the gradient vanishing problem in deep neural network model, the advantages and disadvantages of many activation functions ReLU [22] and Softplus [23] are analyzed.

In fact, the preliminary study of the activation functions went through three phases:

- Simple linear function: this model does not reflect the nonlinear characteristics of the activation function and

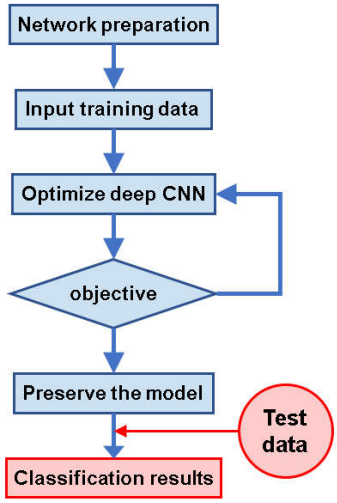

Fig. 2. The experimental flow and steps of deep CNN models for image classification.

does not have the characteristics of classification, so it is rarely used.

- Linear threshold function: This kind of function usually has good classification performance, but because it is a non-conductible function, it is difficult to find an effective learning algorithm.
- Nonlinear function: This type of function gives the deep neural network model a powerful nonlinear fitting ability to capture the features and patterns behind massive data.

Now, choosing the most effective one from many complex activation functions is one of the most critical steps in the practical application. Some of the classic activation functions in the MATLAB platform are shown in Fig. 1.

III. Popular Activation Functions in Deep CNN Model for Image Classification

In this section, we first introduce some popular activation functions in deep convolution neural networks (CNNs) for image classification and respective properties. Furthermore, we propose a new activation function used for improving the classification performance of neural networks. The flow of the entire algorithm is shown in Fig. 2.

A. Effect of Activation Function in Deep Convolutional Neural Networks

Activation function in deep neural network refers to the preservation and mapping of the "characteristics of activated neurons" through non-linear functions, which is the key to solving non-linear optimization problems in neural networks. When the activation function is linear, the linear combination of massive linear equations can only be expressed linearly. Even if the network has multiple layers, it is equivalent to a linear network with single hidden layer. This kind of linear representation of the input is only equivalent to a multilayer perceptron. This makes it impossible to approximate arbitrary functions with nonlinearities. Due to the performance of the network model is far from meeting the practical requirements, researchers have tried to use a combination of nonlinearities [24]. The use of activation function increases the nonlinearity of the neural network model, making the deep neural network meaningful. Furthermore, the traditional activation functions will reduce the input value to a fixed interval, because the gradient-based optimization method will be more stable when the output value of the activation function is limited. The new sparse activation functions based on brain nerves have high training efficiency, but in this case, a smaller learning rate is generally required. Three kinds of constructions of loss planes caused by different activation functions are presented in Fig. 3.

We draw the role of activation function in the whole neural network model, including forward and backward propagation stages, as shown in Fig. 4. It can be seen that the output of the activation function in the deep convolutional neural network is defined as

$$a = f(\boldsymbol{w}x + \boldsymbol{b}) \tag{1}$$

where $f(\cdot)$ represents the activation function, $\boldsymbol{w}$ represents the weights of deep neural networks and $\boldsymbol{b}$ denotes the bias. It can be seen from equation (1) that the input vector first makes an inner product with the weight vector, the inner product and the offset term, and finally outputs through an activation function. The role of the activation functions here is to input the data. The feature is preserved and mapped by its own nonlinear characteristics and passed to the next neuron in the next layer. In the forward propagation stage, the entire data transmission process from the input layers to the hidden output layers is through the information processing characteristic in nonlinear activation functions. On the other hand, the back-propagation process is aimed to update the weights and offset values of the network. Now, we introduce how activation functions make a role in the training and testing process of deep convolutional neural networks.

Take the mean square error function as an example, after a single layer of forward-propagation, the output error of deep neural network can be expressed as

$$E_{input\to output} = \frac{1}{2}(label - activation)^2 \tag{2}$$

In the back-propagation process, the weight $\boldsymbol{w}$ needs to be updated. Take the $\boldsymbol{w}_0$ as an example, according to the chain rule, we can get its update rule as

$$\frac{\partial E_{input\to output}}{\partial \boldsymbol{\omega}_0} = \frac{\partial E_{input\to output}}{\partial activation}\frac{\partial activation}{\partial output}\frac{\partial output}{\partial \boldsymbol{\omega}_0} \tag{3}$$

where

$$\frac{\partial E_{input\to output}}{\partial activation} = \frac{\partial}{\partial activation}\left[\frac{1}{2}(label - activation)^2\right] = activation - label \tag{4}$$

$$\frac{\partial activation}{\partial output} = f'(output) \tag{5}$$

$$\frac{\partial output}{\partial \boldsymbol{\omega}_0} = \frac{\partial}{\partial \boldsymbol{\omega}_0}\left[a_0\boldsymbol{\omega}_0 + a_1\boldsymbol{\omega}_1 + b\right] \tag{6}$$

Then $\boldsymbol{w}_0$ is updated according to

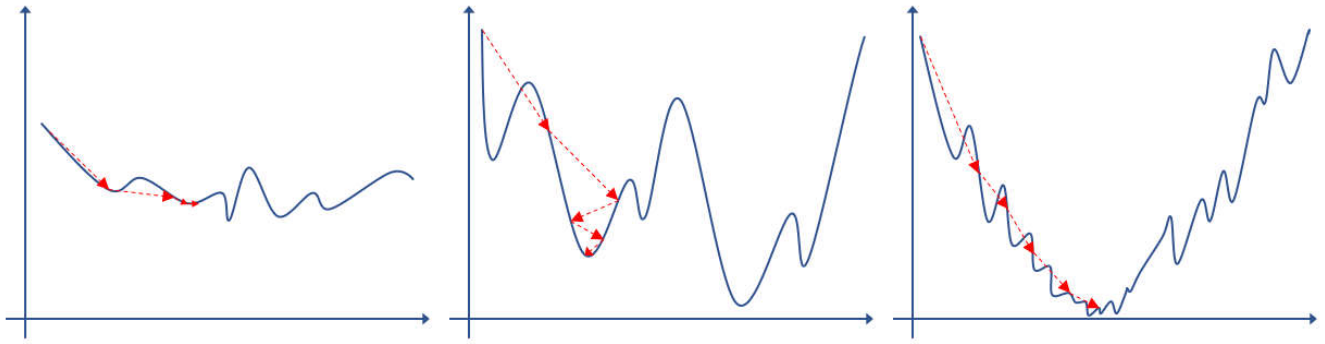

Fig. 3. Three kinds of constructions of loss planes caused by different activation functions.

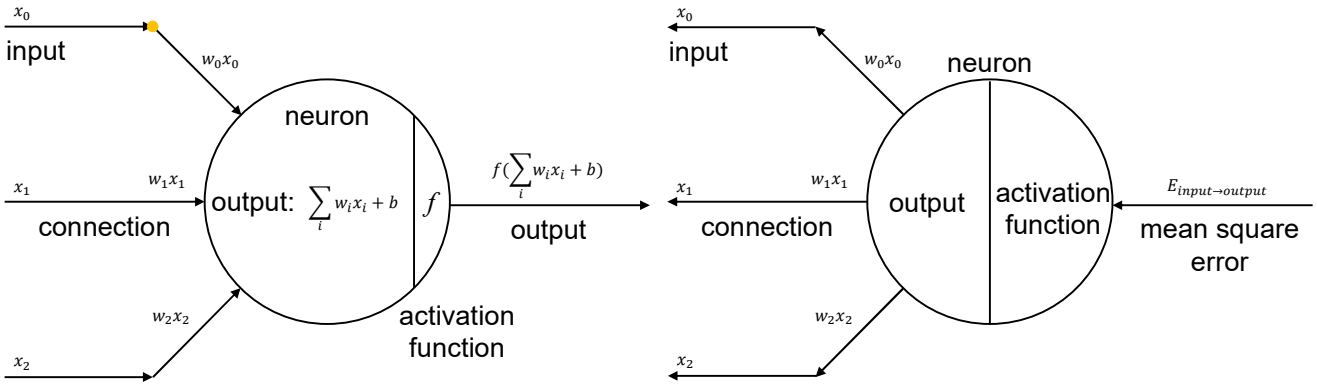

Fig. 4. The forward and backward propagation process of deep convolutional neural network.

