IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 27, 2022, accepted February 7, 2022, date of publication February 9, 2022, date of current version February 18, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3150320

CMSEA: Compound Model Scaling With Efficient Attention for Fine-Grained Image Classification

JINZHENG GUANG AND JIANRU LIANG
College of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai 201620, China
Corresponding author: Jinzheng Guang (guangjinzheng@qq.com)

ABSTRACT The purpose of fine-grained image classification is to distinguish subcategories belonging to the same basic-level category, for example, two hundred subcategories belonging to birds. It has been a challenging topic in the field of computer vision in recent years due to the small inter-class variance among different subcategories (e.g., color and texture) and the large intra-class variance in the same subcategory (e.g., pose and viewpoint). In this paper, we propose a Compound Model Scaling with Efficient Attention (CMSEA) for fine-grained image classification, which carefully balances the various dimensions of width, depth, and image resolution in model scaling. Furthermore, the proposed method utilizes an additional computational low attention module to efficiently learn subtler features from discriminative regions. In addition, regularization and data augmentation were employed to improve accuracy in the training. Extensive experiments demonstrate that CMSEA achieves 90.63%, 94.51%, and 95.19% accuracy on CUB-200-2011, FGVC-Aircraft, and Stanford Cars datasets, respectively. In particular, CMSEA on CUB-200-2011 obtains 2.3% higher accuracy with 18% fewer network parameters than the original approach. Consequently, our method has better accuracy and parameter efficiency compared to most existing methods.

INDEX TERMS Fine-grained image classification, EfficientNet, image recognition, channel attention, convolutional neural networks.

I. INTRODUCTION

Fine-Grained Visual Categorization (FGVC) is one of the most fascinating and prominent research topics in the field of computer vision in recent years [1]. Fine-grained image classification, unlike coarse-grained image classification, tries to recognize several hundred subcategories that all correspond to the same basic level category, such as two hundred subcategories for birds [2], one hundred subcategories for planes [3], and 196 subcategories for cars [4]. As illustrated in Figure 1, fine-grained image classification recognizes different subcategories under the same category. For example, yellow-headed blackbirds and red-winged blackbirds belong to the subcategory of blackbirds. As presented in Figure 2, the main challenge of FGVC is due to the large intra-class variance (red rectangular box) and small inter-class variance (black rectangular box). These subcategories are usually similar in overall appearance and are distinguished by subtle variations, such as the color of heads, the texture of feathers, and the shape of toes in birds. Furthermore, these small changes may be found in regions of the object or its parts. Most people without professional knowledge can easily identify basic categories, such as birds, airplanes, and cars, but it is extremely difficult to distinguish two hundred or more subcategories.

The key to FGVC is how to locate distinguishing regions and learn fine-grained features from these regions. Strongly supervised learning methods and weakly supervised learning methods are the two primary groups of recent FGVC algorithms. Strongly supervised learning methods employ some extra information (e.g., bounding box/part annotations) in the image to locate discriminative regions [5]. Unfortunately, it is not feasible due to high labor expenses and the absence of precise bounding box/part annotations in the real deployment. Weakly supervised learning methods utilize only image-level labels in training and automatically locate discriminative regions through various attention mechanisms [6]–[9]. Recent research on the fine-grained image has mainly focused on the latter. While attention mechanism approaches (e.g., MAMC [10]) can achieve excellent results, they increase model complexity. Furthermore, most

The associate editor coordinating the review of this manuscript and approving it for publication was Orazio Gambino.

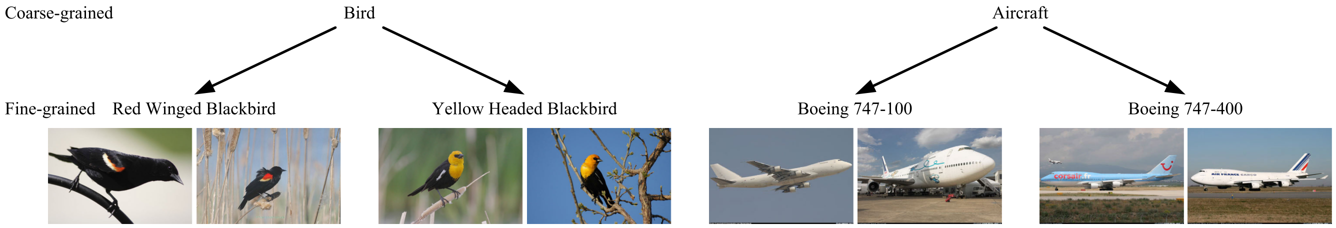


FIGURE 1. Some examples of attribute hierarchies on CUB-200-2011 [2] and FGVC-Aircraft [3]. Coarse-grained image classification recognizes different categories (e.g., birds and aircraft), while fine-grained image classification identifies different subcategories under the same category. For example, red-winged blackbirds and yellow-headed blackbirds belong to the subcategory of blackbirds.

