Journal of Computers Vol. 36 No. 1, February 2025, pp. 43-57
https://doi.org/10.63367/19911599202502360100 4

Based on the Depth-Separable Convolution of Lightweight Networks

Qi-Wen Zhang, Ying Li*, and Wen-Kui Wu

School of Computer and Communication, Lanzhou University of Technology,
Lanzhou, 730050, China
{823869941, 2730948254, 2059394530}@qq.com

Received 18 April 2024; Revised 27 August 2024; Accepted 1 October 2024

Abstract. Currently, the ubiquitous 1×1 convolution operation makes the network face the challenge of high computational complexity. At the same time, many models are dependent on data, and there are obvious differences between small sample data, which reduces the classification ability of the model. To this end, this paper proposes a lightweight network based on depth-wise separable convolution, which improves the classification ability of the model by reducing the complexity of the network, reduces the model dependence, and improves the classification accuracy by improving the attention and loss function. Firstly, a significant reduction in computational complexity is achieved by replacing the traditional 1×1 convolution with a grouped convolution. A cross-channel feature weighting module is introduced to overcome the limitation of grouped convolution in terms of information exchange between channels. In addition, to effectively deal with the data sample class imbalance problem, this research introduces an attention mechanism that leverages channel and feature space equalization to effectively capture global information. Additionally, it formulates a loss function that incorporates both intra- and inter-class metrics, aiming to enhance model accuracy. Finally, through validation on multiple datasets, the methods in this study show their efficiency and superiority.

Keywords: small sample image classification, depth separable convolution, attention, cross-channel feature weighting

1 Introduction

Few-shot learning [1, 2], characterized by its ability to rapidly adapt to new tasks with limited data, leverages prior knowledge to identify novel patterns from a small number of samples. This approach enables models to learn effectively from extremely scarce datasets, achieving accurate predictions. Its significance is particularly pronounced in academic research domains where data scarcity poses a significant challenge. It constitutes an n-way k-shot learning paradigm, where each task consists of N categories and each category has K labeled samples, and a small number of labeled samples are used to train the model to quickly adapt to unknown new tasks. Currently, a diverse array of few-shot image classification techniques has been introduced, encompassing model-based, optimization-based, measure-based, and data augmentation-based methodologies. This research offers an overview of model-based and measure-based approaches. The model-based few-shot image classification method is to use the model for classification. For example, the convolutional neural network is combined with few-shot learning, and a few-shot classification model based on the convolutional neural network is proposed. Metric-based few-shot image classification methods use a small number of samples to learn a metric space in which similar sample pairs are close together and dissimilar sample pairs are far apart. For a new task, the test data label is determined by querying the distance between the test sample and a small number of labeled samples in the new task. Typical measures are Siamese neural networks [3], matching networks [4], prototype networks [5] and relational networks [6].

Although many methods have been proposed to solve few-shot classification problems, the field of few-shot learning still faces several challenges. Owing to the scarcity of training samples, model-based few-shot classification approaches may succumb to overfitting due to their inherent complexity, thereby adversely affecting the model's predictive accuracy. Meanwhile, in the context of small-sample learning, there are significant differences between different classes of samples, which may have a significant impact on the learning effectiveness of the model. Numerous researchers have proposed a series of solutions to this problem.

* Corresponding Author

To overcome the overfitting problem caused by complex model structures as well as to reduce the prediction accuracy, researchers have proposed a series of lightweight models. In 2017, the MobileNet [7, 8] family of models was first proposed, which deconstructs the traditional convolution operation into two independent steps: deep convolution and point-by-point convolution. In the deep convolution stage, one convolution kernel is applied to each channel of the input separately, and the convolution is performed only in the spatial dimension, without involving the mixing of information across channels. Immediately after that, the point-by-point convolution stage uses a 1×1 convolution kernel for information fusion across channels while keeping the spatial dimension constant. This decomposition strategy significantly reduces the number of parameters and computational complexity of the model. In the same year, researchers further proposed the Xception [9], which employs an improved Inception [10] module, in which the traditional Inception module is replaced by depth-separable convolution. This improvement assumes that the cross-channel convolution operation on the spatial feature mapping and the spatial convolution operation within each channel are separable. In 2018, researchers proposed the ShuffleNet [11, 12] family of models, which significantly reduces the computational complexity and the number of model parameters by employing grouped convolution and channel shuffling techniques. However, the channel blending operation lacks interpretability and performs low on certain performance metrics.

However, although the lightweight method can reduce the complexity of the network and solve the over-fitting phenomenon, it also easily leads to the reduction of the classification accuracy of the network. Attention mechanisms have shown great advantages in improving the accuracy of neural networks. Attention mechanisms are mainly divided into channel-based attention and space-based attention. Channle-based attention focuses on the relationship between feature channels and emphasizes important features by feature channel weighting. For example, in 2017 Jie Hu, Li Shen, and Gang Sun et al. proposed SE (Squeeze-and-Excitation) attention [13], which improves the performance of the network by explicitly modeling the interdependence between feature channels and adaptively reclaiming the feature responses of the channels. In 2019, Qilong Wang et al. proposed ECANet (Efficient Channel Attention) [14], and proposed a new Channel Attention module Efficient Channel Attention (ECA). This module effectively implements the attention mechanism through the local cross-channel interaction strategy and one-dimensional convolution, while significantly reducing the model complexity. In addition, they developed a method for adaptively selecting the size of 1D convolution kernels to determine the coverage of local cross-channel interactions. Space-based attention focuses on the spatial location relationship in the feature map, and it highlights important regions by weighting different locations in the feature map. For example, Kaiming He et al. proposed CBAM (Convolutional Block Attention Module) [15] in 2018, which combines channel attention and spatial attention. By calculating the attention weights in the channel dimension and the spatial dimension, the attention weights of these two dimensions are then multiplied to finally obtain the enhanced feature map. This approach has been shown to significantly improve model performance on tasks such as image classification and object detection.