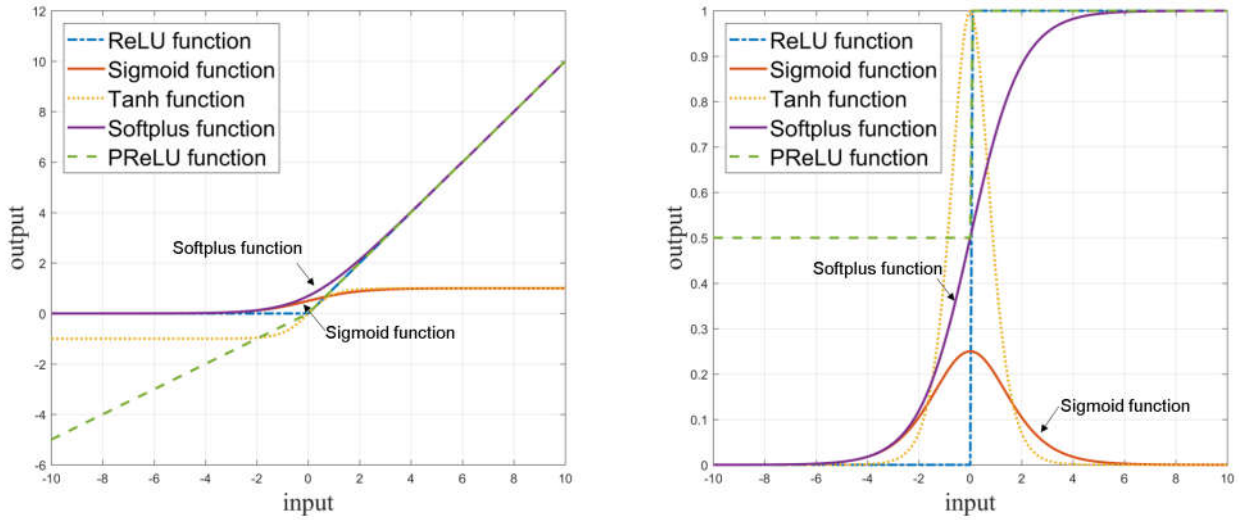

Fig. 5. Popular activation functions in deep neural networks and their corresponding derivatives.

$$\boldsymbol{\omega}_0' = \boldsymbol{\omega}_0 - \alpha \frac{\partial E_{input \to output}}{\partial \boldsymbol{\omega}_0} \tag{7}$$

The updating way for other weights in the network is similar to the above process. Equation (7) shows that in each updating process, the residual is multiplied by learning rate and partial derivative at each level. If inappropriate activation function is selected, the deep convolutional neural network will unable to be effectively trained because of the small bottom residual after the multi-layer propagation. Therefore, in the process of back-propagation, the choice of activation function will affect the convergence results of the whole model.

B. Popular Activation Functions

Activation functions, as an indispensable component of deep learning, playing a vital role in it. The revival of neural networks benefits from the design of a specific activation function (*i.e.*, ReLU). It solves the "vanishing gradient" problem in the deep neural network, and makes the training of deep network model come true. The deeper the network is, the more complex and abstract the semantic features will be captured, which is very effective for object classification. At present, a large number of excellent activation functions have been proposed, including Leaky-ReLU [17], PReLU [25], ELU [26], and Swish [27]. However, there are no conclusions as to what activation function can perform best in what scenario. There are also no universal standards to determine which property is effective [45].

Each neuron node in the neural network accepts the output value of the upper layer of neurons as the input value of the neuron, and passes the input value to the next layer. The neuron node in the input layer passes the input value directly to the underlying layers (hidden layers or output layers). In

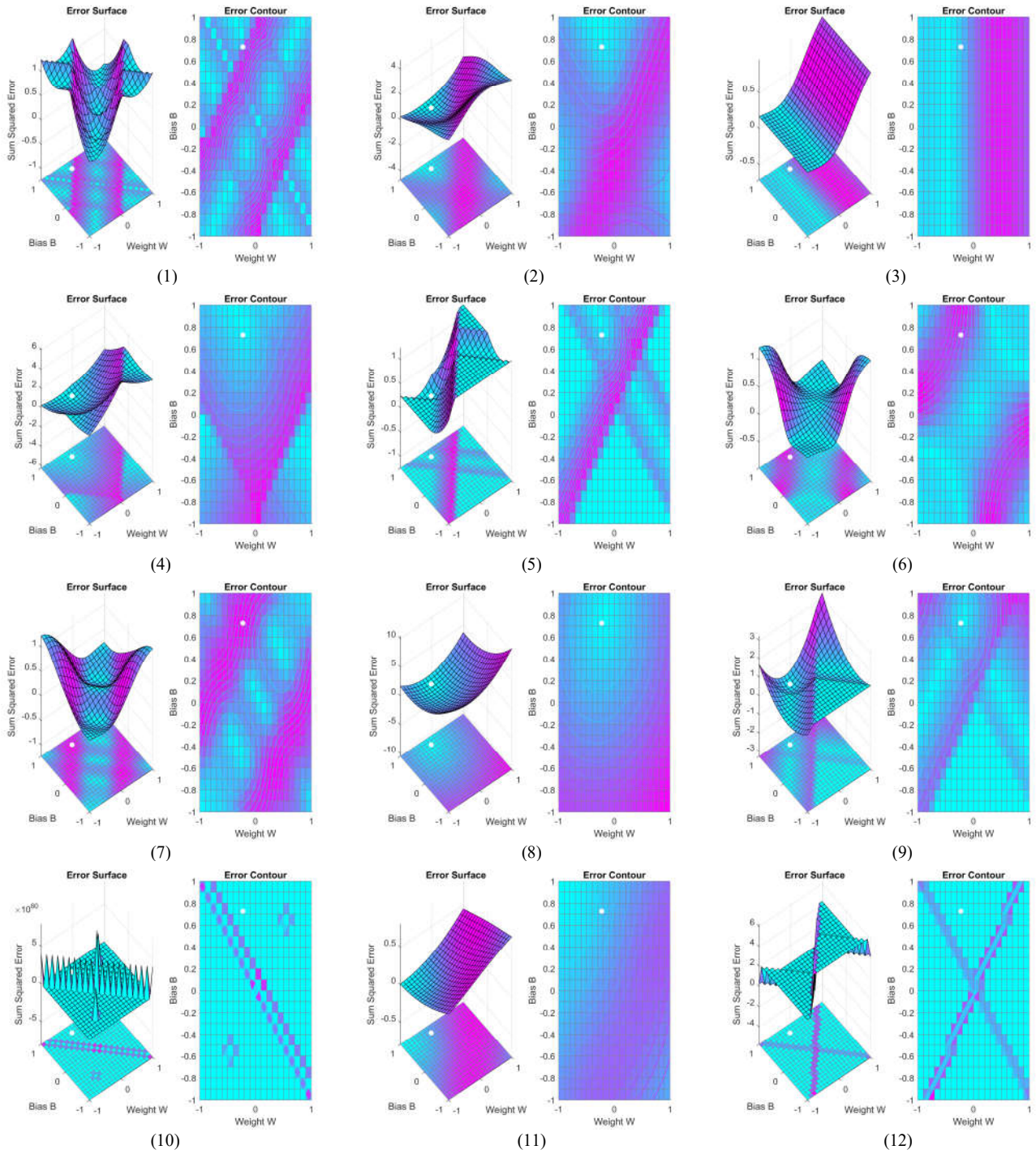

Fig. 6. Loss landscapes of the neural network model with multiple activation functions.

the multi-layer deep neural network models, there exists complex functional relationship between the output of the upper node and the input of the underlying node. This function is called an activation function (also called an excitation function).

