Journal of Big Data

RESEARCH Open Access

# CCNNet: a novel lightweight convolutional neural network and its application in traditional Chinese medicine recognition

Hu Gang[1,2*], Sheng Guanglei[1,3*], Wang Xiaofeng[1,2] and Jiang Jinlin

*Correspondence:
hugang@xaut.edu.cn;
shenggl@163.com

[1] School of Computer Science and Engineering, Xi'an University of Technology, No. 5 South Jinhua Road, Xi'an, Shaanxi 710048, People's Republic of China
[2] Department of Applied Mathematics, Xi'an University of Technology, Xi'an, Shaanxi 710054, People's Republic of China
[3] Department of Electronics and Information Engineering, Bozhou University, Bozhou, Anhui 236800, People's Republic of China

## Abstract

With the development of computer vision technology, the demand for deploying vision inspection tasks on edge mobile devices is becoming increasingly widespread. To meet the requirements of application scenarios on edge devices with limited computational resources, many lightweight models have been proposed that achieves good performance with fewer parameters. In order to achieve higher model accuracy with fewer parameters, a novel lightweight convolutional neural network CCNNet is proposed. The proposed model compresses the modern CNN architecture with "bottleneck" architecture and gets multi-scale features with downsampling rate 3, adopts GCIR module stacking and MDCA attention mechanism to promote the model performance. Compares with several benchmark lightweight convolutional neural network models on CIFAR-10, CIFAR-100 and ImageNet-1 K, the proposed model outperforms them. In order to verify its generalization, a fine-grained dataset for traditional Chinese medicine recognition named "TCM-100" is created. The proposed model applies in the field of traditional Chinese medicine recognition and achieves good classification accuracy, which also demonstrates it generalizes well. The bottleneck framework of the proposed model has some reference values for the design of lightweight model. The proposed model has some promotion significance for classification or recognition applications in other fields.

**Keywords:** Lightweight model, Convolution neural network, Channel attention mechanism, Traditional Chinese medicine

## Introduction

Recently, with the increasing development of deep convolutional neural networks, many deep convolutional neural network models have been proposed which perform very SOTA in vision tasks, such as VGGNet [1], GoogleNet [2], ResNet [3], ConvNeXt [4], Vision Transformer [5], and Swin- Transformer [6] perform well in classification, semantic segmentation, and object detection task. The application fields of computer vision are becoming more and more widespread, but it is not advanced enough in domains of traditional Chinese medicine(TCM). Since the outbreak of the neo-coronavirus epidemic, the traditional Chinese medicine has played an increasingly important role in the fight against the epidemic, and there is evidence that TCM

combinations can be effective against the neo-coronavirus. In order to promote informationization in the field of TCM and provide technical support for the intellectualization development of TCM construction, the development of technologies and methods for TCM recognition and detection is one of the emerging applications of computer vision. Currently, there are some research works on deep learning models in the field of Chinese herbal medicine recognition, but are just only some attempts for single-object classification tasks. Some datasets are created in the literature [7–9], but small size. And the effective deep learning models are not constructed, the existing models(ResNet, VGG, MobileNets, ShuffleNet, etc.) are used to accomplish the attempt of Chinese medicine classification task. Due to the limitations of the dataset and models, the accuracy of Chinese medicine classification in the preliminary works is not good performance. In addition, the application scenario of single-target recognition in traditional Chinese medicine is the variety identification process, which often happens in the transactions of Chinese medicine market, retailing in Chinese medicine stores, and public use of medicine. It is more suitable to use portable mobile devices. Therefore, further research is needed to promote the development of TCM recognition.

The present approaches for model lightweight are mainly model compression algorithms [10], neural network architecture search (NAS) [11] and artificial design. Among them, compression algorithms for convolutional neural networks [10] include rule-based neural network model compression (weight pruning, weight quantization, low-rank decomposition, knowledge distillation, and others), and automatic machine learning-based automatic neural network model compression (AMC algorithm).

The representative one is the knowledge distillation model. Although the knowledge distillation model is weightlighted, the number of parameters is still large. Neural architecture search [11] (NAS) refers to the automatic design of a high-performance neural network architecture to solve a specific task in a specific search space (global, local, etc.) based on a search strategy (reinforcement learning, evolutionary algorithms, etc.). The representative models are MobileNetV3 [12], EfficientNet [13], NasNet [14], etc. Artificial designed lightweight neural network model compression techniques rely on heuristic and rule-based strategies, and the representative models are MobileNetV1 [15], MobileNetV2 [16], SqueezeNet [17], GhostNet [18], ShuffleNet [19], MobileNext [20], etc. The performances of them are relatively invariant when the number of parameters is significantly reduced. However, they do not yet meet the classification needs of all practical applications. Transformer-based lightweight models are also developing rapidly, and their main work is the lightweight of multi-headed self-attention module, and the representative of the Transformer-based model is MobileVit [21]. However, Transformer is computation complex and resource intensive, so it is not suitable for deploying applications on edge mobile devices.

On the other hand, due to the relative lag in the development of TCM intelligence technology, the public datasets for TCM have not yet been established. The preliminary literature [7–9] have made some attempts to classify and identify TCM and built some datasets, but the datasets are not publicly available. No new lightweight models have been built for TCM classification. Therefore, we decided to create a traditional Chinese medicine classification dataset and build a novel lightweight model.

Our aims is to design a model with fewer number of parameters and higher accuracy, and applied in the field of the traditional Chinese medicine classification task. The major contributions of this paper are as follows.

(1) A novel lightweight model CCNNet is proposed, and the accuracy of the model outperforms the existing lightweight models mentioned in this paper.
(2) An effective lightweighting module component GCIR block and attention mechanism MDCA are designed.
(3) According to the catalog of the Pharmacopoeia of the People's Republic of China on Chinese medicine and drinking tablets, a fine-grained dataset(TCM-100) is created.
(4) The proposed model is applied to the field of traditional Chinese medicine recognition and achieves an accuracy of 92.5% on TCM-100.

## Related work

### Lightweight convolutional neural network model

In recent years, with the development of deep learning, many lightweight deep convolutional neural network classification models have emerged within the field of computer vision. It mainly includes two types of lightweight deep convolutional neural network models based on artificial design and neural architecture search, among which the deep learning models based on artificial design is MobileNet[15], MobileNetV2 [16], SqueezeNet [17], GhostNet [18], ShuffleNet [19], MobileNeXt [20], MobileVit [21], DenseNet [22], NASNet [23], ShuffleNetV2 [24], SENet [25], PP-LCNet [26], etc. They design the lightweight convolutional operations or modular structures, including small convolutional kernels instead of large convolutional kernels[17], feature reuse[18, 22], grouped convolutional and channel shuffle[19, 24], depthwise separable convolution [15], bottleneck structure and inverted residual structure[16], and other modules or components to reduce the model parameters. And use attention mechanism to improve the model performance [25]. The models based on neural architecture search are [12, 13, 27, 28], etc. They can only be searched on a known search space, i.e., the module structure or basic operation units are known. Neural architecture search can obtain lightweight models with balanced model width, depth, resolution, and other factors. But it cannot obtain models on unknown search spaces, i.e., neural architecture search cannot obtain new efficient basic operations or units.

Regardless of any lightweight models, the design aims to reduce the number of parameters and speed up computations while balancing the model accuracy. The above-mentioned papers have done a lot of work in reducing the number of model parameters and improving the classification accuracy of the model by designing convolutional operations or module structures, but further research work is needed to obtain a lightweight classification model with higher performance. In addition, all of the above lightweight models are designed for convolutional operations or module structures, but there are not many researches on the overall framework of lightweight models, and further research is needed to optimize the overall framework of model to reduce the number of parameters and improve model speed and performance.

In order to significantly reduce the number of model parameters and inference time while improving the accuracy of the model, the overall model architecture of modern CNNs is compressed, and a new lightweight module GCIR and MDCA attention is designed to improve the model accuracy.

#### Attention mechanism

Recently, many attention mechanisms have been proposed, including channel attention SE [25], efficient channel attention (ECA) [29], convolutional block attention module (CBAM) [30], Shuffle Attention [31], Triplet Attention (TA) [32], Coordinate Attention(CA) [33], SimAM [34] non-referential attention mechanism, self attention(SA) [35]. Among them, SE [25], ECA [29] are channel-based attention which captures information about channel features. SA [35]is a spatial-based attention which gets global information relations. CBAM [30], Shuffle Attention [31] are attention based on combination of channel and spatial, which extract local and global important features. Triplet Attention [32] and Coordinate Attention (CA) [33] are a 3D(channel, width, height) cross-complementary attention which can selects important feature information. In addition, we also see a graph attention mechanism GAM [36], which can reduce spatial information loss and enhance feature representation.