Similarly, to mitigate the effect of excessive differences between the training and test samples on the model learning effectiveness. In 2017, Tsung-Yi Lin and colleagues proposed Focal Loss [16], a loss function whose main purpose is to solve the category imbalance problem that exists in the field of target detection. The function effectively counteracts the tendency of models to optimize a larger number of categories while ignoring a smaller number of categories in the case of extreme category imbalance by redesigning the cross-entropy loss. However, Focal Loss introduces two key hyperparameters, which means that finding optimal parameter configurations on different tasks and datasets may require extensive experimentation and fine-tuning. The Dice Loss function has been widely adopted in medical image processing research in recent years. Dice Loss focuses directly on solving the category imbalance problem and optimizes the model performance by maximizing the overlap region between the predicted segmentation and the true segmentation. Given that medical image segmentation tasks often involve highly imbalanced datasets, Dice Loss is particularly favored in this area. However, the Dice Loss function also suffers from several limitations, especially in some cases when the overlap between predicted and true labels is extremely limited or completely absent, which may lead to the problem of vanishing gradients.

Overall, most current network models for small-sample image classification have improved in terms of structural lightweight and the ability to adapt to different sample data. However, the following challenges remain: 1. The current model structure needs to be further optimized to achieve a balance between lightweight and accuracy. 2. Many convolutional models are highly task-specific and lack adaptability to new classes, which may lead to reduced prediction accuracy. To solve the above problems, this paper proposes a lightweight network based on depth-separable convolution.

The rest of this paper is divided into the following chapters, and Section III focuses on related work. Section IV mainly introduces and summarizes the proposed method in detail. In Section 5, the proposed method is ap-

plied to different data sets for comparative experiments to verify the effectiveness and superiority of the proposed method. Finally, the basic conclusions of this paper are summarized.

2 Related Works

In the field of few-shot classification, the proposed lightweight network based on depth-wise separable convolutions contributes to two main ways. First, in the few-shot domain, 1×1 convolution has a large amount of calculation in the model. This model uses group convolution to replace 1×1 convolution to reduce network complexity and adds a feature fusion module and attention mechanism to improve the classification ability of the network without increasing parameters. Second, the amount of data in the few-shot domain is too small, which leads to the dependence of the network model on specific data, resulting in the decline of the classification ability of the network model when the new category data is input. Therefore, a loss function based on intra-class and inter-class measurement is proposed to consider the relationship between classes, effectively deal with the class imbalance problem, and reduce the dependence of the network model on specific tasks. The specific related research content is as follows.

2.1 Deep Separable Convolution

Traditional convolutional operations generally consist of applying a convolutional kernel to each channel of the input feature map independently for spatial convolution, followed by aggregating the convolution outcomes across all channels to produce the output feature map. Deeply separable convolution reduces the computational complexity by decomposing this process into two separate stages: channel-by-channel convolution and pointwise convolution. Depth separable convolution mainly consists of these two modules, namely channel-by-channel convolution and point-by-point convolution. Channel-by-channel convolution involves applying a convolution kernel to each channel of the input feature map independently. Following channel-by-channel convolution, point-by-point convolution combines the convolution results of each channel by applying a 1x1 convolution kernel. However, 3x3 channel-by-channel convolution accounts for only a small fraction of the overall computation, while more than 90% of the computation is consumed by 1x1 point-by-point convolution, and thus depth-separable convolution still involves a significant amount of computation overall.

2.2 SE Attention

The SE (Squeeze-Excitation) attention mechanism [13] aims to enhance the performance of convolutional neural networks by explicitly modeling the dependencies between channels. The mechanism first performs a compression (Squeeze) operation to pool the global average of the output feature maps of the convolutional layers to form a channel descriptor. This descriptor generates global information about the channel by aggregating information from all spatial locations on each channel. Subsequently, an Excitation operation is performed, and the SE attention mechanism learns the importance of each channel through a fully connected network. Fully connected networks usually consist of two layers: a compression layer and an expansion layer. The compression layer aims to reduce the dimensionality of the descriptors, while the expansion layer reverts to the original number of channels. Between these two layers, the ReLU activation function is usually used to introduce nonlinear properties. The output of the FC network is a vector of weights for the channels, which is subsequently used to adjust each channel of the original feature map for adaptive weight assignment between channels. The SE channel attention mechanism mainly works by augmenting the input features with channel features while keeping the size of the input feature map constant. However, it is found that SE attention only performs pooling operations on the channel's internal information and does not fully consider the contextual information. This leads to a weakening of the feature extraction ability of SE attention on datasets where the training samples are significantly different from the test samples and consequently affects the prediction accuracy.

2.3 Loss Function

A suitable loss function accelerates the convergence of the model, assists the model in quickly finding the pa-

rameters that minimize the loss, and enhances the model's ability to generalize over unseen data, thus preventing overfitting and improving prediction accuracy. Therefore, researchers have studied the loss function in depth. Mean Absolute Error (MAE) [17] assesses the accuracy of prediction by calculating the average of the absolute difference between the predicted and actual values. However, this loss function has discontinuous derivatives at zero, which makes gradient-based optimization algorithms challenging to train. Afterward, the cross-entropy loss function [18] compensates for this shortcoming, mainly by minimizing the difference between the predicted and true probability distributions. This loss function is continuously derivable in most cases so that gradient-based optimization algorithms are easier to train. However, when facing the problem of imbalanced category distribution, the cross-entropy loss function is biased, resulting in poorer identification of a few categories and making the model less accurate. Subsequently, a series of adaptive loss functions [19] have been proposed, which can adaptively adjust the weights of the loss function according to the inhomogeneity of the sample distribution, and thus cope with the data imbalance problem more effectively. However, such loss functions may increase the complexity and computational cost of the model, especially when dealing with large-scale datasets, and this additional computational overhead may become a significant problem. This suggests that the reduced convergence speed of the loss function may lead to a decrease in model accuracy when faced with significant differences between the training and test samples.