At present, the most successful and widely-used activation function is the Rectified Linear Unit (ReLU), which is defined as

$$f(x) = \max(x, 0) \tag{8}$$

The use of ReLU function was a breakthrough that enabled the fully supervised training of the state-of-the-art deep networks. Deep networks with ReLUs are more easily optimized than networks with sigmoid or tanh units (which are defined in Eq. (9) and Eq. (10), respectively), because gradient is able to flow when the input to the ReLU function is positive. Thanks to its simplicity and effectiveness, ReLU has become the default activation function used across the deep learning community.

$$S(x) = \frac{1}{1+e^{-x}} \tag{9}$$

$$T(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{10}$$

Softplus can be seen as a smoothed version of ReLU, which is defined as

$$P(x) = \log(1 + e^x) \tag{11}$$

ReLU function sets all negative values to zero. Instead, Leaky-ReLU assigns a non-zero slope to all the negative values. The Leaky-ReLU activation function was first proposed in the acoustic model and was defined as

$$L(x) = \begin{cases} x, if\, x > 0 \\ \dfrac{x}{\alpha}, if\ x \le 0 \end{cases} \tag{12}$$

where $\alpha \in (0, +\infty)$ is a fixed parameter.

Then we calculate the derivatives of the following five activation functions, as shown below.

$$S'(x) = S(x)\left[1 - S(x)\right] \tag{13}$$

$$f'(x) = \begin{cases} 1, x > 0 \\ 0, x \le 0 \end{cases} \tag{14}$$

$$T'(x) = 1 - T^2(x) \tag{15}$$

$$P'(x) = \frac{e^x}{1 + e^x} \tag{16}$$

$$L'(x) = \begin{cases} 1, x > 0 \\ \dfrac{1}{\alpha}, x \le 0 \end{cases} \tag{17}$$

The output of sigmoid function is not zero-centered, which reduces the efficiency of weight updating. When the input is slightly away from the coordinate origin, the gradient of this function becomes very small (*i.e.*, almost equaling to zero). Finally, this will lead to the weight has little effect on the loss function, which is not conducive to the optimization of network weight, resulting in gradient saturation problem. When the weight of deep neural network is initialized to a value in the (1, +∞) interval, there will be the gradient explosion problem. On the other hand, the output of sigmoid function is not zero-centered. This is undesirable because it will cause the neurons in the latter layer to receive the signals of the non-zero mean output from the upper layer as input. One result is that if the data enters the neuron positively (*e.g.*, elementwise), the calculated gradient will always be positive.

In the specific applications, the tanh function is usually superior to the sigmoid function, mainly due to the sigmoid function is sensitive to changes in function values when the input is between interval [0, 1], and loses sensitivity once it approaches or exceeds interval in a saturated state, affecting the accuracy value predicted by the neural network. And the output and input of tanh can maintain a nonlinear monotonic rise. However, it also has the problem of gradient saturation. Generally, in binary classification problems, tanh function is usually used in the hidden layer and sigmoid function is used in the output layer.

The ReLU function is a popular activation function. When the input is positive, there is no gradient saturation problem. ReLU function has only the linear relationship. Whether it is on forward or backward propagation stage, it is much faster than sigmoid and tanh functions. However, it also has some drawbacks. If inputs are negative, ReLU is not to be activated at all, which means that once a negative number is entered, ReLU will have zero output. PReLU is an improved version

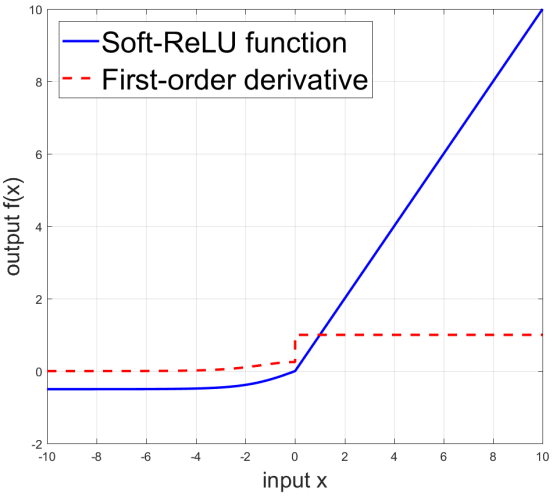

Fig. 7. Popular activation functions in deep neural networks and their corresponding derivatives.

of ReLU. In the negative region, PReLU has a small slope, which can be used to avoid the problem of neuronal necrosis in ReLU.

If researchers do not use the activation function (actually equivalent to the activation function is $f(x) = x$). In this case, the input of each node in one layer is a linear function of the output of the upper layer. It is easy to verify that no matter how many layers of the neural network has, the output is a linear combination of the inputs, which is equivalent to the effect of no hidden layers. This means that your model is the most primitive perception machine (*i.e.*, perceptron). Then the approximation ability of the network is quite limited.

The primary role of the activation functions in deep convolutional neural networks is to provide the nonlinear modeling capabilities of the network. Assuming that a deep neural network model contains only linear convolution and full connection operation, the network can only express linear mapping. Even if the depth of the network is increased, it is still linear. It is difficult to effectively tune the nonlinearly distributed data in the actual environment. After adding the (non-linear) activation function, the deep neural network has the hierarchical non-linear mapping ability. Therefore, the activation function is an essential component of the deep neural network model.

In fact, PReLU is also a variant of Leaky-ReLU. In the PReLU function, the slope of the negative value is trained rather than pre-defined. Finally, we plot the curves of all activation functions and their corresponding derivatives, as shown in Fig. 5.

Generally speaking, the above activation functions have their own advantages and disadvantages. At present, there are no conclusions that which activation functions are good or bad, which must be obtained according to experiments.

C. *Property of Activation Functions*

Typically, loss landscapes of deep convolutional neural network model can reflect the nature and property of the activation function. In the past, a large number of linear or non-linear dimension reduction methods have been developed to observe the parameter distribution and training process of the model. However, some traditional methods such as PCA dimension reduction [28] and LDA dimension reduction [29] are difficult to observe the distribution of

TABLE I
CLASSIFICATION RESULTS OF VARIOUS ACTIVATION FUNCTIONS ON FOUR IMAGE CLASSIFICATION BENCHMARKS ON ALEXNET

Activation Functions	MNIST	CIFAR10	CIFAR100	ImageNet
ReLU	99.27%	93.11%	77.9%	76.4%
PReLU	99.46%	95.43%	81.7%	77.2%
Leaky ReLU	99.37%	94.42%	81.4%	76.8%
Softplus	99.41%	93.37%	79.2%	74.5%
ELU	99.17%	92.27%	78.8%	74.7%
MPELU	98.85%	92.84%	76.8%	74.1%
Swish	99.43%	94.47%	80.4%	76.7%
Ours	99.27%	94.81%	79.4%	75.5%

TABLE II
CLASSIFICATION RESULTS OF VARIOUS ACTIVATION FUNCTIONS ON FOUR IMAGE CLASSIFICATION BENCHMARKS ON VGGNET

Activation Functions	MNIST	CIFAR10	CIFAR100	ImageNet
ReLU	99.31%	93.20%	78.2%	76.7%
PReLU	99.49%	95.47%	82.2%	77.4%
Leaky ReLU	99.36%	94.46%	81.6%	76.4%
Softplus	99.44%	93.33%	79.5%	74.8%
ELU	99.22%	92.40%	79.0%	75.1%
MPELU	98.87%	93.01%	77.2%	74.6%
Swish	99.76%	94.66%	80.7%	77.4%
Ours	99.33%	95.52%	82.9%	77.8%