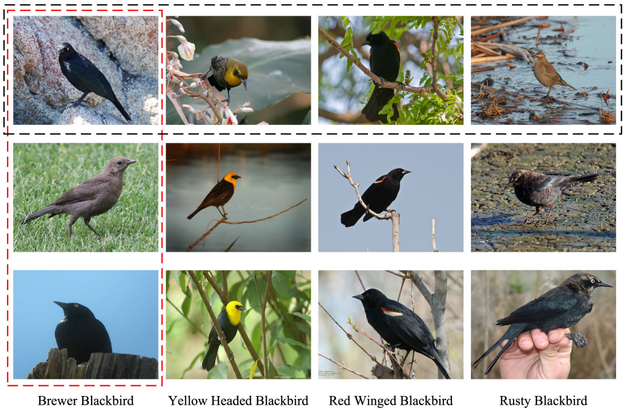


FIGURE 2. Some examples on CUB-200-2011 [2]. Classifying them, even for humans, is an extremely challenging task due to the large intra-class variance (red rectangular box) and small inter-class variance (black rectangular box).

networks in previous work commonly scale only one of the three dimensions (i.e., depth, width, and image resolution) to learn subtle features. For instance, the depth can be scaled up by using more layers (e.g., from ResNet-50 [11] to ResNet-101 [10]), but it will increase the model complexity By introducing numerous network parameters.

To deal with the above problems, we propose a Compound Model Scaling with Efficient Attention (CMSEA) for fine-grained image classification without extra artificial labeling information (e.g., bounding box/part annotations). First of all, CMSEA utilizes EfficientNetV2-S [12] as the basic backbone, which carefully balances each dimension of network width, depth, and image resolution in model scaling. Then, regularization and data augmentation are employed to improve accuracy in the training. Furthermore, we replace all two fully-connected layers (FC) in the attention modules with a fast 1D convolution, which involves only a handful of network parameters while learning subtler features from discriminative regions efficiently. Experimental results in Table 1 demonstrate our method is effective. The following is a list of contributions to our work:

TABLE 1. Experimental results. Note that our method achieves better accuracy with fewer network parameters.

Model	CUB-200-2011	FGVC-Aircraft	Stanford Cars	Params
Original	88.31%	91.96%	93.30%	20.43M
Ours	**90.63%**(+2.3%)	**94.51%**(+2.5%)	**95.19%**(+1.9%)	**16.73M**(-18%)

- We propose a Compound Model Scaling with Efficient Attention (CMSEA), which carefully balances the various dimensions of width, depth, and image resolution in model scaling.
- We utilize an additional computational low attention module to efficiently learn subtler features from discriminative regions.
- Extensive experiments demonstrate our CMSEA achieves state-of-the-art performance on three well-known fine-grained image datasets.

The following is the rest of this paper: Section 2 briefly describes related work on FGVC. Section 3 introduces our CMSEA method, and experimental results and analysis are given in Section 4. Section 5 is the conclusion of this paper.

II. RELATED WORK

Many researches have been conducted recently to improve model performance. The accuracy of FGVC is influenced by two major factors: discriminative region localization and discriminative feature extraction. This section reviews the related work in these two aspects.

A. DISCRIMINATIVE REGION LOCALIZATION

To localize discriminative regions, early part localization approaches [5] recommended utilizing some extra information (e.g., bounding box/part annotations). However, this strategy was rendered unfeasible due to the high labor expenses and the absence of precise part annotation in the real deployment. In recent years, weakly supervised FGVC methods [13]–[26] have attempted to localize distinguishable regions using only image-level labels. For example, RA-CNN [14] is a recurrent attention network for learning discriminative areas on various scales in an iterative manner.

Some recent researches have attempted to locate the distinguishing regions via designing some complex attention modules. DCL [17] automatically detected the distinguished regions through a region confusion mechanism. Following that, DCL uses a jigsaw puzzle generator to learn multi-grained areas gradually. A trilinear attention module was suggested by TASN [20], which converts the convolutional feature mapping into an attention mapping. Consequently, these approaches utilize more complex attention modules to accurately locate critical discrimination regions, however, they undoubtedly increase model complexity. In contrast, we have attempted to utilize a lightweight attention module to locate the discriminative regions.

B. DISCRIMINATIVE FEATURE EXTRACTION

How to discover discriminative features is a critical challenge in image classification. Some recent techniques have concentrated on extracting discriminative characteristics from prominent areas, which are frequently represented by attention maps. These techniques may be divided into two groups. The first group [14]–[16] crop substantial portions of the original image intentionally. They use attention techniques to adaptively pick acceptable bounding boxes to accurately pinpoint the differentiated regions, which is inspired by object detection. The second group [20]–[22] uses local amplification and sampling to emphasize the important locations. S3N [21], for example, processes images using non-uniform sampling based on 2D probability. Although these methods can yield excellent results, they increase the model complexity due to introducing numerous network parameters.