3 Method

3.1 Group Convolution

Deeply separable convolution is a classical network architecture designed to be lightweight. However, it suffers from a significant limitation: dense 1x1 pointwise convolution operations. During the computation of deep separable convolution, the 3x3 group convolution accounts for only a small portion of the computation, while more than 90% of the computation is consumed by 1x1 pointwise convolution operations. Because of this, in this study, group convolution is used instead of 1x1 point convolution to reduce computational complexity and achieve a lightweight network. Grouped convolution divides the input channels into multiple groups when processing multi-channel input data and performs the convolution operation independently for each group. The input data is $C_{in} = H \times W$, and the number of groups is g. If the size of the convolution kernel of each group is $n \times 1 \times 1$, the output feature size obtained is $C_{out} = W \times H$, where $C_{out} = n \times g$.

The calculation amount of 1×1 point convolution and packet convolution module are shown in the following formula respectively. The calculation amount of the DWGC network module is significantly reduced, and the lightweight optimization of the model is realized:

$$1 \times 1 \times h \times w \times c = h \times w \times c \ . \tag{1}$$

$$(1 \times 1 \times h \times w \times c)/ \ g = (1/g) \times h \times w \times c \ . \tag{2}$$

h, w denotes the height and width of the input features, $c = c_{in} \times c_{out}$, c_{in} denotes the number of input channels, and c_{out} denotes the number of output channels. g is the number of groups of grouped convolutions, and according to experimental comparisons.

3.2 Cross-Channel Feature Weighting

Replacing the standard 1x1 pointwise convolution with grouped convolution in MobileNetV1 significantly reduced the computational complexity of the model. However, it was found that the deficiency of grouped convolution in the exchange of information between channels may lead to a decrease in the prediction accuracy of the model. Because of this, this study proposes a feature fusion module (CCF) based on cross-channel feature weighting, which aims to achieve effective information exchange between channels without adding additional parameter burden. As shown in Fig. 1, although the original high-dimensional HW space can fully characterize all the information of the channel, it is not suitable for direct representation because it imposes a huge computational burden. In this paper, each channel is converted to a one-dimensional vector, by which the high-dimensional matrix can be characterized by a low-dimensional vector without losing too much information. By sampling

the average pooled values of rows and columns from the image as representation vectors, the main information within each channel is preserved.

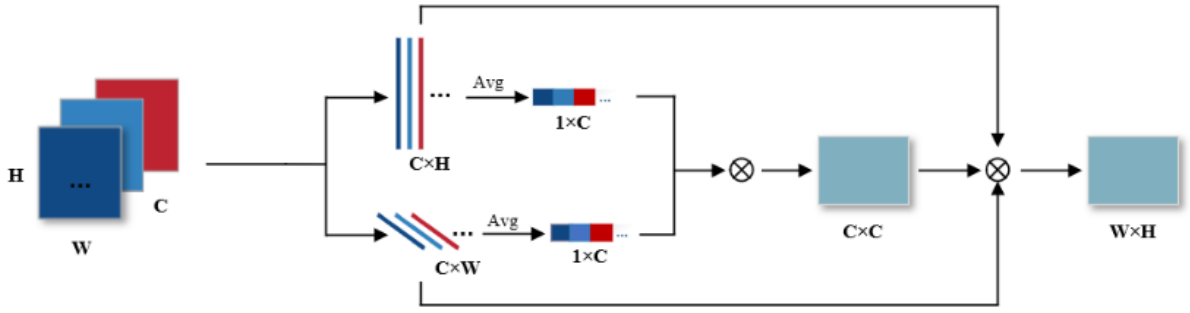

Fig. 1. Cross-Channels feature

$$X_w = \frac{1}{H}\sum_{i=1}^{H} X_i \quad . \tag{3}$$

$$X_h = \frac{1}{W}\sum_{i=1}^{W} X_i \quad . \tag{4}$$

X_i denotes the input data. X_w and X_h denote the row vector and column vector, respectively. After obtaining two representative vectors for each channel, a compressive transform was performed, which feeds the vectors in parallel to two compression transform operations, compressing them from 1×H and 1×W dimensions to 1×1, and generating the channel attention weights by computing the weights point by point averaging, corresponding to the horizontal vectors X_W^C and the vertical vector X_H^C, respectively.

$$X_h^c X_w^c = \frac{1}{W}\sum_{i=1}^{W} X_{i,\mathrm{w}} \quad . \tag{5}$$

$$X_h^c = \frac{1}{H}\sum_{j=1}^{H} X_{j,h} \quad . \tag{6}$$

$$W_c^c = X_w^c \times X_h^c \quad . \tag{7}$$

$$W_{w,h} = W_{c,c} \times X_w^c \times X_h^c \quad . \tag{8}$$

Subsequently, the obtained attentional weights of the two channels are multiplied to obtain the inter-channel correlation matrix $W_{C,C}$, which reflects the relationship between the channels. Then, this matrix is multiplied by the horizontal vector X_w^c, and the vertical vector X_h^c, so that each pixel of each image can be affected by the correlation matrix. Finally, the output channel weights $X_{w,h}$ are used as the weights of cross-channel features.

3.3 Contextual Pooled Attention

In the field of small sample learning, lightweight networks often lead to weak feature extraction capability due to their simple structure. To overcome this limitation, this paper introduces the SE (Squeeze-Excitation) attention mechanism to enhance the feature extraction ability of the model. However, SE attention only pools the informa-

tion inside the channel and does not fully consider the contextual information, which leads to a decrease in the feature extraction ability of SE attention on datasets where the training samples are significantly different from the test samples. Because of this, this paper improves the SE attention mechanism by introducing the context pooling operation, so that the attention mechanism can still extract features effectively when facing datasets with large differences, without adding additional parameter burden. The amount of computation for CPSE and SE is shown in the following formula respectively:

$$C \times C / R + C \times N \times C / R \Rightarrow (N+1 \quad C^2 / R \ . \tag{9}$$

$$2 \times N \times N \times C \times C / R = 2N^2C^2 / R \ . \tag{10}$$

Overall, unlike the generation of adaptive vectors for each instance in SE attention, CPSE attention, which is primarily for each task, allows for a better understanding of the image, enabling the model to exhibit higher accuracy and robustness in tasks such as classification. As can be seen from the formula, the computation of CPSE attention is also significantly reduced compared to SE attention. The specific structure of CPSE attention is shown in Fig. 2.