TABLE III
CLASSIFICATION RESULTS OF VARIOUS ACTIVATION FUNCTIONS ON FOUR IMAGE CLASSIFICATION BENCHMARKS ON GOOLENET

Activation Functions	MNIST	CIFAR10	CIFAR100	ImageNet
ReLU	99.33%	93.32%	79.1%	76.8%
PReLU	99.34%	95.41%	82.1%	77.2%
Leaky ReLU	99.32%	95.57%	83.1%	77.5%
Softplus	99.47%	93.51%	79.7%	75.4%
ELU	99.36%	93.17%	79.2%	74.7%
MPELU	98.90%	93.22%	78.4%	75.1%
Swish	99.74%	94.41%	81.1%	77.7%
Ours	99.45%	95.37%	82.4%	77.1%

network parameters effectively because of the severe over-parameterization of deep convolution neural networks. The reduction of massive dimensions makes the information seriously distorted, and it is difficult to reflect the training process and convergence results of the model.

Therefore, we plan to use a reverse exploratory approach to visualize the loss landscape of the model. We use the final convergence position to explore the loss value in the random direction, so as to form a two-dimensional loss landscape under the two dimensions of network weight and bias.

Finally, we plot the loss planes of the model with multiple activation functions, as shown in Fig. 6. Among them, (1)-(12) represent the radbasn, radbas, purelin, logsig, hardlims, tribas, tansig, softmax, satlins, satlin, linear, and sawtooth function , respectively. By comparing the visualization results of loss landscapes of deep convolutional neural networks under multiple various activation functions, it can be clearly seen that the different distributions are obtained due to different properties. Some functions have smoother boundaries than the others, while they are jagged, such as (10) and (12). On the other hand, the degree of nonlinearity of boundary can also reflect their fitting capability: the fitting ability of activation functions (1) and (7) is much higher than that of (2) and (3). However, in practical application, a higher fitting ability does not mean a better classification performance. The over-fitting problem caused by the over high fitting ability is a typical example. In fact, how to achieve a balance between model optimization and generalization is the key to solve the problem at present.

D. Piecewise Bi-Nonlinear Activation Function

Based on the analysis of above activation functions, we try to design a new function to balance the network optimization and generalization. The new proposed activation function (soft-ReLU) utilizes both their advantages of several different activation functions, and it is defined as

$$f_{act}(x)=\begin{cases} x, & x \geq 0 \\ \dfrac{1}{1+e^{-x}}, & x < 0 \end{cases} \tag{18}$$

TABLE IV
CLASSIFICATION RESULTS OF VARIOUS ACTIVATION FUNCTIONS ON FOUR IMAGE CLASSIFICATION BENCHMARKS ON NETWORK IN NETWORK

Activation Functions	MNIST	CIFAR10	CIFAR100	ImageNet
ReLU	99.21%	93.29%	78.4%	76.1%
PReLU	99.36%	95.37%	80.8%	75.4%
Leaky ReLU	99.21%	95.50%	82.3%	77.2%
Softplus	99.25%	93.42%	78.4%	75.1%
ELU	99.19%	93.10%	79.0%	74.0%
MPELU	98.77%	93.07%	78.1%	73.9%
Swish	99.53%	94.28%	80.4%	74.6%
Ours	99.33%	95.15%	81.1%	75.2%

TABLE V
CLASSIFICATION RESULTS OF VARIOUS ACTIVATION FUNCTIONS ON FOUR IMAGE CLASSIFICATION BENCHMARKS ON RESNET-101

Activation Functions	MNIST	CIFAR10	CIFAR100	ImageNet
ReLU	99.32%	93.23%	78.6%	75.8%
PReLU	99.44%	95.18%	81.1%	76.1%
Leaky ReLU	99.07%	95.07%	82.0%	76.7%
Softplus	99.34%	93.66%	79.8%	75.5%
ELU	99.46%	93.20%	79.4%	74.7%
MPELU	98.94%	93.41%	79.2%	74.2%
Swish	99.38%	94.33%	81.1%	74.2%
Ours	99.29%	95.07%	80.8%	74.7%

TABLE VI
CLASSIFICATION RESULTS OF VARIOUS ACTIVATION FUNCTIONS ON FOUR IMAGE CLASSIFICATION BENCHMARKS ON WIDE RESNET

Activation Functions	MNIST	CIFAR10	CIFAR100	ImageNet
ReLU	99.51%	93.20%	79.2%	73.9%
PReLU	99.34%	94.85%	80.4%	74.4%
Leaky ReLU	99.20%	94.72%	81.2%	75.3%
Softplus	99.41%	93.28%	79.6%	75.0%
ELU	99.40%	93.44%	79.2%	74.1%
MPELU	99.12%	93.30%	79.7%	74.4%
Swish	99.07%	94.12%	80.4%	73.4%
Ours	99.40%	94.88%	79.4%	74.4%

where $f_{act}(\cdot)$ represents the proposed soft-ReLU function. The first order derivative can be obtained by

$$f'_{act}(x)=\begin{cases}1, & x\geq 0\\ \dfrac{e^{-x}}{(1+e^{-x})^2}, & x<0\end{cases} \tag{19}$$

Finally, we present the function curves and its first order derivative in Fig. 7. It can be clearly seen that proposed soft-ReLU is a piecewise bi-nonlinear activation function. When $x > 0$, the gradient is always kept at 1; while $x < 0$, the gradient begins to disappear at the great distance, which avoids the neuronal necrosis phenomenon while ensuring over-effect.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

In this section, we perform a large number of experiments on four image classification benchmark datasets (*i.e.* MNIST, CIFAR10, CIFAR100, and ImageNet) by using four deep neural network models (*i.e.* AlexNet, VGGNet, GoogLeNet, and NIN) to observe the nature and performance of different activation functions, including ReLU, PReLU, Leaky ReLU, Softplus, ELU, MPELU, Swish, and Soft-ReLU functions. Finally, the experimental results and detailed analysis are presented in each section.

A. Experimental Setup

In this part, we first introduce the experimental settings, including the introduction of the database, the selection of network model, and the setting of hyper-parameters.

Experimental Datasets for Image Classification. We select four image classification datasets in the experimental process. The MNIST handwritten dataset [30] is from the National Institute of Standards and Technology (NIST). The training set consists of handwritten images from 250 different people, 50% of whom are high school students and 50% of staff from the Census Bureau. It includes a total of 60 000 images from 0 to 9. The test set is also the same proportion of handwritten digital data, which contains a total of 10 000 images. The CIFAR-10 dataset [31] consists of 60 000 32×32

TABLE VII
MNIST: CLASSIFICATION RESULTS OF VARIOUS ACTIVATION FUNCTIONS ON ALEXNET UNDER VARIOUS INITIALIZATION METHODS

Activation Functions	LSUV [39]	Xavier [40]	MSRA [41]	Gaussian Distribution
ReLU	99.37%	99.14%	99.15%	99.24%
PReLU	99.23%	99.52%	99.39%	99.34%
Leaky ReLU	99.42%	99.25%	98.99%	98.97%
Softplus	99.18%	99.40%	99.03%	99.17%
ELU	99.04%	98.66%	99.21%	99.02%
MPELU	98.65%	99.10%	98.91%	98.92%
Swish	99.59%	99.66%	99.75%	99.44%
Ours	99.32%	99.46%	99.15%	99.34%