Although all attention mechanisms can improve model performance, different models and scenarios use different attention mechanisms to achieve different effects. Therefore, further research work on attention mechanism is needed to adapt different models.

In summary, a large amount of literature has been carried out on model lightweight methods and attention mechanisms. We will design a new lightweight block, attention mechanism and construct a new model.

## Approach

#### Motivation

In order to applied in traditional Chinese medicine classification and detection, we apply the design rules and methods of lightweight convolutional neural networks to design a lightweight neural network model. Although the MobileNets [12, 15, 16], GhostNet [18], ShuffleNet [19], SqueezeNe [17] reduce the number of parameters, but also the classification accuracy is reduced. These methods all significantly reduce the number of parameters on the modular components but do not optimize the overall model framework. To further reduce the number of model parameters, the ConvNeXt [4] model is compressed from 4 to 3 stages by using the downsampling rate 3, but the performance of model do not descend significantly. Subsequently, we observed the downsampling of some classical models, including the head downsampling method, downsampling rate, number of channels, etc., as shown in Table 1.

The performance of the models is generally good with a downsampling rate of 1, 2, or 4 as shown in Table 1. To demonstrate that the performance of the model is still excellent using downsampling rate 3, we take some experiments on ConvNeXt [4] to validate the performance with downsampling rate 3. We set the downsampling rate of the ConvNeXt [4] model to 3, which causes a series of changes in the model, the feature map size of the middle layer of the model shrinks at an accelerated rate, and its four stages are compressed into three stages, which results in a significant reduction in the number

**Table 1** Comparison of downsampling layer in classical model

| Model | Stemlayer Kernelsize | Downsample rate(stride) | Output Feature map size | Stages | Channels |
|---|---|---|---|---|---|
| AlexNet | conv11 × 11 | 4 | 56 × 56 | 4 | 96 |
| GoogleNet | conv7 × 7 | 2 | 112 × 112 | 4 | 64 |
| ResNet | conv7 × 7 | 4 | 56 × 56 | 4 | 64 |
| VGGNet | conv3 × 3 | 1 | 224 × 224 | 4 | 64 |
| ConvNeXt | conv4 × 4 | 4 | 56 × 56 | 4 | 96 |
| ShuffleNetV2 | conv3 × 3 | 4 | 56 × 56 | 3 | 24 |
| GhostNet | conv3 × 3 | 2 | 112 × 112 | 4 | 16 |
| EfficientNet | conv3 × 3 | 2 | 112 × 112 | 4 | 32 |

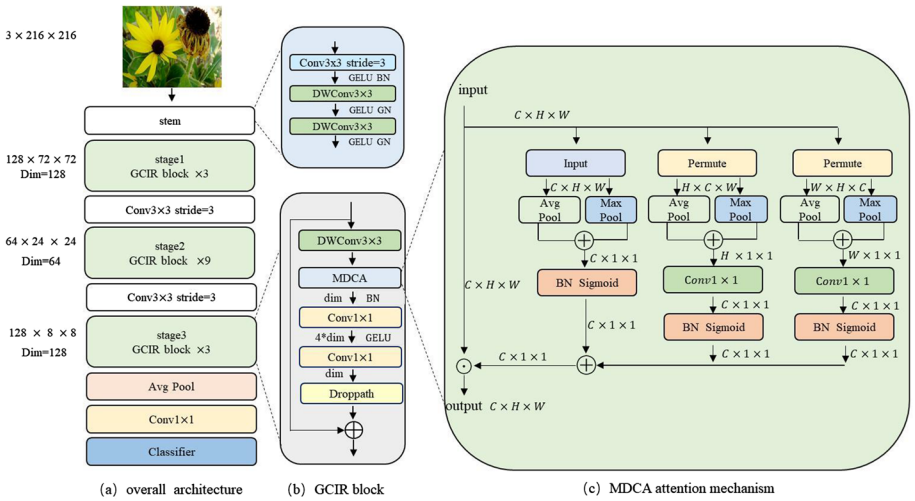



**Fig. 1** The CCNNet Architecture

of parameters of the model. The compressed model architecture is similar to the framework of ShuffleNetV1, ShuffleNetV2 [19, 24], i.e., the middle layers of the model are 3 stages. We train it on the public datasets cifar-10 and cifar-100, and the results show that the accuracy of the compressed model is not significantly reduced. To compensate for the degradation of the performance of the compressed model and improve the accuracy of the compressed model, we fully consider the principles and strategies of the lightweight network to design the lightweight module GCIR, MDCA attention mechanism and construct the CCNNet (Compressed Convolution Neural Network) model. The SRFBN mechanism [37] is referenced. The structure of the model is shown in Fig. 1.

Firstly, CCNNet feeds the input image ($3 \times 216 \times 216$) into the stem layer, performs a downsampling operation using a $3 \times 3$ convolution with stride 3 and two $3 \times 3$ depthwise separable convolutions, the output is $128 \times 72 \times 72$. Depthwise separable convolution is used to extract rich fine-grained features, which are then put into GCIR block of stage1 for transform stacking to make the features richer. The GCIR module uses residual structure, depthwise separable convolutions, and inverted bottleneck structure to extract local feature information, which greatly reduces the number of parameters. To enhance the capability of channel feature perceiving and global relationship capturing, the GCIR module embeds the MDCA (multidimensional channel attention mechanism).

According to the idea of GhostNet [18], it is known that the features captured in stage 1 are rich but redundant, so we compress the features in downsample layer to reduce the redundancy, i.e., the number of channels becomes half. In stage2, the input is stacked 9 times by GCIR module, and the output is $64 \times 24 \times 24$. Subsequently, the output of stage2 is downsampled to extend the dimension using the downsample layer, and an $128 \times 8 \times 8$ output is obtained. In stage3, we get high-level semantic information by 3 times GCIR module stacking operation. Subsequently, the output is performed by the classification head module. The logits for classification is obtained after a global average pooling and $1 \times 1$ convolution operation. The classification task is completed. Overall, it seems that the variation of channels in CCNNet is a bottleneck architecture, which achieves the purpose of reducing the number of parameters.

#### Overall architecture

We designed a lightweight convolutional neural network model CCNNet, whose overall architecture is shown in Fig. 1(a). It consists of a stem layer, 3 stages in intermediate layers, and an output classification layer. From the perspective of model width, the number of channels in the model varies from $128 \rightarrow 64 \rightarrow 128$, and it is a bottleneck architecture. This architecture can greatly reduce the number of parameters.

We used a threefold downsampling rate with feature maps of sizes from $216 \rightarrow 72 \rightarrow 24 \rightarrow 8$, respectively, making the middle layers of the model into 3 stages. Given the SOTA model ConvNeXt [4], we adopted its stacking setup method, i.e., setting the ratio of the number of stacks per stage to 3:9:3. Such a design inspired by revisiting some design laws of ResNet [3], SwinTransform [6], ConvNeXt [4], ShuffleNet [19], and ShuffleNetV2 [24]. In ResNet, the number of blocks stacked in the middle stage(stage2-3) is big, and the number of blocks stacked from stage1 to stage4 is (3, 4, 6, 3), i.e., the ratio of stage1: middle layer (stage2, stage3): stage4 is 3:10:3, which is about 1:3:1. The Swin-T of Swin-Transformer [6] model is 1:1:3:1, which is approximately 1:3:1 ignoring stage1. And the number of stacking in ConvNeXt [4] is (3, 3, 9, 3), and analogously its ratio is also approximately 1:3:1. The lightweight models ShuffleNet[19] and ShuffleNetV2[24] directly adopt three stages, and the stacking times ratio of each stage is 3:7:3, approximately 1:3:1. We find that it is feasible to compress the hidden layer of the model to three stages. The stacking ratio of the three stages is 1:3:1. The three stages model can greatly reduce the number of parameters of the model.

Using modern CNN model design concepts [3–6], our model uses three stages for generating stacked transformations of feature maps at different scales, which are important for dense prediction tasks [38]. To generate multi-scale features, a downsampling(DS) layer consisting of a $3 \times 3$ convolution with a stride of 3 and a batch normalization (BN) [39] is applied before each stage to reduce the size of the middle layer feature map and project it to the dimension of next stage. In each stage, several GCIR (Group convolution inverted Residual) modules are sequentially stacked to perform feature transformation while keeping the input feature map resolution constant. As an example, the third stage of CCNNet contains three GCIR modules, as shown in Fig. 1(b), and we embed MDCA attention into the GCIR module to capture more feature representatives. We will describe the GCIR module and MDCA attention in detail in Sect. "GCIR Block" and Sect. "MDCA (Multi Dimension Channel Attention)", respectively. The final output of

the model is a classification head consisting of a global average pooling layer, a projection layer, and a softmax classification layer.