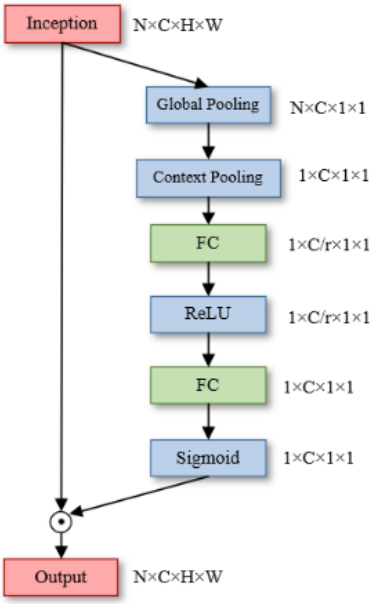

Fig. 2. CPSE attention

3.4 A Lightweight Network Module Based on Deeply Separable Convolutions

In this study, a lightweight network module based on depth-separable convolution is proposed. As shown in Fig. 3, this module achieves the lightweight of the network by replacing 1x1 pointwise convolution with groupwise convolution to reduce the computational complexity. However, a limitation of grouped convolutional layers is that features between different groups cannot communicate effectively, which may reduce the feature extraction capability originally provided by pointwise convolution. Therefore, this study introduces a cross-channel feature weighting (CCF) based feature fusion mechanism to facilitate the exchange of information between channels.

In addition, the residual structure can output the input features directly through the shortcut path, thus ensuring the stability of the gradient. Based on this, this paper adopts the residual [20] structure to realize the fusion of features and feature weighting. By introducing residual scaling with factor α, we can obtain features. α is set mainly based on two reasons: First, the direct use of residual learning (e.g., $\alpha = 1$) may lead to numerical insta-

bility during training. Second, the nature of residual connectivity allows CCF to be seamlessly inserted into any pre-trained network without significantly affecting its initial behavior. (e.g., $\alpha \rightarrow 0$) By applying CCF, subsequent convolutional layers perceive the entire space, even if their receptive fields are limited in size. CCF allows the network to focus on richer visual features and enables the exchange of information across channels. Ultimately, the outputs of the grouped convolutional paths are summed with the feature maps that have undergone cross-channel feature fusion to form the residual structure. In addition, for small sample data, the model is prone to pay excessive attention to detailed features, which leads to a decrease in generalization ability. Therefore, to better utilize the global information, this paper introduces the CPSE channel attention mechanism after the residual structure, which helps the model to understand the image content more comprehensively, and thus show higher accuracy and robustness in the classification task. The overall computational flow of this lightweight module is as follows:

$$Output = DW \cdot CPSE = dw(O) \cdot CPSE = dw(\alpha \times G \times F + G) \cdot CPSE \ . \tag{11}$$

Output is the final output, DW is the feature after depth separable convolution is completed, CPSE is the attention added to context pooling, $dw(\cdot)$ is the depth separable convolution function, G denotes the feature after grouped convolution, F denotes the feature weighting after cross-channel feature fusion is performed, denotes the output feature after cross-channel feature weighting is completed, and α is the residual scaling factor.

The common application of the ReLU activation function brings the so-called “dead ReLU” problem. To solve this problem, this paper adopts the ACONC [21] activation function, which can be automatically adjusted according to the needs of data and tasks, thus effectively alleviating the “dead ReLU” phenomenon and reducing the complexity to a certain extent. In addition, the channel spatial form of the β-gating factor helps to capture the inter-channel feature variability more effectively. The formula is as follows:

$$\beta = \sigma W_1 W_2 \sum_{h=1}^{H} \sum_{w=1}^{W} x_{c,h,w} \ . \tag{12}$$

$$ACONC[x] = max(p_1 x, p_2 x) = (p_1 - p_2)x\sigma[\beta(p_1 - p_2)x] + p_2 x \ . \tag{13}$$

σ denotes the Sigmoid function, p_1 and p_2 use two learnable parameters for self-tuning. when $x \rightarrow -\infty$, the gradient is p_1, and when $x \rightarrow +\infty$, the gradient is p_2. $W_1 \in C \times C / r$ and $W_2 \in C / r \times C$. The switching factor β is used to determine whether or not to activate the neuron ($\beta = 0$ indicates inactivity), and the input sample $x \in R^{C \times H \times W}$, which can be switched between nonlinearity and linearity by adjusting the β gating factor.

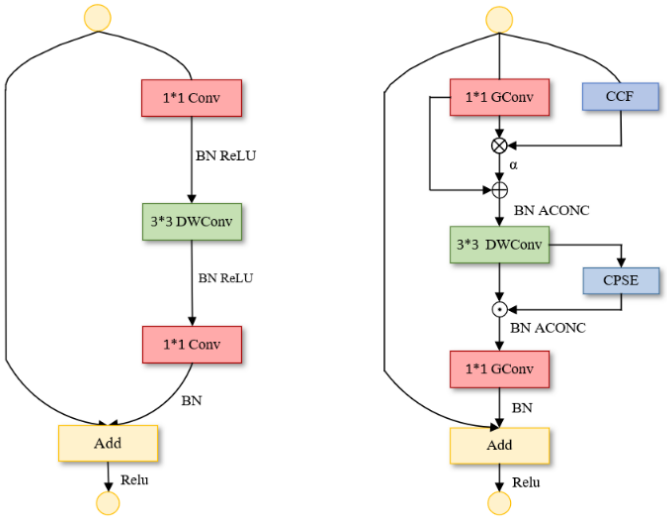

Fig. 3. Left: MobileNetV1. Right: DWGC

The DWGC-LWNet model is divided into two core steps: fusion and selection. First, the model fuses the spatial global features under different channels; second, it obtains the weights of the features and global features under different channels. Then, the initial global features are multiplied with the corresponding weights to realize the enhancement of global features. The proposed DWGC-LWNet network module significantly reduces the number of parameters of the deep model, while the prediction accuracy of the model is improved for datasets with large class differences.