TABLE VIII
CIFAR10: CLASSIFICATION RESULTS OF VARIOUS ACTIVATION FUNCTIONS ON ALEXNET UNDER VARIOUS INITIALIZATION METHODS

Activation Functions	LSUV [39]	Xavier [40]	MSRA [41]	Gaussian Distribution
ReLU	92.95%	93.02%	92.90%	93.05%
PReLU	95.47%	95.78%	95.83%	95.67%
Leaky ReLU	95.26%	95.15%	95.00%	95.43%
Softplus	93.57%	93.75%	93.69%	93.83%
ELU	93.29%	93.14%	93.42%	92.78%
MPELU	94.23%	93.57%	93.44%	92.83%
Swish	94.73%	93.86%	93.86%	93.92%
Ours	95.10%	95.09%	95.05%	94.79%

TABLE IX
CIFAR100: CLASSIFICATION RESULTS OF VARIOUS ACTIVATION FUNCTIONS ON ALEXNET UNDER VARIOUS INITIALIZATION METHODS

Activation Functions	LSUV [39]	Xavier [40]	MSRA [41]	Gaussian Distribution
ReLU	78.97%	78.50%	79.00%	78.88%
PReLU	81.18%	80.68%	81.54%	80.99%
Leaky ReLU	82.05%	81.74%	81.99%	81.74%
Softplus	79.45%	79.42%	78.79%	79.70%
ELU	79.75%	79.08%	79.24%	78.99%
MPELU	79.32%	78.94%	79.60%	78.83%
Swish	80.95%	81.02%	80.97%	81.54%
Ours	80.81%	80.35%	80.41%	81.26%

TABLE X
IMAGENET: CLASSIFICATION RESULTS OF VARIOUS ACTIVATION FUNCTIONS ON ALEXNET UNDER VARIOUS INITIALIZATION METHODS

Activation Functions	LSUV [39]	Xavier [40]	MSRA [41]	Gaussian Distribution
ReLU	73.99%	74.09%	74.15%	73.88%
PReLU	73.83%	74.70%	73.99%	74.32%
Leaky ReLU	74.96%	75.49%	75.13%	75.23%
Softplus	74.81%	74.94%	75.16%	74.75%
ELU	74.52%	74.16%	74.46%	74.11%
MPELU	73.77%	74.13%	73.86%	74.41%
Swish	72.81%	73.72%	73.96%	73.81%
Ours	73.97%	73.99%	74.76%	74.78%

color images of 10 classes, each with 6000 images. There are 50 000 training images and 10 000 test images in CIFAR 10. CIFAR100 [32] has 100 classes, each of which containing 600 images where includes 500 training images and 100 test images. ImageNet [33] is a computer vision based recognition project and is the largest database for image recognition in the world, which contains a total of 1 000 000 natural images in 1000 categories.

Deep Convolutional Neural Networks. AlexNet [34], VGGNet [35], GoogLeNet [36], NIN [37], ResNet [38], and Wide ResNet [39] with different activation functions are used to perform image classification task. The above six models include 17, 19, 22, 14, 101, and 23 layers, respectively. In fact, comparing the classification performance of many activation functions across various models can comprehensively reflect their properties.

Hyper-parameters Setting. The models are initialized by a Gaussian distribution function. Learning rate is set to 0.01

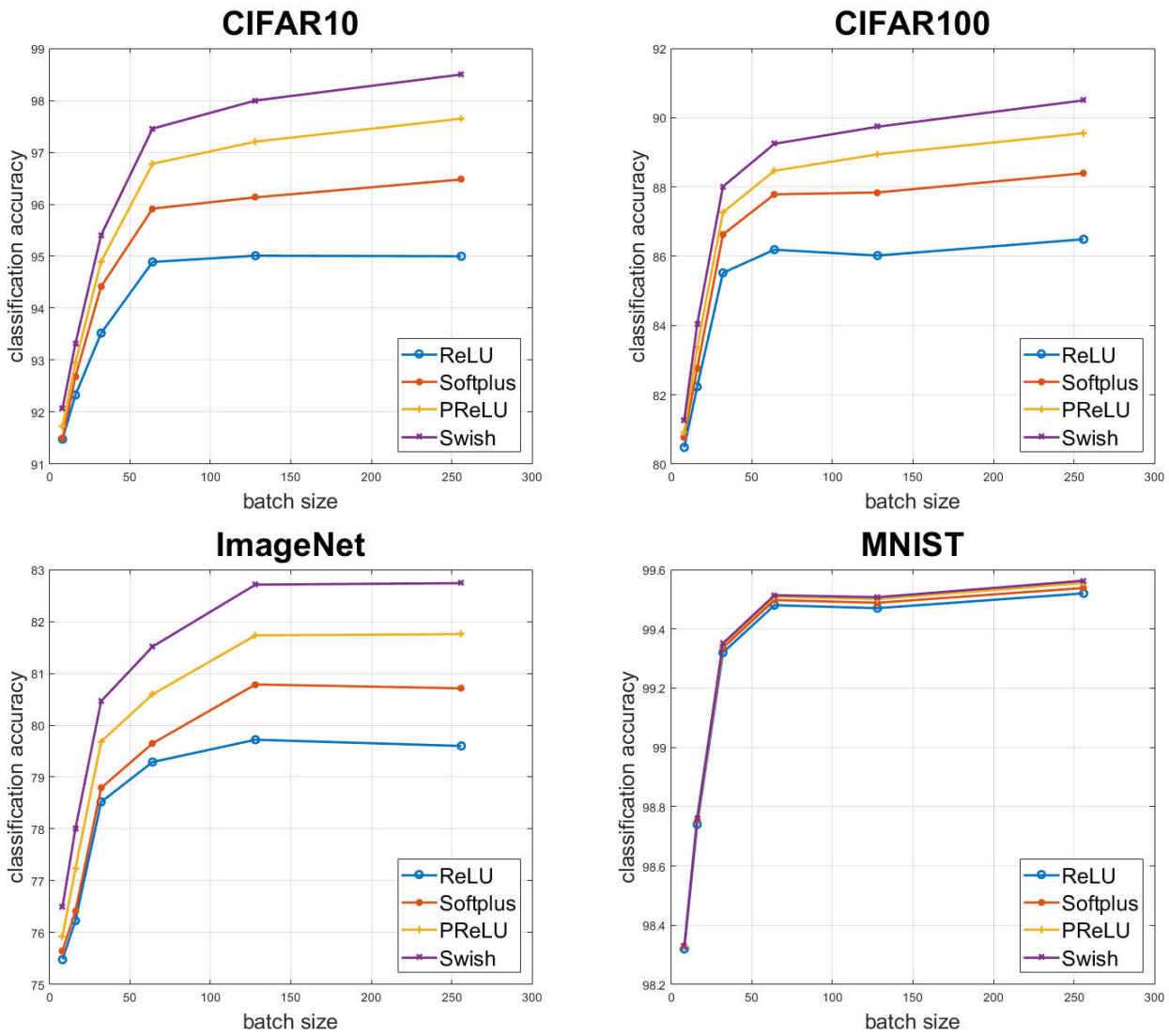

Fig. 8. Classification performance of four activation functions on four image datasets.

and gradually decreases, which can accelerate the optimizing speed and eventually converge to a better position. Then, momentum is set to 0.9 for alleviating the interference of local minimum, and the weight decay is set to 0.0005 to avoid the over-fitting problem of deep convolutional neural network. Moreover, Dropout [38] is used to improve the generalization ability of deep CNN and the rate is set to 0.5 while it is opened in the training process.