In addition, most networks in previous work commonly scaled only one of the three dimensions of the network to learn fine-grained features. As illustrated in Figure 4, depth can be scaled up by using more layers (e.g. from ResNet-50 [11] to ResNet-101 [10]). Another common method is to scale up models by image resolution (e.g. from 224 × 224 to 448 × 448). They will undoubtedly introduce a significant number of network parameters, even though they can improve accuracy. Unlike them, our CMSEA carefully balances the three dimensions in model scaling while efficiently learning the subtle features.

III. PROPOSED METHOD

In this section, we carefully study EfficientNet architecture and channel attention modules and then introduce our CECAMBConv module and CMSEA architecture.

A. OVERVIEW

The key to FGVC is how to locate the distinguishing regions and learn the subtle features from them. Figure 3 illustrates the general workflow of our network to discriminative areas. Specifically, we use the CECAMBConv attention module to locate distinguishing regions, such as the head, torso, and tail of the bird. We then learn the subtle features through CMSEA, which carefully balances the various dimensions of network width, depth, and image resolution in model scaling.

B. COMPOUND MODEL SCALING

Scaling up Convolutional Neural Networks (CNNs) is commonly used to obtain higher accuracy. However, CNNs in previous work were typically scaled in only one of the three dimensions – depth [27], width [28], and image resolution [29]. As illustrated in Figure 4, depth can be scaled by using more layers (e.g., from ResNet-50 [11] to ResNet-101 [10]). It normally introduces numerous network parameters and thus increases the model complexity. Another common approach is to scale up models by image resolution (e.g., from 224 × 224 to 448 × 448). Although they can improve accuracy by scaling arbitrary dimensions of depth, width, and image resolution, the model performance is usually sub-optimal. Therefore, we utilize EfficientNet [30] to carefully balance each dimension of network width, depth, and image resolution in model scaling.

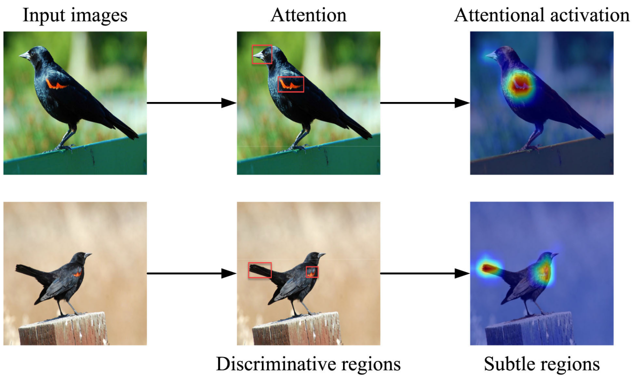


FIGURE 3. The workflow of our network. We first select different parts of the input image, such as the head, body, and tail of the bird, and then use the attention module to focus attention on more subtle regions. This red area indicates the region where activation is the center of attention.

TABLE 2. EfficientNet family. Each row describes a model with width coefficient w, depth coefficient d, resolution, Params, and FLOPs.

Model	w	d	Resolution	Params	FLOPs
EfficientNet-B0	1.0	1.0	224^2	5.3M	0.4B
EfficientNet-B1	1.0	1.1	240^2	7.8M	0.7B
EfficientNet-B2	1.1	1.2	260^2	9.2M	1.0B
EfficientNet-B3	1.2	1.4	300^2	12M	1.8B
EfficientNet-B4	1.4	1.8	380^2	19M	4.2B
EfficientNet-B5	1.6	2.2	456^2	30M	9.9B
EfficientNet-B6	1.8	2.6	528^2	43M	19B
EfficientNet-B7	2.0	3.1	600^2	66M	37B

EfficientNet is a family of CNNs that achieve better accuracy and parameter efficiency than previous CNNs. The baseline EfficientNet-B0 was designed using neural architecture search and the best values are $d = 1.2, w = 1.1$, and $r = 1.15$, under the constraint of $d \cdot w^2 \cdot r^2 \approx 2$. Table 2 was obtained by fixing d, w, and r as constants and then using a compound coefficient λ to uniformly scale network w, d, and r:

$$depth = d^{\lambda}, \tag{1}$$

$$width = w^{\lambda}, \tag{2}$$

$$resolution = r^{\lambda}, \tag{3}$$

where d, w, and r are constants that could be obtained by neural architecture search. Additionally, λ is a user-defined coefficient that determines the number of additional resources available for model scaling.