Given an input image, we can obtain three different resolutions of hierarchical feature maps, and the span of the above feature maps are 3, 9, and 27. The proposed model can get a multi-scale representation of the input image, which can be well applied to downstream tasks such as object detection and semantic segmentation. The specific structure of the network is shown in Table 2.

### Stem layer

GoogleNet[2] and ResNet[3] use $7 \times 7$ convolution with $2 \times$ downsampling rate and $3 \times 3$ maximum pooling with stride 2 to get the result of fourfold downsampling, which will lead to overlapping sampling regions, and the redundancy of perceived features will increase. While in ConvNeXt [4] model, the $4 \times 4$ convolution with stride 4 and non-overlapping downsampling operation are used. This may result in less detailed features captured and loss of fine-grained information. The $3 \times 3$ convolution operation proposed in the VGGNet model to extract fine-grained depth feature information works best. To capture rich and redundant fine-grained features, the stem module is designed. First, the $3 \times 3$ standard convolution operation with stride 3 is performed on the input features and gets the output of $72 \times 72$ feature map with 128 channels. The reason for the larger model width (number of channels 128) is that more channels can capture much richer features [13, 18]. Then followed by two $3 \times 3$ depthwise separable convolution with stride 1 which can increase the perception field while better extracting local fine-grained feature information [38]. Each of the $3 \times 3$ depthwise separable convolutions is followed a group normalization and GELU activation [40] operation. This operation can improve

**Table 2** The specific structure of the CCNNet

| Outsize | Layer | CCNNet1.0X | CCNNet1.5X | CCNNet2.0X | CCNNet3.0X |
|---|---|---|---|---|---|
| $72 \times 72$ | Stem | $3 \times 3, 128, stride = 3$ | $3 \times 3, 160, stride = 3$ | $3 \times 3, 192, stride = 3$ | $3 \times 3, 256, stride = 3$ |
| | | $[3 \times 3, 128] \times 2$ | $[3 \times 3, 160] \times 2$ | $[3 \times 3, 192] \times 2$ | $[3 \times 3, 256] \times 2$ |
| Stage1 | GCIR | $\begin{bmatrix} 3 \times 3, 128 \\ MDCA \\ 1 \times 1, 512 \\ 1 \times 1, 128 \end{bmatrix} \times 3$ | $\begin{bmatrix} 3 \times 3, 160 \\ MDCA \\ 1 \times 1, 640 \\ 1 \times 1, 160 \end{bmatrix} \times 3$ | $\begin{bmatrix} 3 \times 3, 192 \\ MDCA \\ 1 \times 1, 768 \\ 1 \times 1, 192 \end{bmatrix} \times 3$ | $\begin{bmatrix} 3 \times 3, 256 \\ MDCA \\ 1 \times 1, 1024 \\ 1 \times 1, 256 \end{bmatrix} \times 3$ |
| $24 \times 24$ | DS | $3 \times 3, 64, stride = 3$ | $3 \times 3, 80, stride = 3$ | $3 \times 3, 96, stride = 3$ | $3 \times 3, 128, stride = 3$ |
| Stage2 | GCIR | $\begin{bmatrix} 3 \times 3, 64 \\ MDCA \\ 1 \times 1, 256 \\ 1 \times 1, 64 \end{bmatrix} \times 9$ | $\begin{bmatrix} 3 \times 3, 80 \\ MDCA \\ 1 \times 1, 320 \\ 1 \times 1, 80 \end{bmatrix} \times 9$ | $\begin{bmatrix} 3 \times 3, 96 \\ MDCA \\ 1 \times 1, 384 \\ 1 \times 1, 96 \end{bmatrix} \times 9$ | $\begin{bmatrix} 3 \times 3, 128 \\ MDCA \\ 1 \times 1, 512 \\ 1 \times 1, 128 \end{bmatrix} \times 9$ |
| $8 \times 8$ | DS | $3 \times 3, 128, stride = 3$ | $3 \times 3, 160, stride = 3$ | $3 \times 3, 192, stride = 3$ | $3 \times 3, 256, stride = 3$ |
| Stage3 | GCIR | $\begin{bmatrix} 3 \times 3, 128 \\ MDCA \\ 1 \times 1, 512 \\ 1 \times 1, 128 \end{bmatrix} \times 3$ | $\begin{bmatrix} 3 \times 3, 160 \\ MDCA \\ 1 \times 1, 640 \\ 1 \times 1, 160 \end{bmatrix} \times 3$ | $\begin{bmatrix} 3 \times 3, 192 \\ MDCA \\ 1 \times 1, 768 \\ 1 \times 1, 192 \end{bmatrix} \times 3$ | $\begin{bmatrix} 3 \times 3, 256 \\ MDCA \\ 1 \times 1, 1024 \\ 1 \times 1, 256 \end{bmatrix} \times 3$ |
| $1 \times 1$ | GAP | $AvgPool(1 \times 1)$ | | | |
| $1 \times 1$ | FC | $1 \times 1, 1000$ | | | |
| Parameter | | 1.4 M | 2.2 M | 3.1 M | 5.4 M |
| Flops | | 0.36B | 0.46B | 0.55B | 0.74B |

the accuracy of the model. The experiments show that our stem can improve the model performance by 0.2% counter to the stem operation of ConvNeXt [4] which not using GELU activation.

#### GCIR block

The proposed GCIR module consists of a deformed inverted residual block, a MDCA attention block, and a Droppath, as shown in Fig. 1(b) and specifically described below. It consists of a $3 \times 3$ depthwise separable convolution with stride 1, batch normalization, a $1 \times 1$ convolution to ascend fourfold dimension, GELU [40] activation, and a $1 \times 1$ convolution to descend fourfold dimension, and a DropPath [41]. The $3 \times 3$ depthwise separable convolution can extract fine-grained feature information while reducing the number of parameters. The role of the MDCA module is to learn the features from three dimensions, i.e., channel, height, and width, respectively, and fuse them to generate the MDCA attention. It will be described in Sect. "MDCA (Multi Dimension Channel Attention)". The $1 \times 1$ convolution raises the channel dimension to four times the original. Followed by a GELU activation operation at the higher dimension, and then reduces the channel dimension to the original by $1 \times 1$ convolution. The linear inverted bottleneck structure [12, 16] is very important because the GELU activation operation causes a more loss of low-dimensional feature information, but the linear expansion module prevents the non-linear activation from destroying too much information [12]. The role of the subsequent DropPath [41] operation is to prevent overfitting, and DropPath is a regularization tool that can randomly "delete" the multi-branch structure from the deep learning model. When the drop rate is 0, the multi-branch structure in the model is an identity projected. If the drop rate is a probability value greater than zero, a random deactivation operation of the branch structure is performed with the drop rate probability. Finally, there is a residual operation. The above operations can be formulated as below:

$$Output = X + Drop\,(Conv_{4d\to d}(Act(Conv_{d\to 4d}(BN(MDCA(Conv(X))))))) \tag{1}$$

where, $X$ is the input feature map, $X \in R^{H\times W\times d}$, $H \times W$ is the input feature map resolution of the current stage, and $d$ is the number of channels of the feature. $DWConv()$ is a depthwise separable convolution,$MDCA()$ indicates the multidimensional channel attention mechanism, $Conv_{d\to 4d}$ signify that using the $1 \times 1$ convolution to ascend the dimension from d to 4d, $Conv_{4d\to d}$ express that descend the dimension from 4d to d by using the $1 \times 1$ convolution, $BN()$ means the batch normalization, $Act()$ denotes the GELU activation function, $Drop()$ operation will randomly deactivate multi-branch structure of the model according to the probability.

The GCIR module is inspired by the SandGlass module, which is shown in Fig. 2a, and the GCIR module is shown in Fig. 2b. The performance of the SandGlass module has been demonstrated in the MobileNeXt[20] model. We embed the MDCA attention mechanism into the SandGlass module and drop the final depthwise separable convolution. To promote the performance of the model, we add a BN normalization operation before the inverted bottleneck structure and activation at the higher dimension. It will reduce the loss of feature information. Finally, we give a Droppath operation to prevent

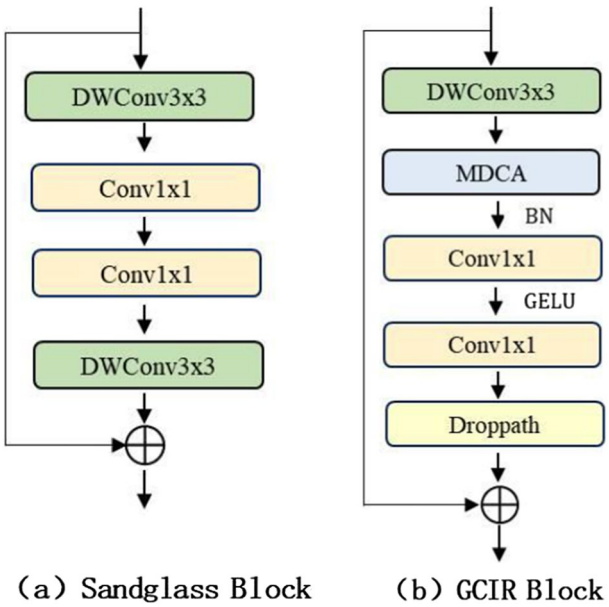



**Fig. 2** Comparison between Sandglass and GCIR

overfitting. The accuracy of the model rose by 0.25% on CIFAR-10 dataset and 0.15% on CIFAR-100 dataset by using the GCIR module.