B. Classification Performance

We calculate the performance of the six models on the four image classification benchmarks, as shown in Table I to Table VI. In the model AlexNet, PReLU function almost achieves the best results on both four datasets: 99.46%, 95.43%, 81.7%, and 77.2% on MNIST, CIFAR10, CIFAR100 and ImageNet datasets, respectively. The performance of Swish is also better than others. The performance of our proposed activation function Soft-ReLU is acceptable, exceeding the performance of Leaky ReLU and ELU on CIFAR10/100 and ImageNet datasets. In the deep convolutional neural network VGGNet, Soft-ReLU performs best on three datasets: 95.52%, 82.9%, and 77.8% on CIFAR10, CIFAR100, and ImageNet. Swish performs best on MNIST and achieves the classification accuracy of 99.76%. Most of the activation functions perform similarly on the MNIST dataset, which is caused by the simple background of samples in MNIST. In the GoogLeNet, inception architecture plays a key role in the training process. Leaky ReLU has good performance on CIFAR10 (95.57%) and CIFAR 100 (83.1%) while Swish function achieves the best classification accuracy on ImageNet (77.7%). In the NIN network model, various activation functions have their own advantages and disadvantages on different datasets (Swish: 99.53% on MNIST, PReLU: 99.50% on CIFAR 10, Leaky ReLU function: 82.3% on CIFAR 100, and ReLU: 76.1% on ImageNet). In the ResNet 101, the short-connection structure makes optimization of ultra-deep neural networks possible. Wide ResNet proves that even a lighter neural network model can achieve good results with Soft-ReLU activation function: 99.40%, 94.88%, 79.4%, and 74.4% on MNIST, CIFAR10, CIFAR100, and ImageNet.

C. Influence of Network Initialization

In this part, we observe the impact of network initialization on the optimization and generalization of deep convolutional neural networks, including LSUV [39], Xavier [40], MSRA [41], and Gaussian distribution. The classification results on four image classification benchmarks under the AlexNet are shown in Table VII to Table X.

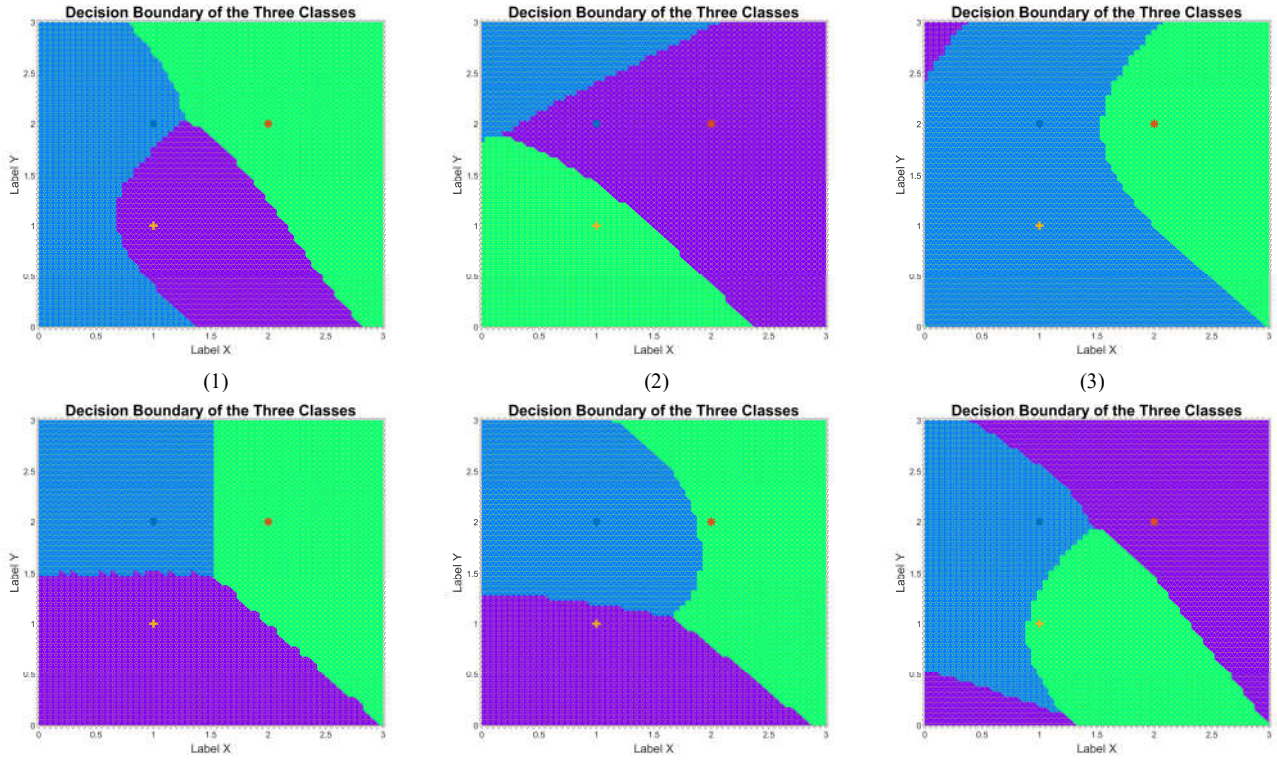

Fig. 9. Decision boundary of deep CNN under various activation functions.

The idea of Xavier is to make the variance of the input weight of a neuron (the output when backpropagating) equal to the reciprocal of number of inputs, the purpose of which is to allow gradient information to be evenly distributed across the network. If we pay more attention to forward propagation, we can choose the number of inputs, that is, the number of inputs for forward propagation; if we pay more attention to backward propagation, we choose the number of outputs. If both are considered, we can set the average of the two term as the final result. MSRA initializes the weights as the Gaussian distribution with a mean of 0 and a variance of 2 / input, which is also different from Xavier filler; it is especially suitable for the ReLU activation function.

According to the experimental results, it can be seen that there is almost no difference in the different initializations, that is, there is no obvious influence of initialization on the optimization and generalization of the deep neural network. This is due to the advantages brought by large-scale data. The network model can easily avoid the local minimum under the training of large-scale data, except for some extremely poor initialization positions [43][44].

D. Hyper-parameter Sensitivity and Visualization of Decision Boundary

In this part, we explore the hyper-parameter sensitivity to activation function, such as the batch size. The classification results under multi various batch size are shown in Fig. 8. It can be seen that as batch size increases, the performance of each activation function gradually increases, and tends to be stable at the batch size of 128. On the other hand, the Swish performs better than other three activation functions (ReLU, Softplus, and PReLU) on the four datasets.

In general, within a reasonable range, the larger the batch size makes the gradient drop direction more accurate, then the smaller the oscillating will be; on the other hand, if the batch size is too large, a local minimum may occur. Small bath size usually introduce more randomness, it is difficult to achieve convergence, and in rare cases it may work better.

Next, we observe the decision boundary of neural network with different activation functions on a two dimensional toy dataset, as shown in Fig. 9. Graphs (1) to (6) in the Fig. 9 represent the ReLU, PReLU, Leaky ReLU, Swish, Softplus, and Soft-ReLU functions, respectively. According to the first order derivative and smoothness of the boundary of neural networks, we can observe the properties of the corresponding activation function.

Furthermore, we calculate and observe weight distribution of different activation functions on four deep CNN models, including AlexNet, VGGNet, GoogLeNet and NIN, as shown in Fig. 10. From the point of view of minimizing structural risk, we choose the simplest structure as the best network in many models, rather than the most complex one with the most discrete weight distribution.

V. Conclusion

Here, we try to conclude the useful properties of activation functions in deep learning:

- Nonlinear: the derivative is not a constant. It is the foundation of the multi-layer deep neural network that guarantees that the neural network does not degenerate into a single-layer linear network.
- Then the differentiability guarantees the computability of overall gradient in optimization. For the SGD algorithm, since it is almost impossible to converge to the location near zero, the limited non-differentiable points will not have a great impact on the optimization results.