The computational amount of the lightweight networks MobileNetV1, ShuffleNetV1, and the module after replacing 1x1 pointwise convolution with groupwise convolution in MobileNetV1 in this paper are shown in the following equations. The computational volume of the DWGC network module is significantly reduced, which achieves the optimization of model lightness.

$$1\times1\times h\times w\times c+3\times3\times h\times w\times c+1\times1\times1\times h\times w\times c=11\times h\times w\times c \ . \tag{14}$$

$$(1\times1\times h\times w\times c)/g+3\times3\times h\times w\times c+(1\times1\times h\times w\times c)/g=(9+2/g)\times h\times w\times c \ . \tag{15}$$

$$(1\times1\times h\times w\times c\times\beta)/g+3\times3\times h\times w\times c+(1\times1\times h\times w\times c)/g=(9+(1+\beta)/g)\times h\times w\times c \ . \tag{16}$$

h, w denote the height and width of the input feature $c = c_{in} \times c_{out}$, c_{in} denotes the number of input channels, and c_{out} denotes the number of output channels. Here $\beta \leq 1$, β is the switching factor of the ACONC activation function, which is determined by the input features. g is the number of groups of grouped convolutions, and according to experimental comparisons, the best result is achieved when g = 4 is chosen.

3.5 Intra- and Inter-class Based Loss Functions

Currently in the field of small samples, in the face of new classes that differ greatly from the training samples, the loss function cannot fully utilize the hierarchical information of different classes of images, resulting in lower model convergence speed and accuracy. Therefore, this paper proposes loss functions based on intra- and inter-class metrics to improve the generalization ability of the model. Among them, the intra-class loss can reduce the distance between different samples within the same class, which helps to make the model better capture the internal features and variations of each class. Interclass loss increases the distance between different classes, thus increasing the variability of data points between classes. The loss function for samples versus non-affiliated classes helps the model learn to better separate samples from different classes, forces the model to assign samples to their correct classes, and ensures that there is a clear distinction between samples and samples from non-affiliated classes. There are obvious differences between the categories, the fused loss function can improve the differentiation between the categories and make the training process of the model more stable.

Intra- class Losses. Specifically, this study employs an Euclidean distance metric to assess intra-class spacing. In implementing this metric, this study utilized the support samples in each class of the prototype network to solve for a representative sample p_i , i.e.:

$$P_i=\frac{1}{n}\sum_{k=1}^{n}f(x_k) \ . \tag{17}$$

where X_k is the image samples in the support set, and the support set features of each class are directly regarded as the prototype of the class p_i . Then, using the Euclidean distance [22] to measure the distance between each support sample and the representative samples, the sum of these distances is used as the loss function within each class in the training data, i.e., the intraclass loss:

$$L_{self}=\sum_{i=1}^{m}\sum_{k=1}^{n}d(f(x_k),p_i) \ . \tag{18}$$

Inter-category Losses. In this study, a prototype network is used to extract representative features for each category to construct a prototype set. Subsequently, this set of prototypes is further processed to obtain the full representative prototypes. Finally, the distance between each prototype in the prototype set and the full representative prototype is measured using the Euclidean distance metric, which serves as the basis for the calculation of inter-class loss. The details are as follows:

$$\overline{p} = \frac{1}{m}\sum_{i=1}^{m} p_i \quad . \tag{19}$$

The Euclidean distance is used to measure the difference between each prototype and the full set of representative prototypes as the basis for the calculation of inter-class loss. Here i.e. $d(\cdot)$ denotes the Euclidean distance, i.e:

$$L_{class} = \sum_{i=1}^{m} d(\overline{p}, p_i) \quad . \tag{20}$$

Sample And Non-affiliated Losses. The relationship measure between samples and non-affiliated categories considers the similarity between samples and their affiliated categories and the differentiation between samples and other categories. Specifically, this study used the Euclidean distance method to calculate the distance between a single sample and a prototype of a non-affiliated category. i.e:

$$L_{else} = \sum_{i=1}^{m-1}\sum_{k=1}^{n} d(f(x_k), p_i)(x_k \notin i) \quad . \tag{21}$$

Losses Based on Intra- and Inter-class Metrics. Finally, the cross-entropy loss function is used to find the loss function L_{query} , which is combined with the loss functions of Eqs. (18), (20), (21) to obtain the optimal loss function:

$$L = L_{query} + \lambda \frac{L_{self}}{L_{class} + L_{else}} \quad . \tag{22}$$

The hyperparameter $1 \geq \lambda \geq 0$ is used to adjust the weights of the two auxiliary loss functions, which focus on capturing intra-class relationships when $\lambda \to 0$, and gradually favor the capture of inter-class relationships when $\lambda \to 0$. By comprehensively considering the relationships between samples and their belonging and non-belonging classes, this study significantly improves the generalization ability of the classification model, which is especially suitable for those complex classification tasks with subtle differences between classes or unbalanced data distribution.

3.6 Lightweight Networks Based on Deeply Separable Convolutions

The infrastructure of the DWGC-LWNet lightweight network consists mainly of the DWGC module, which is constructed step by step through four stages: The first stage uses a 3x3 convolutional kernel for convolutional operations to extract the base features and a sliding strategy with step size 2. The second stage consists of four DWGC modules, where the first module does not use grouped convolution and has a step size of 2, and the remaining three modules use grouped convolution operations with a step size of 1. The third stage consists of eight DWGC modules and the fourth stage consists of four DWGC modules, where the first modules in both stages employ a step-size 2 strategy and the remaining modules employ a step-size 1. Finally, the DWGC-LWNet network is constituted by global pooling and fully connected operations. This network architecture not only realizes the lightweight of the model but also ensures the prediction accuracy of the model, thus improving the generalization ability of the model. The parameter configuration of the specific model is shown in Table 1, and the detailed structure of the model is shown in Fig. 4.