#### Downsample layer

The downsample layer consists of a $3 \times 3$ convolution with stride 3 and a batch normalization. In ResNet [3], ConvNeXt [4], and SwinTransform [6] models, spatial downsampling is implemented by separating the layers separately, using a normalization layer and a $2 \times 2$ convolution with stride 2. Separate downsampling layers are added between the stages separately. We adopted a similar strategy of spatial downsampling using a normalization layer and a $3 \times 3$ convolution layer with stride 3. Subsequently, experiments show that adding normalization layers in which the spatial resolution changed help the training to be stable.

#### MDCA (Multi dimension channel attention)

Depthwise separable convolution is used in the GCIR module, so the weights of each channel are very important in the feature representation. To enhance the ability of the CCNNet model to perceive local features and global dependencies, we investigated SE [25], CBAM [30], ECA [29], SIMAM [34], CA(Coordinate Attention) [33], SPM [42], TA (Triplet Attention) [32], SA (Shuffle Attention) [31], etc. Considering low consumption, the proposed model using the SE channel attention after depthwise separable convolution leads to good performance in training. The SE channel attention obtains the relationship between channels to reweight the importance of each channel, but ignores the possible influence of spatial height dimension features and spatial width dimension features on channel attention. Our motivation is to lightweight and optimize the SE attention by considering the influence of the height dimensional features and the width dimensional features on channels. Firstly, our method performs a 2D (spatial width and channel) global pooling and global max

pooling from the view of height dimension to extract the features that represent the relationship between height and channel with feature shape $\mathrm{H} \times 1 \times 1$, and then a 2D (spatial height and channel) global pooling and global max pooling from the view of width dimension to extract the features that represent the relationship between height and channel with feature shape $\mathrm{W} \times 1 \times 1$. These two features are then projected to the channel dimension and then summed with the SE channel dimension attention, thus injecting the influence factor of spatial features into the SE channel attention and also bringing in some global information for optimizing the SE channel attention. This helps the channel attention to represent the weights of each channel more accurately.

Drawing on Coordinate Attention [33] and Triplet Attention [32] attention mechanism methods, the multidimensional channel attention(MDCA) is proposed. As shown in Fig. 1c, the specific method is as follows: input feature maps $X \in R^{H \times W \times C}$, C, H, W denote channel, height, and width, respectively. 2D global average pooling $GAP()$ and global maximum pooling $GMP()$ are performed along three different dimensions, such as channel dimension, spatial height dimension, and spatial width dimension, respectively. Then the results of global average pooling and global maximum pooling are summed up, and this operation is referred to as $Z - pool_i$. That is $Z - pool_i(X) = GAP(X) + GMP(X)$, i indicates that the operation is performed in the corresponding dimension, and i takes the values C, H, W indicating the channel, height, and width, respectively. $BN()$ denotes the normalization operation, $sigmoid()$ denotes the nonlinear activation operation, and $Conv_{1\times1}()$ denotes the $1 \times 1$ convolution to ascend dimension operation.

The channel dimension attention AC can be described as,

$$AC = sigmoid(BN(Z - pool_C(X))) \tag{2}$$

The height dimension attention AH can be shown as,

$$AH = sigmoid(BN(Conv_{1\times1}(Z - pool_H(X)))) \tag{3}$$

The width dimension attention AW can be expressed as,

$$AW = sigmoid(BN(Conv_{1\times1}(Z - pool_W(X)))) \tag{4}$$

Finally the feature information of channel dimension $AC$, height dimension $AH$ and width dimension $AW$ are fused to generate MDCA attention. As shown in the Fig. 1(c), it can be expressed formally as

$$Attention_{MDCA} = f(AC, AH, AW) \tag{5}$$

where $f()$ indicates a summation operation.

The results of the ablation experiments show that the MDCA multidimensional channel attention module is better than SE attention module at capturing focal information and focusing on a wider area. The MDCA multidimensional channel attention module instead of the SE attention module in the CCNNet model can rise the accuracy of the model by about 0.5% on Imagenet dataset.

## Experimental

In this section, we investigate the effectiveness of the CCNNet model by experimenting with classification tasks on image classification datasets CIFAR-10 [43], CIFAR-100 [43] and ImageNet1K [44]. We compare the performance of the proposed CCNNet model with existing lightweight models on classification task. The CCNNet model is compared and validated on the Chinese medicine recognition dataset to demonstrate that its application is effective and feasible in the field of traditional Chinese medicine recognition.

### Experimental environment

The experimental environment: Win10, 64-bit OS with pytorch1.10 environment, Intel(R) Core(TM) i9-10900 K CPU, NVIDIA GeForce RTX 3090 GPU, 32G RAM. The datasets are CIFAR-10, CIFAR-100 [43] and ImageNet1K [44]. We split the training sets and validation sets on CIFAR-10, CIFAR-100 [43], and ImageNet1K [44] in the same environment with a ratio of 8:2, and use the same training method for MobileNetv3small [12], ShuffleNetV2 [24], MobileNext [20], respectively. Each model has been trained and their accuracies on the test set are compared under the same conditions.

### Datasets

In this paper, all experimental comparisons are performed on four datasets, including three publicly available datasets and one self-built TCM recognition dataset. CIFAR-10, CIFAR-100 [43] and ImageNet1K [44] were chosen as the public datasets. And ImageNet1K is an image classification benchmark, i.e., the ILSVRC2012 dataset. Currently, the improved traditional method [45] based on SVM achieves 75.64% classification accuracy on ImageNet1K dataset, while the deep learning method Swin-Transformer [6] achieves 87.3% classification accuracy. It demonstrators that deep learning methods have better classification accuracy.

The TCM dataset (TCM-100) is a dataset with fine-grained features, in which 100 categories of TCM images are collected. In the 2020 edition of the Pharmacopoeia of the People's Republic of China, there are 2711 categories of Chinese herbal medicines and more than 1000 categories of common Chinese herbal medicines and tablets. According to the outline catalog of the classification of Chinese herbal medicines and indexes in the Pharmacopoeia of the People's Republic of China, we create a fine-grained features dataset with 100 categories for Chinese herbal medicines classification.

According to the description of the Pharmacopoeia of the People's Republic of China on the standard of Chinese medicine tablets, the raw materials of Chinese medicine tablets are generally from the roots and stems, bark, flowers, leaves, and fruits of plants [7–9, 46]. The leaves and flowers of plants are dried and shaped into finished Chinese medicine tablets, the roots and stems of Chinese medicine tablets are mostly made in the form of slices, which are classified into thin slices, thick slices, slanted slices, straight slices, filaments, blocks, etc. Although they are both slices, the shapes of slices are round slices, cylindrical thick slices, round-like slices, oval slices, irregular slices, etc. The fruits of Chinese medicine tablets are also mostly in the form of seeds and granules, which are smaller in volume. The large fruits are generally round slices or irregular slices, such as hawthorn slices and hedgehog slices, see the Pharmacopoeia of the People's Republic of China for details. In addition to the above-mentioned shapes and irregularities, the

color and texture of the Chinese medicine tablets are also very similar. Small discrepancy inter-class and large discrepancy intra-class is an important characteristic of Chinese medicine images, which makes the identification more difficult.

We organized the staff to collect 100 species of traditional Chinese medicine images, accumulating more than 60,000 images. It includes approximately 40,000 images that were taken and labeled by ourselves and about 20,000 images that were publicly available on the Internet. The differences between some of the herbal medicine are so slight that it is often difficult for the human eye to identify their species. Some of the images in the traditional Chinese medicine dataset are shown in Fig. 3.