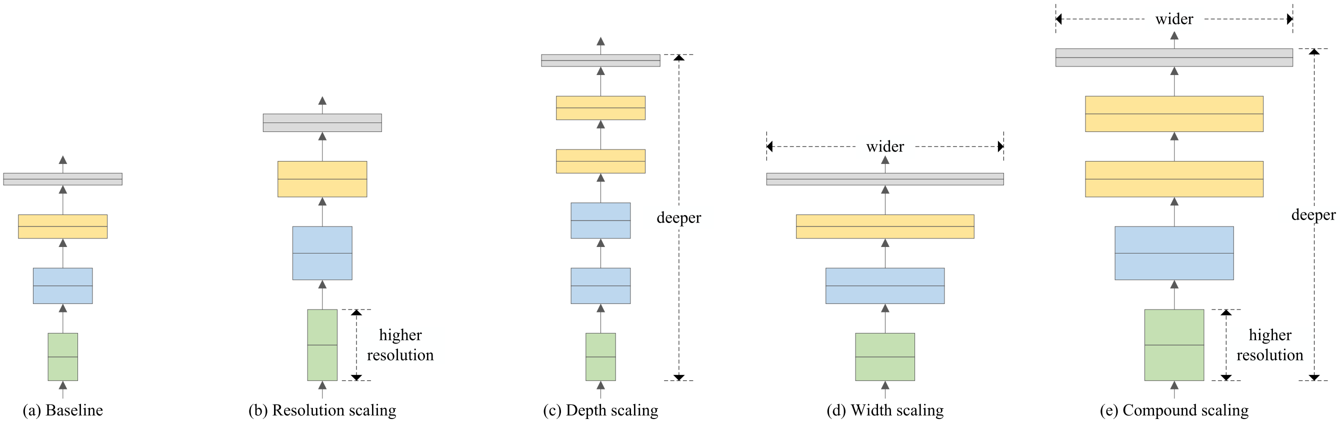


FIGURE 4. **Model scaling. (a) is a baseline network example; (b)-(d) are normal scaling examples that enhance the network image resolution, depth, or width in only one dimension. (e) is a three-dimensional composite scaling approach.**

As shown in Table 2, the image resolution of EfficientNet-B5 (456 × 456) is closest to the image resolution of most previous work (448 × 448). For a fair comparison, we have adjusted the image resolution of EfficientNet-B5 to 448 × 448. As a result, it balances each dimension of the network in model scaling.

C. FUSED-MBConv MODULE

The core structure of EfficientNet is the Mobile inverse Bottleneck Convolution module (MBConv), which is similar to the inverse residual block in MobileNet [31]. The MBConv in Figure 5 consists of an expansion convolution (conv1 × 1), a depthwise convolution (conv3 × 3), an SE block [32], and a regular two-dimensional convolution (Conv2D). However, the training speed of EfficientNet is very slow due to the very large image sizes. The Fused-MBConv in Figure 5 replaces the depthwise convolution and expansion convolution in MBConv with a single regular two-dimensional convolution. It jointly optimizes training speed and parameter efficiency.

D. EFFICIENT ATTENTION MODULE

As illustrated in Figure 5, we present a Circular Efficient Channel Attention block (CECAMBConv), which is an additional computational low attention module to efficiently learn the subtle features from discriminative regions. Specifically, it employs a fast 1D convolution with kernel size (k) to replace two FC layers. The following is a detailed analysis.

We first briefly review the commonly used channel attention modules [33]. Then, the dimensionality reduction effect of the attention module is analyzed. Finally, we give the module without dimensionality reduction. Let the output of the convolution block be

$$X \in R^{W \times H \times C}, \tag{4}$$

where W, H, and C are the width, height, and channel dimension, respectively. Consequently, the weights of channels in the SE block [32] can be computed as

$$w = \sigma(f_{\{W_1, W_2\}}(g(X))), \tag{5}$$

$$\sigma(Z) = \frac{1}{1+e^{-Z}}, \tag{6}$$

$$g(X) = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} u_c(i,j), \tag{7}$$

where $\sigma(Z)$ is a sigmoid function and $g(X)$ is channel-wise global average pooling (GAP). Let $Y = g(x)$,

$$f_{\{W_1, W_2\}}(Y) = W_2 \delta(W_1 Y), \tag{8}$$
$$\delta(Z) = max(0, Z), \tag{9}$$

where $W_1 = C \times (\frac{C}{r})$, $W_2 = (\frac{C}{r}) \times C$, and $\delta(Z)$ indicates the ReLU function. The reduction ratio r is a hyperparameter that allows us to change the capacity and calculate the cost. We can observe that $f_{\{W_1, W_2\}}$ involves all network parameters of attention. Although Eq. (8) simplifies the complexity of the channel attention module, it also eliminates the direct relationship between its weights and channels. Eq. (8), in particular, first projects the channel characteristics into low-dimensional space and then maps them back, resulting in an indirect correlation between the channels and their weights.