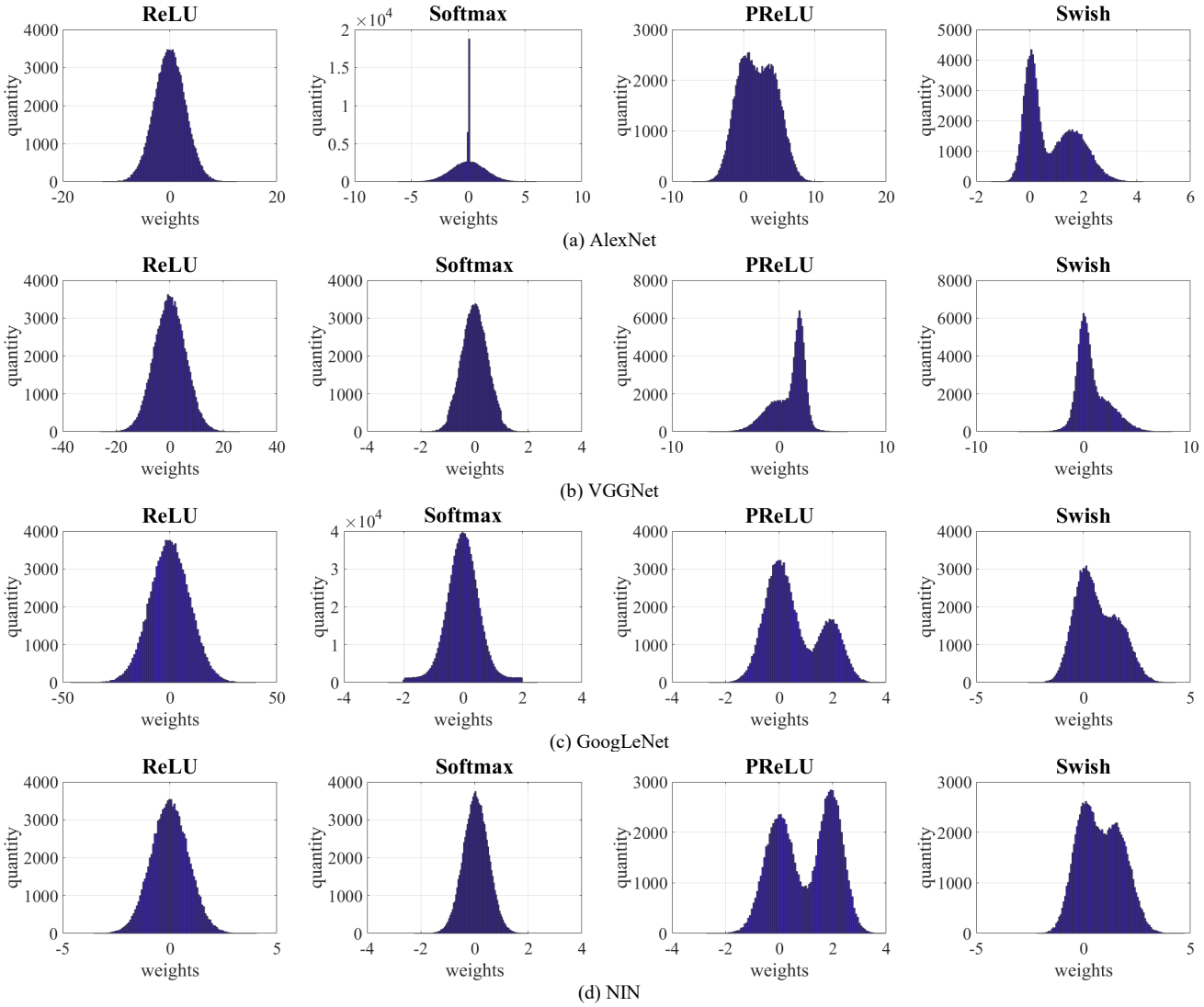

Fig. 10. Weight distribution of different activation functions on four deep CNN models, including AlexNet, VGGNet, GoogLeNet, and NIN.

- Simple calculation property. The number of calculations of activation function in the forward direction of the deep neural network is proportional to the number of neurons, so the simple nonlinear function is more suitable as a proper activation function than complicated one.
- Non-saturation. Saturation refers to the problem that the gradient is close to zero (*i.e.*, the gradient disappears) in some intervals, resulting in the network parameters unable to continue to update.
- Monotonic property. The monotonicity usually makes the overall gradient direction at the activation function not change, making training easier to converge.
- Limited output range. The limited output range makes the deep convolutional neural network more stable for some larger inputs.
- Identity. This has the advantage that the amplitude of the output does not increase significantly with increasing depth, making the network more stable and the gradients can be more easily returned.
- Normalization. The main idea is to automatically normalize the sample distribution to the distribution of zero mean and uniform variance, thus stabilizing the training process.

At present, there are no conclusions in academia which activation functions are better than others or which properties are more important than others. We hope that a large number of experiments can be used to analyze which properties of activation function are conducive to improving the network's optimization and generalization ability, such as monotonicity, smoothness, unbounded above, and bounded below.

REFERENCES

[1] J. B. Bayer and S. W. Campbell, "Texting while driving on automatic: Considering the frequency-independent side of habit," Computers in Human Behavior, vol. 28, no. 6, pp. 2083-2090, 2012.

[2] L. Eciolaza, M. Pereira-Fariña, and G. Trivino, "Automatic Linguistic reporting in driving simulation environments," Applied Soft Computing Journal, vol. 13, no. 9, pp. 3956-3967, 2013.

[3] Q. Zheng, M. Yang, Q. Zhang, and J. Yang, "A Bilinear Multi-Scale Convolutional Neural Network for Fine-grained Object Classification," IAENG International Journal of Computer Science, vol. 45, no. 2, pp. 340-352, 2018.

[4] Q. Zheng, M. Yang *et al.*, "Improvement of Generalization Ability of Deep CNN via Implicit Regularization in Two-Stage Training Process," IEEE Access, vol. 6, pp. 15844-15869, 2018.

[5] Q. Zheng, M. Yang *et al.*, "Understanding and Boosting of Deep Convolutional Neural Network Based on Sample Distribution," in IEEE 2nd Information Technology, Networking, Electronic and Automation Control Conference, Chengdu, China, pp. 823-827, 2017.

[6] D. S. Kermany *et al.*, "Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning," Cell, vol. 172, no. 5, pp. 1122–1131.e9, 2018.

[7] K. Noda *et al.*, "Audio-visual speech recognition using deep learning," Applied Intelligence, vol. 42, no. 4, pp. 722-737, 2015.