## Experimental results and analysis

### *The validation of CCNNet model*

For a fair comparison with recent work, we used the same training approach and data augmentation strategy as in the ConvNeXt [4] model and trained the models for 200 epochs using the AdamW [47] optimizer. The models have been trained with a single NVIDIA 3090 GPU on CIFAR-10, CIFAR-100, and ImageNet for classification experiments. Table 3 and Fig. 4 shows the performance of the CCNNet model on the classification datasets. Compared with other convolution-based or transformer-based models, our model achieves relatively better accuracy with fewer parameters and flops. In particular, with a similar number of parameters, the proposed CCNNet achieves a top-1 accuracy of 70.1% on the ImageNet dataset with only 1.4 M number of parameters, and its top-1 Acc is 2.7% higher than the baseline model MobileNetv3-small [12], 2.4% higher than MobileNeXt0.5X [20], and 1.0% higher than MobileVit-XXS [21]. This indicates that the GCIR module and the MDCA attention mechanism


**Fig. 3** Display of the traditional Chinese medicine recognition Dataset

**Table 3** Comparison to some baseline models on ImageNet-1 k, CIFAR-10 and CIFAR-100

| Model | Parameters/M | Flops/B | Top-1 acc | | |
|---|---|---|---|---|---|
| | | | ImageNet (%) | cifar-10 (%) | cifar-100 (%) |
| MobileNetV3small | 2.5 | 0.3 | 67.4 | 89.6 | 74.2 |
| Shuffle NetV2 0.5X | 1.4 | 0.15 | 61.1 | 88.9 | 71.4 |
| GhostNet 0.5X | 2.6 | 0.14 | 66.2 | 89.8 | 74.1 |
| MobileNeXt 0.5X | 1.8 | 0.3 | 67.7 | 89.9 | 75.2 |
| MobileVit-XXS | 1.3 | 0.7 | 69.0 | 90.2 | 76.3 |
| CCNNet(Ours) | 1.4 | 0.36 | 70.1 | 90.8 | 77.6 |

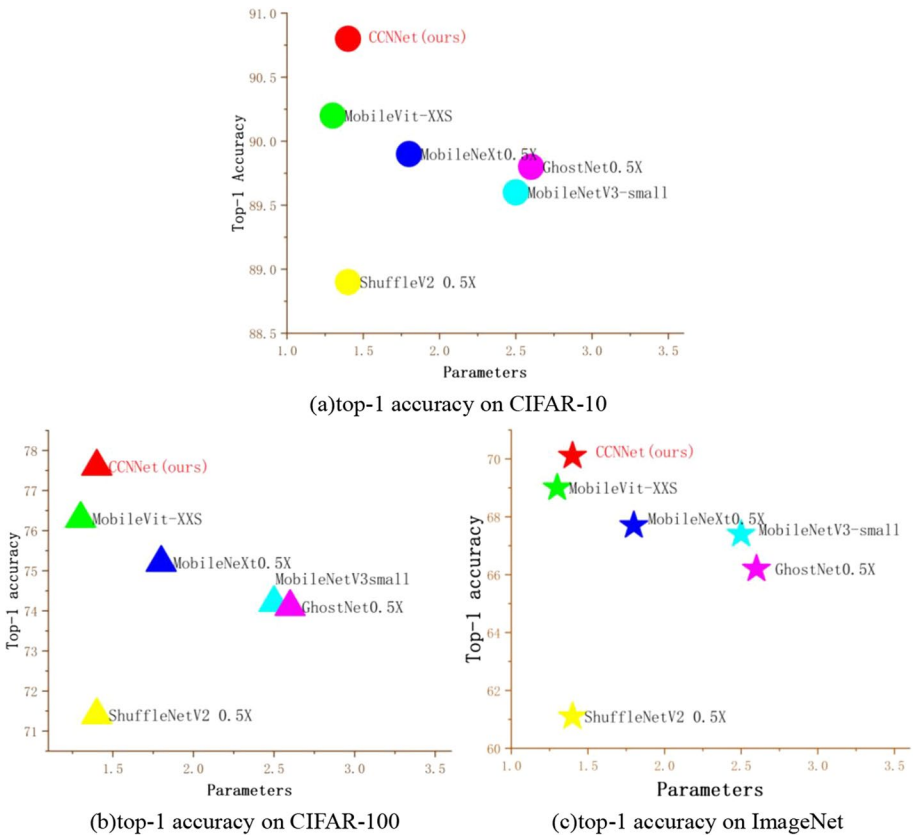


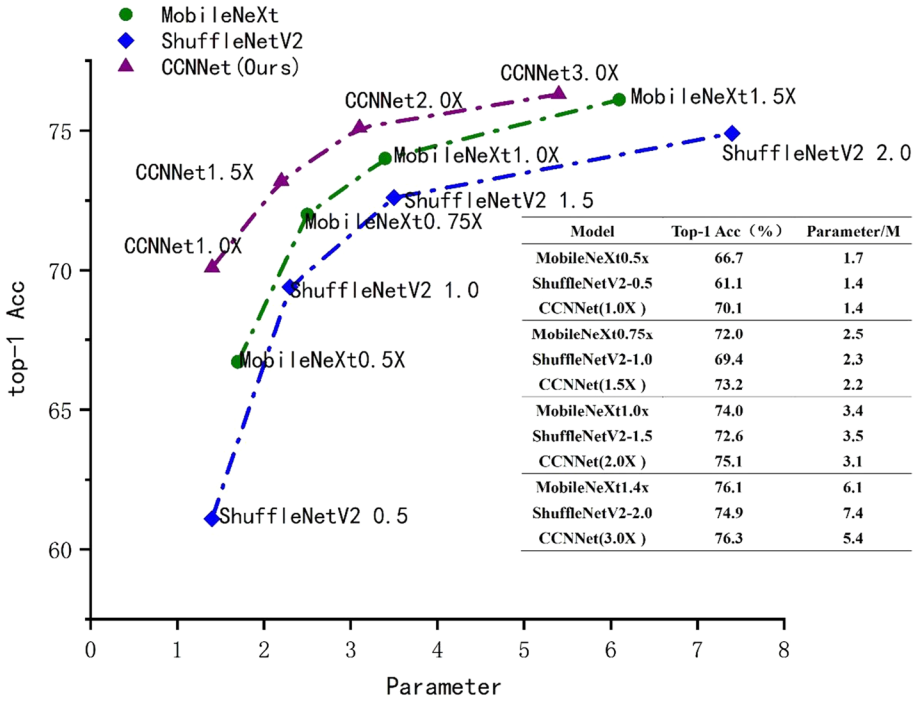

**Fig. 4** Comparison to some baseline models on ImageNet-1 k, CIFAR-10 and CIFAR-100. The red indicates our model performance

are superior in capturing channel feature information and global information. This proves the effectiveness of the proposed CCNNet model.

The CCNNet expanded model is trained on ImageNet [44], CIFAR-10, CIFAR-100 [43], and TCM-100 datasets to verify its robustness and generalization, the experimental results show that the CCNNet expanded model has better robustness and generalization. The comparison of the top-1 accuracy of CCNNet expanded models on 4 datasets is shown in Table 4.

**Table 4** Comparison of experimental data of CCNNet expanded models on 4 datasets

| Model | Parameters/M | Flops/B | Top-1 accuracy | | | |
|---|---|---|---|---|---|---|
| | | | ImageNet (%) | cifar-10 (%) | cifar-100 (%) | TCM-100 (%) |
| CCNNet(1.0X) | 1.4 | 0.36 | 70.1 | 90.8 | 77.6 | 86.8 |
| CCNNet(1.5X) | 2.1 | 0.46 | 73.2 | 91.9 | 79.4 | 87.3 |
| CCNNet(2.0X) | 3.1 | 0.55 | 75.1 | 93.8 | 80.1 | 89.2 |
| CCNNet(3.0X) | 5.4 | 0.74 | 76.3 | 94.7 | 81.6 | 92.5 |




| Model | Top-1 Acc (%) | Parameter/M |
|---|---|---|
| MobileNeXt0.5x | 66.7 | 1.7 |
| ShuffleNetV2-0.5 | 61.1 | 1.4 |
| CCNNet(1.0X) | 70.1 | 1.4 |
| MobileNeXt0.75x | 72.0 | 2.5 |
| ShuffleNetV2-1.0 | 69.4 | 2.3 |
| CCNNet(1.5X) | 73.2 | 2.2 |
| MobileNeXt1.0x | 74.0 | 3.4 |
| ShuffleNetV2-1.5 | 72.6 | 3.5 |
| CCNNet(2.0X) | 75.1 | 3.1 |
| MobileNeXt1.4x | 76.1 | 6.1 |
| ShuffleNetV2-2.0 | 74.9 | 7.4 |
| CCNNet(3.0X) | 76.3 | 5.4 |

**Fig. 5** The CCNNet(ours) Compare with ShuffleNetV2 and MobileNeXt

The CCNNet model compare with the advanced lightweight convolutional neural network model MobileNeXt[20] and the 3-stages ShuffleNetV2 [24] model on ImageNet dataset and find that the CCNNet model performs better with similar number of parameters. As shown in Fig. 5.