In this paper, we explored a method to ensure both effectiveness and efficiency. Specifically, given the aggregated feature $Y \in R^C$ without dimensionality reduction, channel attention can be learned by

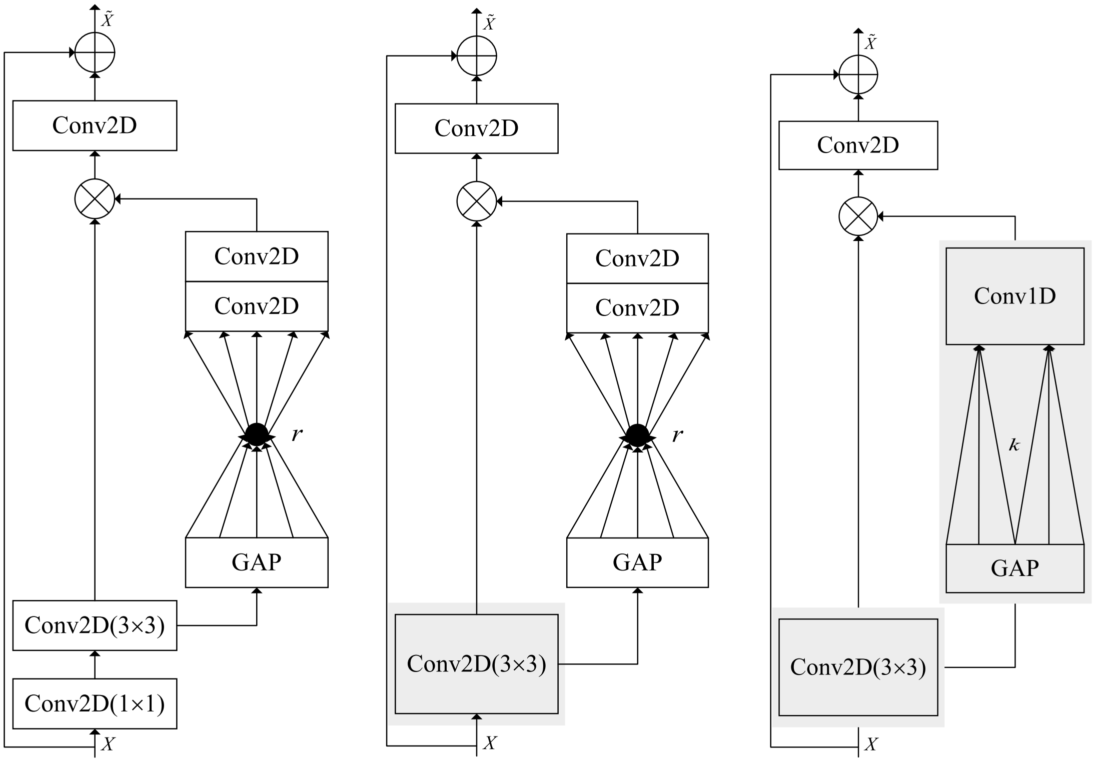
$$w = \sigma(WY), \tag{10}$$

where W is a $C \times C$ parameter matrix. In particular,

$$\mathrm{W}_{SED} = \begin{bmatrix} w^{1,1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & w^{C,C} \end{bmatrix}, \tag{11}$$

learns the weight of each channel independently. Note that W_{SED} is a diagonal matrix, involving $1 \times C$ parameters. Meanwhile,

$$W_{SEF} = \begin{bmatrix} w^{1,1} & \cdots & w^{1,C} \\ \vdots & \ddots & \vdots \\ w^{C,1} & \cdots & w^{C,C} \end{bmatrix}, \tag{12}$$



(a) **MBConv.** (b) **Fused-MBConv.** (c) **CECAMBConv.**

FIGURE 5. Structure of MBConv, Fused-MBConv, and CECAMBConv. Fused-MBConv replaces the depthwise convolution (conv3 × 3) and expansion convolution (conv1 × 1) in MBConv with a single regular two-dimensional convolution (conv3 × 3). Our CECAMBConv replaces the two FC layers with a fast one-dimensional convolution with the kernel size (*k*), which avoids dimensionality reduction and effectively learns the discriminative features. Conv2D, GAP, *k*, and *r* denote two-dimensional convolution, global average pooling, kernel size, and reduction ratio, respectively.

TABLE 3. Comparison of the channel attention modules - *C1D* denotes 1D convolution, constant *k* denotes kernel size of *C1D*, *r* denotes reduction ratio and *C* denotes channel dimension.

Module	Attention	Params
MBConv	$\sigma(f_{\{W_1,W_2\}}(Y))$	$2 \times C^2/r$
CECAMBConv	$\sigma(C1\mathrm{D}_k(Y))$	$k = 7$

employs one single FC layer with dimensionality reduction in the SE block. Note that W_{SEF} is a full matrix, involving $C \times C$ parameters.