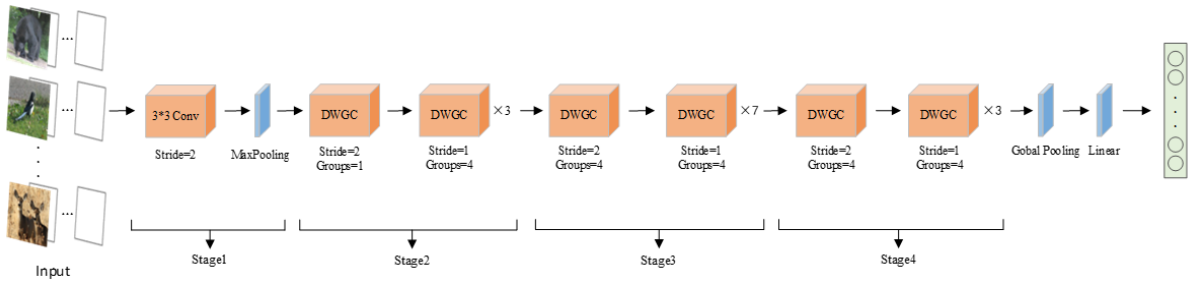

Fig. 4. DWGC-LWNet

4 Experiment

In this section, the implementation details, performance evaluation metrics, and experimental results of the proposed model on the dataset are presented. The model parameters are presented in Table 1. To further evaluate the effectiveness of the proposed model, the results are compared with those of the proposed method within the last five years, the comparison experiments are presented in Table 2. In addition, an ablation study is conducted to analyze the effects of various influencing factors. The results of the ablation study are shown in Table 3.

4.1 Implementation Details

For the hardware environment of the experiments, this study used an NVIDIA Volta Tesla V100 graphics card with 64G of RAM, and all experiments were implemented using the Pytorch framework. The experiments followed most of the training settings and hyperparameters, utilizing the SGD optimization method as the gradient descent algorithm. During training, the hyperparameters were determined through several trials, including the number of groups G for grouped convolution was set to 4, the value of λ was set to 0.001, the value of α was set to 0.01, and the SGD optimizer was used with an initial learning rate of 0.001 and the batch size was set to 100. The study used different evaluation metrics to evaluate the proposed model and the other existing models, including the accuracy rate and the number of parameters (shown in Fig. 5 and Fig. 6).

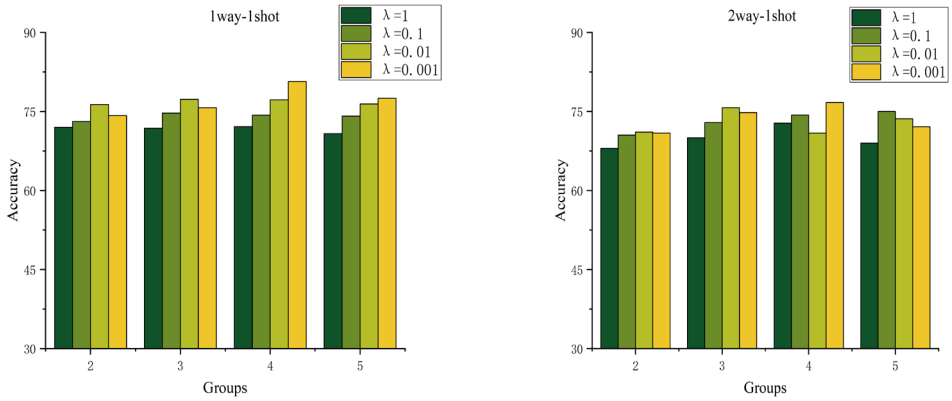

Fig. 5. Hyperparameter tuning: comparison of accuracy at different λ in 1way-1shot and 2way-1shot

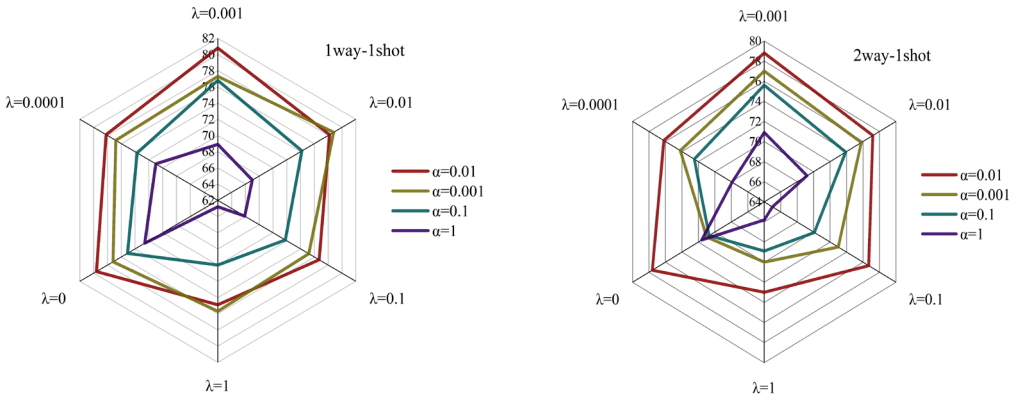

Fig. 6. Hyperparameter tuning: comparison of accuracy at different λ and α in 1way-1shot and 2way-1shot

For the image classification task, the model proposed in this paper is validated on a public dataset, specifically the tiered-ImageNet dataset. The network was experimented on the dataset for the 1way-1shot task. This dataset contains 10,000 training images of 100 categories and 2000 validation images of 100 categories. The key point is that both the training and validation data are preprocessed to verify the generalization ability of the model. For this purpose, validation data, which differed significantly from the training data categories, was chosen for the experiment. Each image was randomly cropped to 224x224 size and randomly flipped horizontally. In the evaluation phase, the images were first resized to 256 short edges, and then the center region of 224x224 size was cropped from them.