In case of similar number of parameters, we compare the expanded model CCNNet2.0X with other classical lightweight models on ImageNet-1 K dataset, including the latest lightweight model PP-LCNet[26]. It verified that the performance of CCNNet model is better, as shown in Table 5.

#### *The CCNNet model validation experiments on TCM-100*

The CCNNet is a classification model. We will conduct model classification accuracy experiments on the TCM-100 dataset. In order to prove that the CCNNet model can satisfy the requirements of model classification accuracy in traditional Chinese medicine recognition scenarios, we compare the performance of ShuffleNetV2 [24], MobileNeXt [20] and CCNNet on the traditional Chinese medicine dataset(TCM-100) respectively.

**Table 5** The comparison of accuracy between CCNNet2.0X and other models on ImageNet

| Model | Parameter/M | MAdd/M | Top1-Acc (%) |
|---|---|---|---|
| CCNNet2.0X(ours) | 3.1 | 546 M | 75.1 |
| MobileNeXt(1.0x) | 3.4 | 300 M | 74.0 |
| MobileNetV2 | 3.6 | 340 M | 72.3 |
| ShuffleNetV2(1.5) | 3.4 | 292 M | 72.6 |
| MobileNetV1 | 3.6 | 578 M | 70.9 |
| MobileNetV3small-1.25x | 3.6 | 100 M | 70.6 |
| PP-LCNet-1x | 3.0 | 161 M | 71.3 |

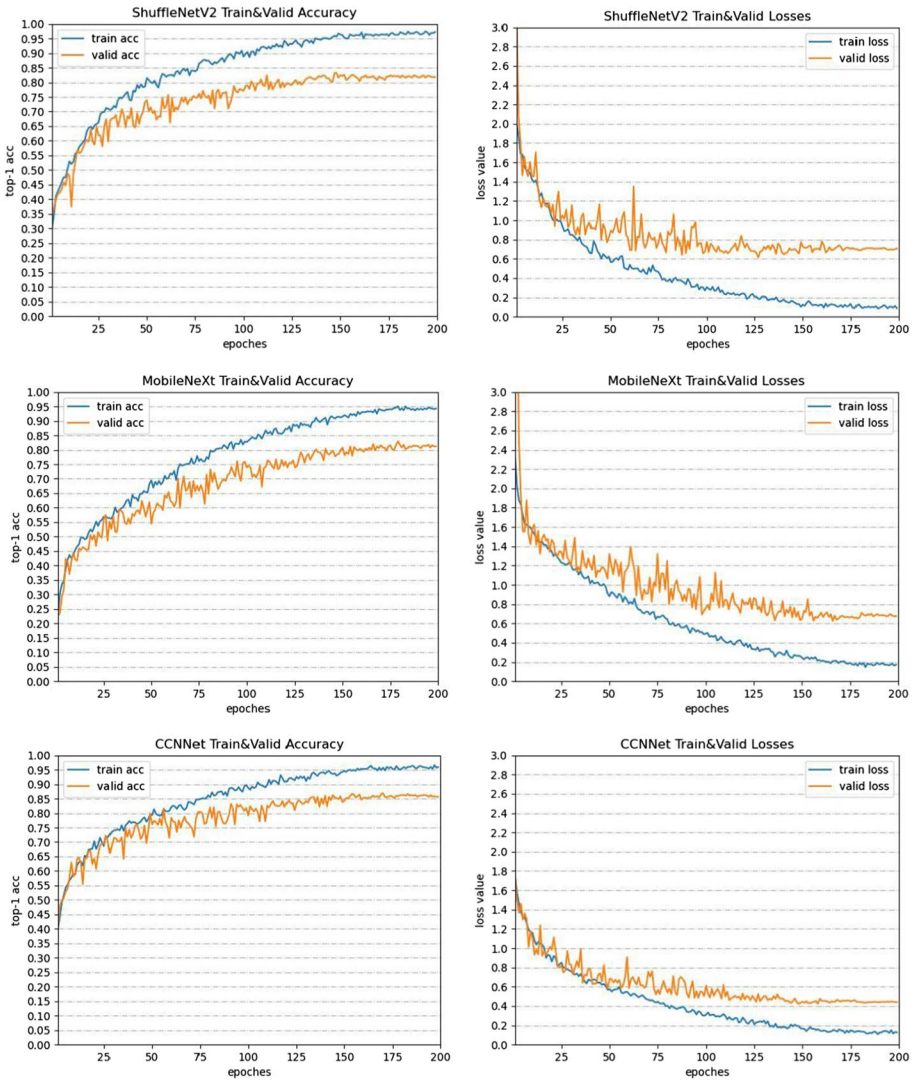



**Fig. 6** Comparison of CCNNet (ours), ShuffleNetV2, and MobileNeXt on TCM-100 dataset

The ratio of the training set and validation set is 5:1, i.e., 50000 images in the training set and 10000 images in the test set. Their training accuracy and loss, validation accuracy and loss are shown in Fig. 6. We found that the gap between the training accuracy

and the validation accuracy may be large, and we analyze the reason is due to the images in the dataset is not big enough, and the TCM-100 dataset should be extended subsequently.

The training and testing of ShuffleNetV2 [24], MobileNeXt [20] and CCNNet on TCM-100 dataset are shown in Table 6. The CCNNet model has the best performance and achieved 86.8% accuracy on the validation set.

In order to match the dual requirements of TCM recognition application scenarios and recognition accuracy, we adopt the CCNNet3.0X model to train and test on the TCM-100 dataset, and the classification accuracy reached 92.5%. In addition, the ROC curve is one of the evaluation criteria for the performance of the binary classification model. The ROC curve is a curve based on a series of different binary classification methods (cut-off values or thresholds), with the true positive rate(TPR) or sensitivity as the vertical coordinate and the false positive rate (FPR) as the horizontal coordinate.

We generalized the ROC curves of the dichotomous model and used two metrics, the Micro-ROC curve, and the Macro-ROC curve, as evaluation criteria [48–50]. The formulas of TPR and FPR are as follows.

$$TPR = \frac{TP}{TP + FN} \tag{6}$$

$$FPR = \frac{FP}{FP + TN} \tag{7}$$

In Eqs. 6 and 7, where TP represents the number of samples that are actually positive and predicted to be positive. FP denotes the number of samples that are actually positive but predicted to be negative. FN refers to the number of samples that are actually negative but evaluated to be positive. TN indicates the number of samples that are actually negative but predicted to be negative. The Micro method is to count the TP, FP, FN, and TN of each category, accumulate the TP, FP, FN, and TN of the whole non-categories sample set, and then calculates *Micro-TPR*, *Micro-FPR*. While the Macro method is to count the TP, FP, FN, and TN of each category, calculate their TPR and FPR respectively, and then take the average to get *Macro-TPR*, *Macro-FPR*. The formula is as follows.

$$Micro - TPR = \frac{\sum_{i=1}^{N} TP_i}{\sum_{i=1}^{N} TP_i + \sum_{i=1}^{N} FN_i} \tag{8}$$

**Table 6** The accuracy and loss of ShuffleNetV2, MobileNeXt, and CCNNet on TCM-100 dataset

| Metrics | ShuffleNetV2 | MobileNeXt | CCNNet1.0X(ours) |
|---|---|---|---|
| Training accuracy | 0.970 | 0.952 | 0.9706 |
| Training loss | 0.098 | 0.159 | 0.100 |
| Testing accuracy | 0.836 | 0.829 | 0.868 |
| testing loss | 0.599 | 0.633 | 0.421 |

$$Micro - FPR = \frac{\sum_{i=1}^{N} FP_i}{\sum_{i=1}^{N} FP_i + \sum_{i=1}^{N} TN_i} \tag{9}$$

$$Macro - TPR = \frac{1}{N} * \sum_{i=1}^{N} TPR_i \tag{10}$$

$$Macro - FPR = \frac{1}{N} * \sum_{i=1}^{N} FPR_i \tag{11}$$

In Eqs. 8, 9, 10, 11, N represents the total number of categories in the samples, and $TP_i$, $FP_i$, $FN_i$, and $TN_i$ denote the number of TP, FP, TN and FN in the samples of category i, respectively. We plotted the ROC curve to verify classification effect, as shown in Fig. 7. It can be seen that the model has a good classification performance on the TCM-100 dataset.