As discussed above, we use a band matrix W_k to learn channel attention, and W_k has

$$\begin{bmatrix} w^{1,1} & \cdots & w^{1,k} & 0 & 0 & \cdots & \cdots & 0 \\ 0 & w^{2,2} & \cdots & w^{2,k+1} & 0 & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & w^{C,C-k+1} & \cdots & w^{C,C} \end{bmatrix}. \tag{13}$$

Note that W_k involves $k \times C$ parameters. Obviously, let $k = 1$, W_k takes the form W_{SED}, and let $k = C$, W_k takes the form W_{SEF}. Thus w takes the form

$$w = \sigma(WY) = \sigma(W_k Y). \tag{14}$$

As for w and W_k, the weight of Y_i is calculated by only considering the interaction between Y_i and its k neighbors, i.e.,

$$w_i = \sigma(\sum_{j=1}^{k} w_i^j Y_i^j),\ Y_i^j \in \Omega_i^k, \tag{15}$$

where Ω_i^k denotes the set of k adjacent channels of Y_i.

$$w_i = \sigma(\sum_{j=1}^{k} w^j Y_i^j),\ Y_i^j \in \Omega_i^k. \tag{16}$$

is a more efficient way to make all channels share the same learning parameters. The way can be readily implemented by a fast 1D convolution with a kernel size k, i.e.,

$$w = \sigma(C1\mathrm{D}_k(Y)), \tag{17}$$

where $C1D$ denotes 1D convolution, and k denotes a kernel size of $C1D$. The 1D convolution uses circular padding rather than zero padding. The module is called by CECAMBConv with k parameters, which guarantees both effectiveness and efficiency.

E. CMSEA ARCHITECTURE

Figure 6 and Table 4 illustrate the overview of our CMSEA architecture. The structure of CMSEA consists of ten

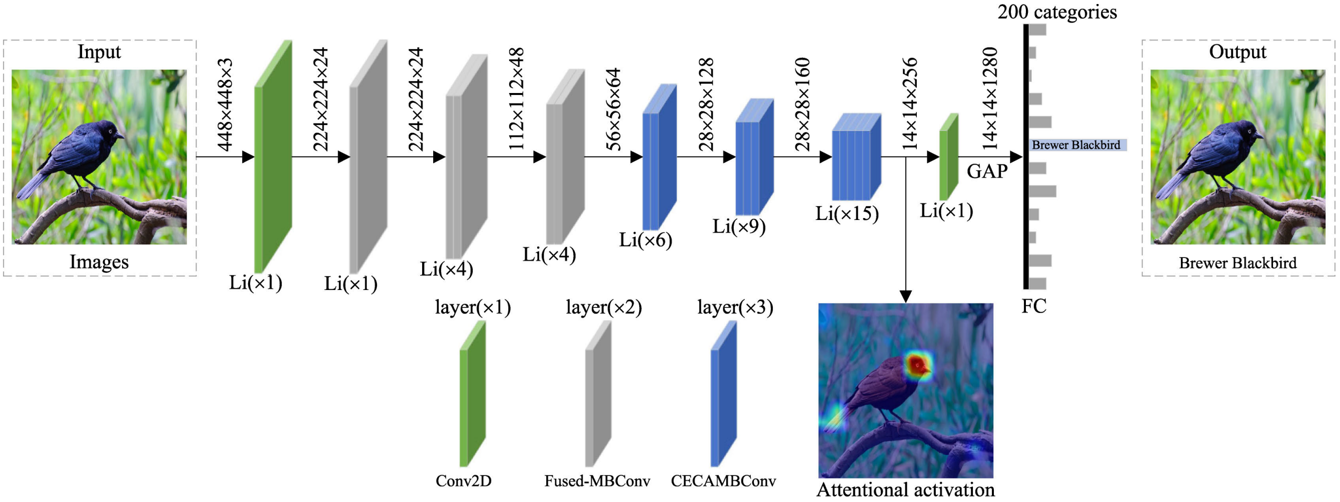


FIGURE 6. Our CMSEA architecture. Li(×num) denotes num layers, and $H \times W \times C$ denotes tensor shape (Height × Width × Channel).

TABLE 4. CMSEA architecture – Fused-MBConv and CECAMBConv blocks are described in Figure 5.

Stage	Operator	Resolution	Channels	Layers
0	Conv3×3	448 × 448	24	1
1	Fused-MBConv1, k3×3	224 × 224	24	2
2	Fused-MBConv4, k3×3	224 × 224	48	4
3	Fused-MBConv4, k3×3	112 × 112	64	4
4	CECAMBConv4, k3×3, k7	56 × 56	128	6
5	CECAMBConv6, k3×3, k7	28 × 28	160	9
6	CECAMBConv6, k3×3, k7	28 × 28	256	15
7	Conv1×1 & Pooling & FC	14 × 14	1280	1

Fused-MBConv, thirty CECAMBConv, two regular convolutional layers, and an FC layer. The main difference of our CMSEA is that the original MBConv module was replaced by the new CECAMBConv modules. As presented in Table 3 and Figure 6, the network parameters of 1D convolution and two FC layers are k and $2 \times C^2/\gamma$, respectively. Obviously, 1D convolution has fewer network parameters than two FC layers. Our CMSEA in channel attention employs a fast 1D convolution to replace two FC layers. Our CMSEA has thirty CECAMBConv modules, and then sixty FC layers are reduced. Therefore, our CMSEA has fewer network parameters.