Table 1. Model parameters

Stage	Operation	Step	Input	Output	Repeat	Kernel	Stride	Group
1	3×3 Conv		3×224×224	24×112×112	1	3×3	2	1
	MaxPooling		24×112×112	24×56×56	1	3×3	2	1
2	DWGC				1		2	1
		1×1 GConv	24×56×56	8×56×56		1×1		
		DWConv	8×56×56	8×28×28		3×3,1×1		
		1×1 GConv	8×28×28	272×28×28		1×1		
3	DWGC				1		2	4
		1×1 GConv	272×28×28	68×28×28		1×1		
		DWConv	68×28×28	68×14×14		3×3,1×1		
		1×1 GConv	68×14×14	544×14×14		1×1		
	DWGC				7		1	4
		1×1 GConv	544×14×14	136×14×14		1×1		
		DWConv	136×14×14	136×14×14		3×3,1×1		
		1×1 GConv	136×14×14	544×14×14		1×1		
4	DWGC				1		2	4
		1×1 GConv	544×14×14	136×14×14		1×1		
		DWConv	136×14×14	136×7×7		3×3,1×1		
		1×1 GConv	136×7×7	1088×7×7		1×1		
	DWGC				3		1	4
		1×1 GConv	1088×7×7	272×7×7		1×1		
		DWConv	272×7×7	272×7×7		3×3,1×1		
		1×1 GConv	272×7×7	1088×7×7		1×1		
	GlobalPooling		1088×7×7	1088×1×1	1	7×7	1	
	Linear		1088×1×1	Numclass	1			

4.2 Images Classification

In this study, the comparison experiments between the proposed model and seven other prediction models in terms of small sample image classification performance are analyzed in detail. The experimental results are summarized in Table 2 and Fig. 7. The models compared include MobileNetV1 [7], MobileNetV2 [8], ShuffleNetV1 [11], ShuffleNetV2 [12], the late MobileVit-S [23], EdgeNext [24], and AlexNet [25] models. Compared with the MobileNet, and ShuffleNet family of models, the GCF model proposed in this paper achieves higher accuracy. Compared with MobileVit-S, and EdgeNext models, the GCF model performs better in terms of lightweight. It is worth noting that all existing methods are implemented in similar experimental environments and with the same dataset to ensure fair comparisons.

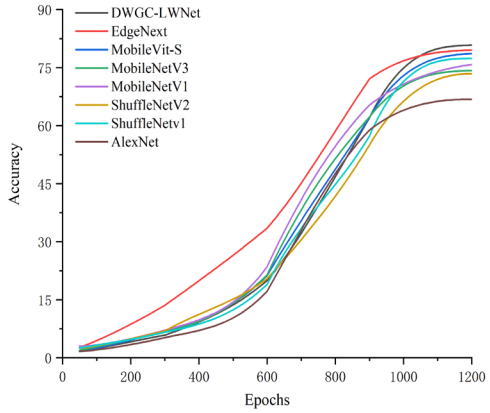

Fig. 7. Model comparison experiment

Through comparative experiments, it can be concluded that using group convolution instead of point convolution in the proposed network model can reduce the model's complexity, and the model's number of parameters is reduced by 0.58M, which also shows obvious advantages compared with other networks. At the same time, the classification accuracy of the network model is improved by 2.31%, which also confirms that CPSE attention and CCF help to improve the feature extraction ability of the network. Finally, aiming at the problem of the decline of classification accuracy caused by the large gap between samples of small sample data, the improved loss function solves this problem and improves the classification accuracy by 1.1% compared with the original network.

Table 2. Model comparison

Model	Accuracy (%)	Parameter (M)
AlexNet	66.87	14.74
ShuffleNetV1	77.37	1.83
ShuffleNetV2	73.43	1.35
MobileNetV1	75.78	3.29
MobileNetV3	74.23	1.76
MobileVit-S	78.62	5.54
EdgeNext	79.58	6.15
Proposed Model	79.68	1.35
Proposed Model + Loss	80.78	1.35

4.3 Ablation Experiment

In this section, we perform an ablation analysis of the DWGC-LWNet lightweight model proposed in this paper

to evaluate the importance of its different components. Table 3 demonstrates the role of the four methods CCF, CPSE, ACONC, and LOSS in the architecture proposed in this paper.

Fig. 8. CPSE heat map

When the CPSE module is removed, the model accuracy decreases by 2.55%, which indicates that the CPSE module is critical to the overall performance of the model. To visualize the effect of the CPSE module, this paper also generates a heat map with the addition of this module. (Shown in Fig. 8) In addition, the contribution of the CCF module is eliminated, although this results in a slight decrease in accuracy. Overall, the model presented in this paper provides a good trade-off between complexity and accuracy.

Table 3. Ablation experiments

Case	CCF	CPSE	ACONC	LOSS	Accuracy (%)	Parameter (M)
1	√				75.01	1.35
2		√			76.33	1.35
3			√		80.52	1.83
4	√	√			78.04	1.35
5	√		√		77.13	1.35
6		√	√		78.94	1.35
7	√	√	√		79.68	1.35
8	√	√	√	√	80.78	1.35

In this paper, we validate Accuracy and Parameter with a stepwise improvement approach. it can be seen through experiments that, compared to the base depth-separable convolutional model MobileNetV1, The DWGC-LWNet lightweight module reduces the number of model parameters from 3.29M to 1.35M, and improves the accuracy by 3.9%. Adding the loss function improves the accuracy by 5%. The specific experimental results are displayed in Table 3.

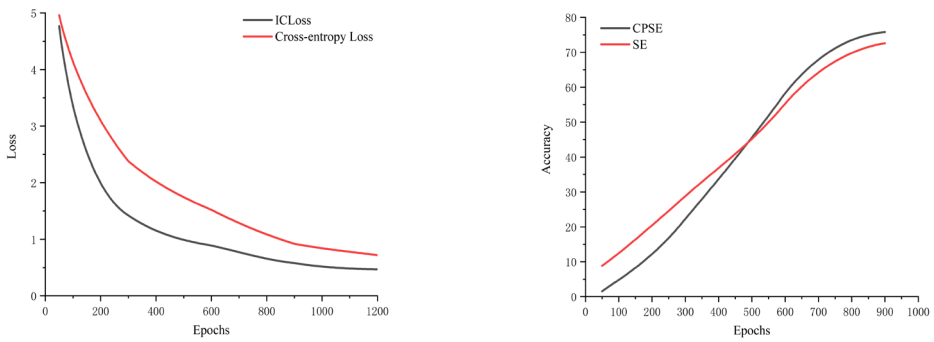

Fig. 9. Comparison of different loss and different attention

In this paper, comparative experiments are conducted for different loss functions and attention, as shown in Fig. 9. ICLoss and Cross-entropy loss functions are compared. The feature extraction ability of the model under the ICLoss loss function is better, the loss is smaller, and the final loss is reduced by 0.41 compared with the cross-entropy loss function. At the same time, compared with SE attention, CPSE attention improves the feature extraction ability of the network, and the classification accuracy of the network is improved by 2.3%.