[8] Z. Zhang *et al*., "Deep Learning for Environmentally Robust Speech Recognition: An Overview of Recent Developments," Acm Trans. on Intelligent Systems & Technology, vol. 9, no. 5, pp. 1-28, 2017.
[9] Q. Zheng, M. Yang, Q. Yang, and X. Zhang, "Fine-grained Image Classification Based on the Combination of Artificial Features and Deep Convolutional Activation Features," IEEE/CIC ICCC, pp. 1-6, 2017.
[10] Q. Nguyen *et al.*, "Effect of Word Sense Disambiguation on Neural Machine Translation: A Case Study in Korean," IEEE Access, vol. 6, pp. 38512 – 38523, 2018.
[11] Z. Rui *et al.*, "Deep learning and its applications to machine health monitoring," Mechanical Systems & Signal Processing, vol. 115, pp. 213-237, 2019.
[12] N. Leal, E. Leal, and S. German, "A Linear Programming Approach for 3D Point Cloud Simplification," IAENG International Journal of Computer Science, vol. 44, no. 1, pp. 60-67, 2017.
[13] V. Pomero *et al.*, "Fast accurate stereoradiographic 3D-reconstruction of the spine using a combined geometric and statistic model," Clinical Biomechanics, vol. 19, no. 3, pp. 240-247, 2004.
[14] X. Yin *et al.*, "A Flexible Sigmoid Function of Determinate Growth," Annals of Botany, vol. 91, no. 6, pp. 753-753, 2003.
[15] E. G. Fan, "Extended Tanh-function Method and its Applications to Nonlinear Equations," Physics Letters A, vol. 277, no. 4, pp. 212-218, 2000.
[16] S. J. Arrowsmith *et al.*, "A seismoacoustic study of the 2011 January 3 Circleville earthquake," Geophysical Journal International, vol. 189, no. 2, pp. 1148-1158, 2012.
[17] S. H. Wang et al., "Classification of Alzheimer's Disease Based on Eight-Layer Convolutional Neural Network with Leaky Rectified Linear Unit and Max Pooling," Journal of Medical Systems, vol. 42, no. 5, pp. 85-95, 2018.
[18] A. Krizhevsky, I. Sutskever, G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in Proceedings of Neural Information Processing System, Lake Tahoe, USA, pp. 1097-1105, 2012.
[19] K. Fukushima, "Neocognitron:a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," Biological Cybernetics, vol. 36, no. 4, pp. 193 -202, 1980.
[20] Y. Lecun *et al.*, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, 1998.
[21] P. Simard *et al.*, "Transformation Invariance in Pattern Recognition – Tangent Distance and Tangent Propagation," International Journal of Imaging Systems & Technology, vol. 11, no. 3. Pp. 181-197, 2001.
[22] A. Anas and L. Yu, "A Regional Economy, Land Use, and Transportation Model (ReLU-Tran©): Formulation, Algorithm Design, and Testing,". Journal of Regional Science, vol. 47, no. 3, pp. 415-455, 2010.
[23] C. Özkan and F. S. Erbek, "The Comparison of Activation Functions for Multispectral Landsat TM Image Classification," Photogrammetric Engineering & Remote Sensing, vol. 69, no. 11, pp. 1225-1234, 2003.
[24] Q. Zheng *et al.*, "Static Hand Gesture Recognition Based on Gaussian Mixture Model and Partial Differential Equation," IAENG International Journal of Computer Science, vol. 45, no. 4, pp. 569-583, 2018.
[25] Q. Zheng and M. Yang, "A Video Stabilization Method based on Inter-Frame Image Matching Score," Global Journal of Computer Science and Technology, vol. 17, no. 1, pp. 35-40, 2017.
[26] F. Cao, L. Bo, and S. P. Dong, "Image classification based on effective extreme learning machine," Neurocomputing, vol. 102, no. 2, pp. 90-97, 2013.
[27] S. Ladhani *et al.*, "Robust activation function and its application: Semi-supervised kernel extreme learning method," Neurocomputing, vol. 144, no. 1, pp. 318-328, 2014.
[28] A. N. Kashif, Z. A. Aziz, F. Salah, and K.K. Viswanathan, "Convective Heat Transfer in the Boundary Layer Flow of a Maxwell Fluid Over a Flat Plate Using an Approximation Technique in the Presence of Pressure Gradient," Engineering Letters, vol. 26, no. 1, pp. 14-22, 2018.
[29] G. Liu, X. Chen, C. Nie, and H. Yu, "Constructions of Normal Extended Functions for Elliptic Interface Problems," IAENG International Journal of Applied Mathematics, vol. 47, no. 3, pp. 271-275, 2017.
[30] E. M. Kussul and T. Baidyk, "Improved method of handwritten digit recognition tested on MNIST database," Image & Vision Computing, vol. 22, no. 12, pp. 971-981, 2004.
[31] H. Li *et al.*, "CIFAR10-DVS: An Event-Stream Dataset for Object Classification," Frontiers in Neuroscience, vol. 11, pp. 309-318, 2017.
[32] Q. Zhang, M. Yang, K. Kpalma, Q. Zheng, and X. Zhang, "Segmentation of Hand Posture against Complex Backgrounds Based on Saliency and Skin Colour Detection," IAENG International Journal of Computer Science, vol. 45, no. 3, pp. 435-444, 2018.
[33] M. Guillaumin, D. Küttel, and V. Ferrari, "ImageNet Auto-Annotation with Segmentation Propagation," International Journal of Computer Vision, vol. 110, no. 3, pp. 328-348, 2014.
[34] B. Moons and M. Verhelst, "An Energy-Efficient Precision-Scalable ConvNet Processor in 40-nm CMOS," IEEE Journal of Solid-State Circuits, vol. 52, no. 4, pp. 903-914, 2017.
[35] H. Ke *et al.*, "Towards Brain Big Data Classification: Epileptic EEG Identification With a Lightweight VGGNet on Global MIC," IEEE Access, vol. 6, pp. 14722-14733, 2018.
[36] Z. Xiao *et al.*, "Airport Detection Based on a Multiscale Fusion Feature for Optical Remote Sensing Images," IEEE Geoscience & Remote Sensing Letters, vol. 14, no. 9, pp. 1469-1473, 2017.
[37] H. Zhuang, M. Yang, Z. Cui, and Q. Zheng, "A Method for Static Hand Gesture Recognition Based on Non-Negative Matrix Factorization and Compressive Sensing," IAENG International Journal of Computer Science, vol. 44, no. 1, pp. 52-59, 2017.
[38] Z. Lu, X. Jiang, and A. Kot, "Deep coupled ResNet for low-resolution face recognition," IEEE Signal Processing Letters, vol. 25, no. 4, pp. 526-530, 2018.
[39] G. Czukor and M. Bayazit, "Casting a Wide Net? Performance Deficit, Priming, and Subjective Performance Evaluation in Organizational Stereotype Threat Research," Industrial & Organizational Psychology, vol. 7, no. 3, pp. 409-413, 2017.
[40] S. M. Tedjojuwono, "Virtual Lines Sensors for Moving Object and Vehicle Counters," IAENG International Journal of Computer Science, vol. 44, no. 4, pp. 421-431, 2017.
[41] K. He *et al.*, "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," in IEEE International Conference on Computer Vision, Santiago, Chile, pp. 1026-1034, 2015.
[42] Q. Zheng, X. Tian, M. Yang, and H. Wang, "Differential Learning: A Powerful Tool for Interactive Content-Based Image Retrieval," Engineering Letters, vol. 27, no. 1, pp. 202-215, 2019.
[43] Q. Zheng, X. Tian, M. Yang, and S. Liu, "Near-infrared Image Enhancement Method in IRFPA Based on Steerable Pyramid," Engineering Letters, vol. 27, no. 2, pp. 352-363, 2019.
[44] Q. Zheng, X. Tian, M. Yang, and H. Wang, "The Email Author Identification System Based on Support Vector Machine (SVM) and Analytic Hierarchy Process (AHP)," IAENG International Journal of Computer Science, vol. 46, no. 2, pp. 178-191, 2019.
[45] Q. Zhang *et al.*, "Segmentation of hand gesture based on dark channel prior in projector-camera system," in IEEE/CIC ICCC, Qingdao, China, pp. 1-6, 2017.
[46] Q. Zheng, X. Tian, M. Yang, Y. Wu *et al.*, "PAC-Bayesian framework based drop-path method for 2D discriminative convolutional network pruning," Multidimensional Systems and Signal Processing, early access, 2019. DOI: 10.1007/s11045-019-00686-z
[47] Q. Zheng, X. Tian, N. Jiang, and M. Yang, "Layer-wise learning based stochastic gradient descent method for the optimization of deep convolutional neural network," Journal of Intelligent & Fuzzy Systems, vol. 37, no. 4, pp. 5641-5654, 2019.

Qinghe Zheng was born in Jining, Shandong, China in 1993. He received his B.S. degree from Xi'an University of Posts and Telecommunications in 2014 and M.S. degree from Shandong University in 2018. Now, he is studying for his Ph.D in Shandong University. His research direction is computer vision and machine learning.