## Ablation study

### *The downsampling rate*

The training is performed on CIFAR-10, CIFAR-100, and ImageNet-1 K datasets following the same method as ConvNeXt [4]. We change the stem layer with 2 × downsampling, 3 × downsampling, and 4 × downsampling for training and comparison, and find that their model accuracies are comparable, and the gap of the model accuracy is not more than 0.1%. The results are shown in Table 7.

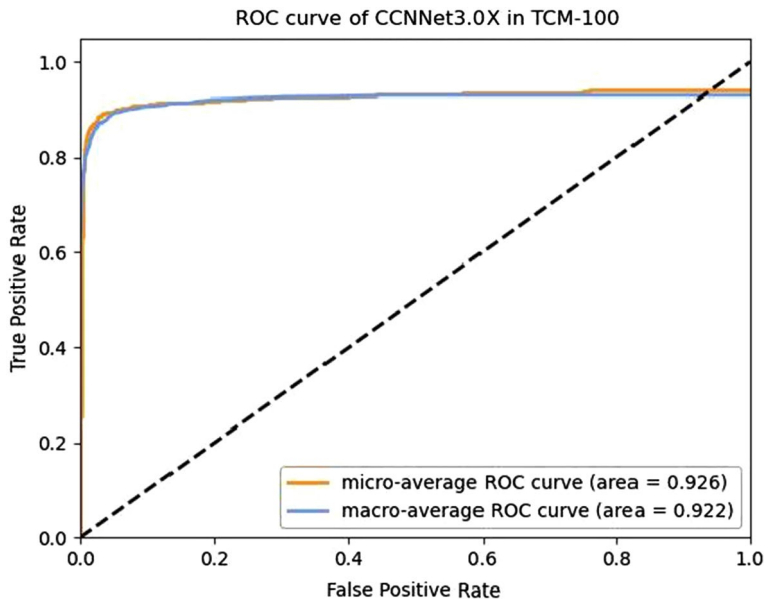



**Fig. 7** The ROC curve of CCNNet3.0X model in TCM-100 validation set

**Table 7** Comparison of the accuracy of ConvNeXt with different downsampling rates

| Model | Downsample rate | Epochs | Top-1 acc | | |
|---|---|---|---|---|---|
| | | | ImageNet (%) | Cifar-10 (%) | Cifar-100 (%) |
| ConvNeXt | 2 | 200 | 83.42 | 90.73 | 76.52 |
| | 3 | 200 | 83.39 | 90.69 | 76.48 |
| | 4 | 200 | 83.27 | 90.63 | 76.43 |

**Table 8** The comparison of accuracy of the CCNNet1.5X embedding different attentions

| Setting | Parameters/M | M-Adds/M | top-1Acc (%) |
|---|---|---|---|
| CCNNet + MDCA | 2.2 | 366 | 73.2 |
| CCNNet + SE | 2.2 | 366 | 72.7 |
| CCNNet + CBAM | 2.2 | 366 | 72.3 |
| CCNNet + CA | 2.35 | 381 | 73.3 |
| CCNNet + SimAM | 2.2 | 366 | 71.8 |

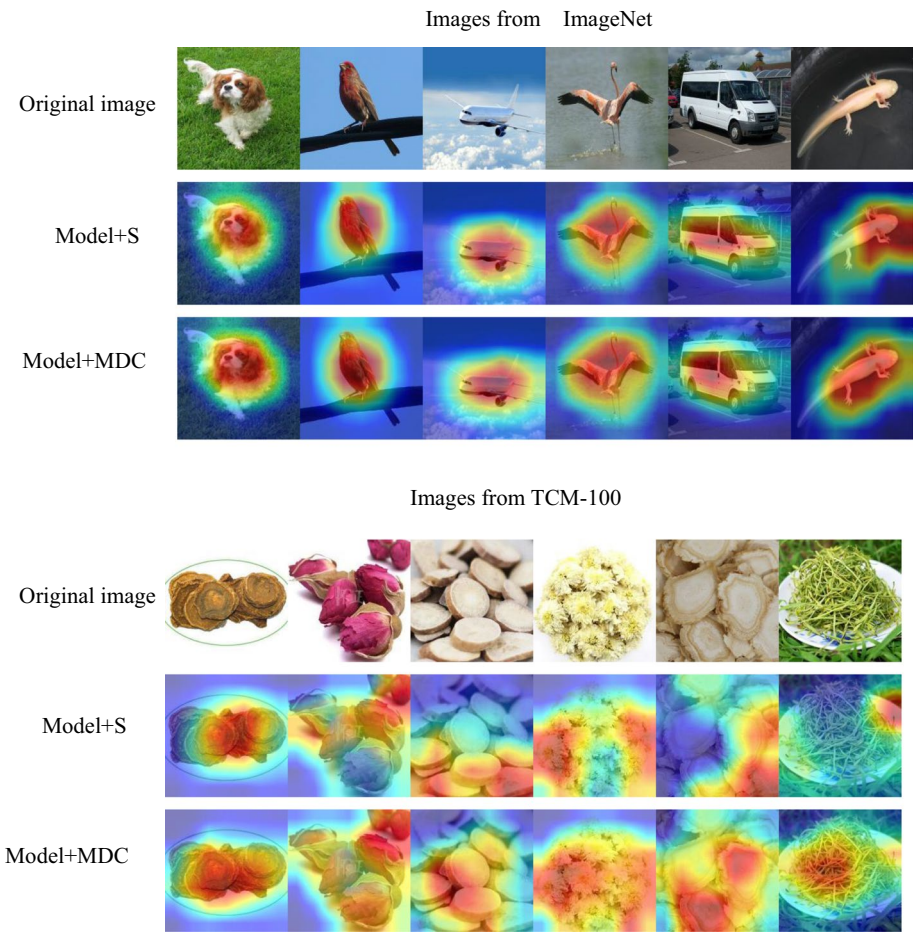



**Fig. 8** The visualization of the Feature map by Grad-CAM

### *MDCA attention module*

The performance of CCNNet-1.5X with various attentions on the ImageNet dataset is validated as shown in Table 8, where the model with CA [33] has the highest accuracy, but CA leads to an increment in the number of parameters. Adding MDCA after the depthwise separable operation of the GCIR module obtains 73.2% model accuracy with no increase in the number of parameters. This indicates that MDCA performs better without increasing the number of parameters.

Visualization of the MDCA attention module. We select some images from the ImagNet dataset and the TCM-100 dataset for visualization by Grad-CAM [51], respectively. The Grad-CAM feature maps with SE attention and the Grad-CAM feature maps with MDCA attention are shown in Fig. 8.

The results indicate that MDCA channel attention is more focused on the image features of interest, with a wide range of attention and a higher level attention compared to SE channel attention.

## Conclusion

In this paper, a lightweight model CCNNet is proposed for traditional Chinese medicine image classification, which consists of a GCIR module and a MDCA attention module, the GCIR module mainly extracts fine-grained feature information, and the MDCA attention module makes the model focus more on the feature information of interest. Thus, the model can extract the fine-grained feature information of interest, which is important for image classification and its downstream tasks such as object detection and semantic segmentation. Compared with existing lightweight classification models, the CCNNet model has higher accuracy and stronger robustness on ImageNet-1 K, CIFAR-10, and CIFAR-100 datasets with the approximate number of parameters, and the model can be scalable. Its expanded model CCNNet3.0X achieves 92.5% classification accuracy on the self-built TCM-100 dataset, which also indicates that our proposed CCNNet model has better generalizability. Compared with other lightweight models, the proposed model has the least number of parameters and higher accuracy, is more suitable for mobile deployment and applications.

**Acknowledgements**
Not applicable.

**Author contributions**
SG and HG majorly contributed to the design, implementation, and analysis of the research with the examination of the manuscript. HG, WX and JJ read and approved the final manuscript.

**Funding**
This study was supported by the National Natural Science Foundation of China (Grant No.51875454). This work was also supported by the Natural Science Foundation of Anhui Province Universities (No.KJ2020A0773, No.2022AH052413, No.2022AH052415).

**Availability of data and materials**
The datasets used and/or analysed during the current study are available from the corresponding author on reasonable request.