IV. EXPERIMENTS

In this section, we will evaluate our CMSEA on three popular fine-grained image datasets. We first introduce the experimental setup including datasets, evaluation metrics, and details of implementation. Comparison with state-of-the-art methods followed by ablation studies. We finally give the analysis and visualization results.

A. DATASETS

As shown in Table 5, we conducted experiments on three well-known fine-grained image classification datasets, CUB-200-2011 [2], FGVC-Aircraft [3], and Stanford Cars [4]. CUB-200-2011 (CUB) is one of the most widely used datasets in fine-grained classification, with 200 different subcategories containing 5,994 training images and 5,794 test images. FGVC-Aircraft (AIR) is an aircraft dataset containing 100 subcategories consisting of 10,000 images equally divided into training, test, and validation. Stanford Cars (CAR) includes 16,185 vehicle images, including 8,144 for training and 8,041 for testing, with 24-84 training images and 24-83 testing images in each subcategory. Figure 7 shows a partial example of the three fine-grained image datasets.

TABLE 5. Detailed statistics of the three used datasets.

Dataset	Category	Training	Testing
CUB-200-2011 [2]	200	5,994	5,794
FGVC-Aircraft [3]	100	6,667	3,333
Stanford Cars [4]	196	8,144	8,041

B. EVALUATION METRICS

The evaluation metrics include accuracy, network parameters (Params), and floating point operations (FLOPs). Accuracy in Eq. (18) is used to evaluate the classification performance of our CMSEA method. FLOPs and Params are measures of model time and spatial complexity, respectively.

$$Accuracy = \frac{Rc}{Ra}, \tag{18}$$

where Ra is the total number of test images and Rc is the total number of images correctly classified in the test phase.

C. DETAILS OF IMPLEMENTATION

All models in this paper are trained on a single NVIDIA GeForce RTX 2080 SUPER GPU. For a fair comparison, we followed most of our previous works using an image resolution of 448 × 448. We train on a standard training set

(a) **CUB-200-2011.** (b) **FGVC-Aircraft.** (c) **Stanford Cars.**

FIGURE 7. Several examples of three datasets with one example per species.

TABLE 6. Training hyper-parameters setup.

Name	Value
Resolution	448 × 448
Epochs	200
Batch size	16
Optimizer	RMSprop
Learning rate	0.0003
Dropout	0.3
Drop connect	0.2
Rand Augment	9/0.5
Erasing prob.	0.2

TABLE 7. Comparison results of our method (CMSEA) on the CUB-200-2011 dataset.

Method	Year	Backbone	Accuracy(%)
OPAM [39]	2017	VGG-16	85.83
Refined-CNN [40]	2017	VGG-16	86.4
RA-CNN [14]	2017	VGG-19	85.3
MA-CNN [15]	2017	VGG-19	86.5
Kernel-Pooling [41]	2017	ResNet-50	84.7
DFL-CNN [42]	2018	ResNet-50	87.4
iSQRT-COV [11]	2018	ResNet-50	88.1
MAMC [10]	2018	ResNet-101	86.5
HBPASM [43]	2019	ResNet-34	86.8
DBTNet-50 [44]	2019	ResNet-50	87.5
Cross-X [45]	2019	ResNet-50	87.7
DCL [17]	2019	ResNet-50	87.8
TASN [20]	2019	ResNet-50	87.9
S3N [21]	2019	ResNet-50	88.5
MGE-CNN [16]	2019	ResNet-101	89.4
MS-SRP-D [46]	2020	ResNet-50	85.5
BBPL [47]	2021	ResNet	87.62
MFF [48]	2021	ResNet-34	87.1
SMA-Net [49]	2021	ResNet-50	87.71
MSEC [50]	2021	ResNet-50	88.3
SSSNet [22]	2021	ResNet-50	89.0
GHNS [51]	2021	ResNet-50	89.06
CMSEA(Ours)	-	EfficientNetV2-S	**90.63**