5 Conclusions

In this study, this paper proposes a lightweight network architecture based on deeply separable convolutions, aiming to achieve a lightweight network while maintaining model accuracy. This goal is achieved by replacing the traditional 1x1 pointwise convolution with grouped convolution, combined with a cross-channel feature weighting module. In addition, this study introduces the Context-aware Spatial Equalization (CPSE) attention mechanism to enhance the extraction of contextual features. To solve the category imbalance problem, this study proposes a loss function based on intra- and inter-class metrics, which can exploit the relationship between categories, thus improving the generalization ability and classification performance of the model. In the current investigation, the hyperparameters of the model were determined through experimental means; however, the full spectrum of hyperparameter possibilities was not exhaustively investigated. Moreover, despite advancements in the model's accuracy, within the domain of few-shot learning, achieving classification accuracy comparable to conventional classification tasks remains a challenge.

Acknowledgments

This work is supported by the National Natural Science Foundation of China 62063021 (Research on HMS Scheduling Optimization and Control and Intelligence System in Manufacturing IoT Environment), National Natural Science Foundation of China 62162040 (Research on Terrain Representation and Dissemination Adaptive Scheduling Strategy for Large-scale Social Network Influence Adaptability) and National Natural Science Foundation of China 52061022 (Research on the regulation mechanism of micro-alloy precipitated phase and texture in low carbon high conductivity high strength steel).

References

[1] W.Y. Chen, Y.C. Liu, Z. Kira, Y.C.F. Wang, J.B. Huang, A closer look at few-shot classification, in: Proc. 2019 International Conference on Machine Learning, 2019.
[2] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural neworks, Communications of the ACM 60(6)(2017) 84-90.
[3] G. Koch, R. Zemel, R. Salakhutdinov, Siamese neural networks for one-shot image recognition, ICML deep learning workshop 2(1)(2015) 1-30.
[4] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra, Matching networks for one-shot learning, in: Proc. 2016 Neural Information Processing Systems, 2016.
[5] J. Snell, K. Swersky, R. Zemel, Prototypical networks for few-shot learning, in: Proc. 2017 31st International Conference on Neural Information Processing Systems, 2017.
[6] A. Santoro, D. Raposo, D.G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, T. Lillicrap, A simple neural network module for relational reasoning, in: Proc. 2017 Proceedings of the 30th International conference on neural information processing systems, 2017.
[7] A.G. Howard, Mobilenets: Efficient convolutional neural networks for mobile vision applications. <https://arxiv.org/pdf/1704.04861>, 2017 (accessed 17.10.23).
[8] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.C. Chen, Mobilenetv2: Inverted residuals and linear bottlenecks, in: Proc. 2018 Proceedings of the IEEE conference on computer vision and pattern recognition, 2018.
[9] F. Chollet, Xception: Deep learning with depthwise separable convolutions, in: Proc. 2017 Proceedings of the IEEE conference on computer vision and pattern recognition, 2017.
[10] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: Proc. 2015 Proceedings of the IEEE conference on computer vision and pattern recognition, 2015.

[11] X. Zhang, X. Zhou, M. Lin, J. Sun, Shufflenet: An extremely efficient convolutional neural network for mobile devices, in: Proc. 2018 Proceedings of the IEEE conference on computer vision and pattern recognition, 2018.
[12] N. Ma, X. Zhang, H.T. Zheng, J. Sun, Shufflenet v2: Practical guidelines for efficient CNN architecture design, in: Proc. 2018 Proceedings of the European conference on computer vision (ECCV), 2018.
[13] X. Jin, Y. Xie, X.S. Wei, B.R. Zhao, Z.M. Chen, X. Tan, Delving deep into spatial pooling for squeeze-and-excitation networks, Pattern Recognition 121(2022) 108159.
[14] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, Q. Hu, ECA-Net: Efficient channel attention for deep convolutional neural networks, in: Proc. 2020 Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020.
[15] S. Woo, J. Park, J.Y. Lee, I.S. Kweon, Cbam: Convolutional block attention module, in: Proc. 2018 Proceedings of the European conference on computer vision (ECCV), 2018.
[16] T.Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: Proc. 2017 Proceedings of the IEEE international conference on computer vision, 2017.
[17] T.O. Hodson, Root mean square error (RMSE) or mean absolute error (MAE): When to use them or not, in: Proc. 2022 Geoscientific Model Development Discussions, 2022.
[18] Z. Zhang, M. Sabuncu, Generalized cross-entropy loss for training deep neural networks with noisy labels, in: Proc. 2018 Neural Information Processing Systems (NeurIPS), 2018.
[19] B. Li, Y. Liu, X. Wang, Gradient harmonized single-stage detector, Proceedings of the AAAI conference on artificial intelligence 33(1)(2019) 8577-8584.
[20] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proc. 2016 Proceedings of the IEEE conference on computer vision and pattern recognition, 2016.
[21] N. Ma, X. Zhang, M. Liu, J. Sun, Activate or not: Learning customized activation, in: Proc. 2021 Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021.
[22] L. Wang, Y. Zhang, J. Feng, On the Euclidean distance of images, IEEE transactions on pattern analysis and machine intelligence 27(8)(2005) 1334-1339.
[23] S. Mehta, M. Rastegari, Mobilevit: light-weight, general-purpose, and mobile-friendly vision transformer, in: Proc. 2022 International Conference on Learning Representations (ICLR), 2022.
[24] M. Maaz, A. Shaker, H. Cholakkal, S. Khan, S.W. Zamir, R.M. Anwer, F.S. Khan, Edgenext: efficiently amalgamated cnn-transformer architecture for mobile vision applications, in: Proc. 2022 European conference on computer vision, 2022.
[25] M.Z. Alom, T.M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M.S. Nasrin, V.K. Asari, The history began from Alexnet: A comprehensive survey on deep learning approaches. <https://arxiv.org/pdf/1803.01164>, 2018 (accessed 04.01.24).