## Declarations

**Ethics approval and consent to participate**
Not applicable.

**Consent for publication**
Not applicable.

**Competing interests**
The authors declare that he has no competing interests.

Received: 12 January 2023 Accepted: 23 June 2023
Published online: 07 July 2023

**References**

1. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv. 2014. https://doi.org/10.48550/arXiv.1711.05101.
2. C Szegedy, W Liu, Y Jia, P Sermanet, S Reed, D Anguelov, D Erhan, V Vanhoucke, A Rabinovich. Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition, 1–9,2015.
3. K He, X Zhang, S Ren, J Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition. 770–778, 2016.
4. Z Liu, H Mao, C-Y Wu, C Feichtenhofer, T Darrell, S Xie. A convnet for the 2020s. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 11976–11986, 2022.
5. Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S, et al. An image is worth 16x16 words: transformers for image recognition at scale. ArXiv. 2020. https://doi.org/10.48550/arXiv.1711.05101.
6. Z Liu, Y Lin, Y Cao, H Hu, Y Wei, Z Zhang, S Lin, Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, 10012–10022, 2021.
7. Yan C, Li-si Z. Intelligent screening of pieces of chinese medicine based on bmfnet-wgan. Chin J Exp Trad Med For. 2021;24:107–14.
8. Yi Z, Hua W, Shuqin T. Technical review and case study on classification of chinese herbal slices based on computer vision. J Comp Appl. 2022;42(10):3224.
9. Chong W, Chao-qun T, Yong-liang H, Chun-jie W, Hu C. Intelligent identification of fritillariae cirrhosae bulbus, crataegi fructus and pinelliae rhizoma based on deep learning algorithms. Chin J Exp Trad Med Form. 2020;24:195–201.
10. Lili GENG, Baoning NIU. Survey of deep neural networks model compression. J Front Comp Sci Technol. 2020;14(9):1441–55.
11. B Zoph, V Vasudevan, J Shlens, QV Le. Learning transferable architectures for scalable image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, 8697–8710, 2018.
12. A Howard, M Sandler, G Chu, L-C Chen, B Chen, M Tan, W Wang, Y Zhu, R Pang, V Vasudevan, et al. Searching for mobilenetv3. In Proceedings of the IEEE/CVF international conference on computer vision, 1314–1324, 2019.
13. M Tan, Q Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In International conference on machine learning, 6105–6114. PMLR, 2019.
14. Zoph B, Le QV. Neural architecture search with reinforcement learning. arXiv. 2016. https://doi.org/10.48550/arXiv.1711.05101.
15. Andrew G, Menglong Zhu, et al. Efficient convolutional neural networks for mobile vision applications. Mobilenets. 2017. https://doi.org/10.48550/arXiv.1711.05101.
16. M Sandler, A Howard, M Zhu, A Zhmoginov, L-C Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE conference on computer vision and pattern recognition, 4510–4520, 2018.
17. Iandola FN, Han S, Moskewicz MW, Ashraf Khalid, Dally WJ, Keutzer K. Squeezenet Alexnet-level accuracy with 50x fewer parameters and <5.0 model size. arXiv. 2016. https://doi.org/10.4855/arXiv.1711.05101.
18. K Han, Y Wang, Q Tian, J Guo, C Xu, C Xu. Ghostnet: More features from cheap operations. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 1580–1589, 2020.
19. X Zhang, X Zhou, M Lin, J Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 6848–6856, 2018.
20. Zhou D, Hou Q, Chen Y, Feng J, Yan S. Rethinking bottleneck structure for efficient mobile network design. In: Vedaldi Andrea, Bischof Horst, Brox Thomas, Frahm Jan-Michael, editors. European Conference on Computer Vision. Cham: Springer; 2020.
21. Mehta S, Rastegari M. Mobilevit: light-weight, general-purpose, and mobile friendly vision transformer. arXiv. 2021. https://doi.org/10.48550/arXiv.1711.05101.
22. G Huang, Z Liu, L Van Der Maaten, KQ Weinberger. Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, 4700–4708, 2017.
23. Qin X, Wang Z. Nasnet: a neuron attention stage-by-stage net for single image deraining. arXiv. 2019. https://doi.org/10.48550/arXiv.1711.05101.
24. N Ma, X Zhang, H-T Zheng, J Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In Proceedings of the European conference on computer vision(ECCV), pages 116–131, 2018.
25. J Hu, L Shen, G Sun. Squeeze-and-excitation networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 7132–7141, 2018.
26. Cui C, Gao T, Wei S, Yuning D, Guo R, Dong S, Bin Lu, Zhou Y, Lv X, Liu Q, et al. Pp-lcnet: a lightweight cpu convolutional neural network. arXiv. 2021. https://doi.org/10.48550/arXiv.1711.05101.
27. M Tan, B Chen, R Pang, V Vasudevan, M Sandler, A Howard, QV Le. Mnasnet: Platform-aware neural architecture search for mobile. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 2820–2828, 2019.

28. Tan M, Le QV. Mixconv: Mixed depthwise convolutional kernels. arXiv. 2019. https://doi.org/10.4855/arXiv.1711.05101.
29. Q Wang, B Wu, P Zhu, P Li, W Zuo, Q Hu. Supplementary material for'eca-net: Efficient channel attention for deep convolutional neural networks. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, Seattle, WA, USA, pages 13–19, 2020.
30. Woo S, Park J, Lee J-Y, Kweon IS. Cbam: Convolutional block attention module. Proc Eur Conf Comp Vision (ECCV). 2018. https://doi.org/10.1007/978-3-030-01234-2_1.
31. Q-L Zhang, Y-B Yang. Sa-net: Shuffle attention for deep convolutional neural networks. In ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 2235–2239. IEEE, 2021.
32. D Misra, T Nalamada, AU Arasanipalai, Q Hou. Rotate to attend: Convolutional triplet attention module. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pages 3139–3148, 2021.
33. Q Hou, D Zhou, J Feng. Coordinate attention for efficient mobile network design. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 13713–13722, 2021.
34. L Yang, R-Y Zhang, L Li, X Xie. Simam: A simple, parameter-free attention module for convolutional neural networks. In International conference on machine learning, pages 11863–11874. PMLR, 2021.
35. A Vaswani, N Shazeer, N Parmar, J Uszkoreit, L Jones, AN Gomez, Ł Kaiser, I Polosukhin. Attention is all you need. Advances in neural information processing systems, 30, 2017.
36. Gao H, Xiao J, Yin Y, Liu T, Shi J. A mutually supervised graph attention network for few-shot segmentation: the perspective of fully utilizing limited samples. IEEE Trans Neural Netw Learn Syst. 2022. https://doi.org/10.1109/TNNLS.2022.3155486.
37. Chen J, Ying H, Liu X, Jingjing G, Feng R, Chen T, Gao H, Jian W. A transfer learning based super-resolution microscopy for biopsy slice images: the joint methods perspective. IEEE/ACM Trans Comput Biol Bioinform. 2020;18(1):103–13.
38. J Guo, K Han, H Wu, Y Tang, X Chen, Y Wang, C Xu. Cmt: Convolutional neural networks meet vision transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022. 12175–12185.
39. S Ioffe, C Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International conference on machine learning, 2015. 448–456. PMLR.
40. Hendrycks D, Gimpel K. Gaussian error linear units (gelus). arXiv. 2016. https://doi.org/10.48550/arXiv.1711.05101.
41. Larsson G, Maire M, Shakhnarovich G. Fractalnet: ultra-deep neural networks without residuals. arXiv. 2016. https://doi.org/10.48550/arXiv.1711.05101.
42. Q Hou, L Zhang, M-M Cheng, J Feng. Strip pooling: Rethinking spatial pooling for scene parsing. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 4003–4012, 2020.
43. A Krizhevsky, G Hinton, et al. Learning multiple layers of features from tiny images.2009.
44. J Deng, W Dong, R Socher, L-J Li, K Li, L Fei-Fei. Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition, pages 248–255. IEEE, 2009.
45. Do T-N. Incremental and parallel proximal svm algorithm tailored on the jetson nano for the imagenet challenge. Int J Inform Syst. 2022. https://doi.org/10.1108/IJWIS-03-2022-0055.
46. Wang J, Mo W, Yan W, Xiaomei X, Li Yi, Ye J, Lai X. Combined channel attention and spatial attention module network for chinese herbal slices automated recognition. Front Neurosci. 2022. https://doi.org/10.3389/fnins.2022.920820.
47. Loshchilov I, Hutter F. Decoupled weight decay regularization. arXiv. 2017. https://doi.org/10.48550/arXiv.1711.05101.
48. Balasubramaniam S, Gollagi SG. Software defect prediction via optimal trained convolutional neural network. Adv Engi Softw. 2022;169: 103138.
49. Daniya T, Vigneshwari S. A novel moore-penrose pseudo-inverse weight-based deep convolution neural network for bacterial leaf blight disease detection system in rice plant. Adv Eng Softw. 2022;174: 103336.
50. Kathamuthu ND, Subramaniam S, Le QH, Muthusamy S, Panchal H, Sundararajan SCM, Alrubaie AJ, Zahra MMA. A deep transfer learning-based convolution neural network model for covid-19 detection using computed tomography scan images for medical applications. Adv Eng Softw. 2023;175: 103317.
51. R Selvaraju, M Cogswell, A Das, R Vedantam, D Parikh, D Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE international conference on computer vision. 2017. 618–626.


## Publisher's Note