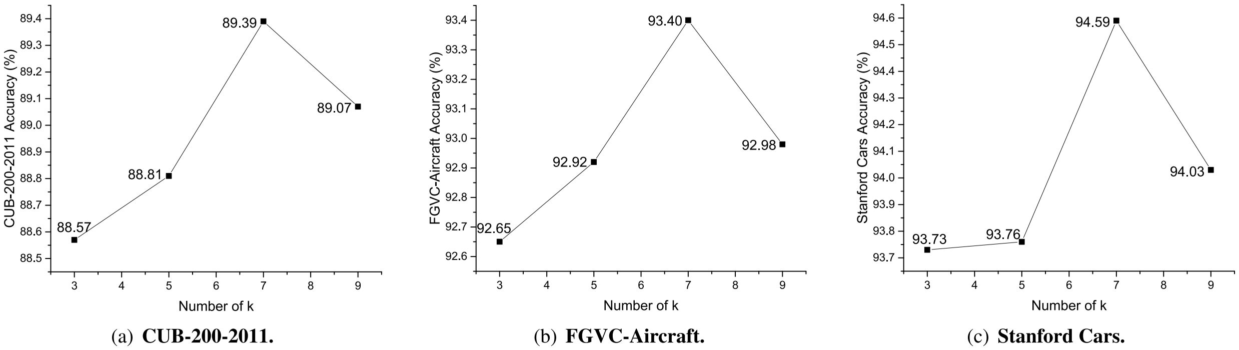
and evaluate on a test set as in previous work. In our experiments, we utilize EfficientNetV2-S [12] as the basic backbone. Firstly, we pre-train EfficientNetV2-S on 13M training images with 21,841 classes of ImageNet21k dataset [34], and then follow DeiT [35] and EfficientNetV2 [12] to fine-tune it on our datasets.

As illustrated in Table 6, our training setup used the EfficientNetV2: RMSProp optimizer with an attenuation of 0.9 and a momentum of 0.9; a batch standard momentum of 0.99; and a weight attenuation of 10^{-5}. With a total batch size of 16, each model was trained for 200 epochs. The learning rates for CUB, AIR, and CAR were first warmed up 3×10^{-4}, 3×10^{-3}, and 1×10^{-3}, respectively. Then, every 2.4 epochs, they declined by 0.97. We employed an exponential moving average (EMA) with a decay rate of 0.9999, as well as RandAugment [36], Random erase [37], Dropout [38]. Finally, the average of three runs determines the accuracy of transfer learning.

D. COMPARISONS WITH STATE-OF-THE-ART METHODS

To compare with other methods, we conducted extensive experiments on three fine-grained image datasets. For a fair comparison, we directly refer to the accuracy of their proposed method without any changes. Tables 7-9 describe the fine-grained image classification results for the CUB, AIR, and CAR datasets, respectively. The "Backbone" column indicates which CNNs were utilized as the backbone. Method, year, backbone, and accuracy are the columns in each table. In addition, all results are obtained without extra artificial labeling information (e.g., bounding box/part annotations).

As illustrated in Table 7, depth can be scaled by using more layers, such as VGG and ResNet. Specifically, MA-CNN (VGG-19) [15] is 0.67% higher than OPAM (VGG-16) [39]. The accuracy of MGE-CNN (ResNet-101) [16] outperformed HBPASM (ResNet-34) [43] and SMA-Net (ResNet-50) [49] by 2.60% and 1.69%, respectively. They generally increase



(a) **CUB-200-2011.**

(b) **FGVC-Aircraft.**

(c) **Stanford Cars.**

FIGURE 8. Results of our attention module for different *k* numbers on three datasets. Note that our attention module achieves the best accuracy at *k* = 7.

TABLE 8. Comparison results of our method (CMSEA) on the FGVC-Aircraft dataset.

Method	Year	Backbone	Accuracy(%)
Refined-CNN [40]	2017	VGG-16	87.7
HIHCA [52]	2017	VGG-16	88.3
RA-CNN [14]	2017	VGG-19	88.2
MA-CNN [15]	2017	VGG-19	89.9
Kernel-Pooling [41]	2017	ResNet-50	85.7
DFL-CNN [42]	2018	VGG-16	92.0
iSQRT-COV [11]	2018	ResNet-50	90.0
HBPASM [43]	2019	ResNet-34	91.3
DBTNet-50 [44]	2019	ResNet-50	91.2
Cross-X [45]	2019	ResNet-50	92.7
S3N [21]	2019	ResNet-50	92.8
DCL [17]	2019	ResNet-50	93.0
GHNS [51]	2021	VGG-16	93.0
MFF [48]	2021	ResNet-34	91.4
SSSNet [22]	2021	ResNet-50	93.3
MSEC [50]	2021	ResNet-50	93.4
CMSEA(Ours)	-	EfficientNetV2-S	**94.51**

TABLE 9. Comparison results of our method (CMSEA) on the Stanford Cars dataset.

Method	Year	Backbone	Accuracy(%)
OPAM [39]	2017	VGG-16	92.19
Kernel-Pooling [41]	2017	VGG-16	92.4
Refined-CNN [40]	2017	VGG-16	92.4
RA-CNN [14]	2017	VGG-19	92.5
MA-CNN [15]	2017	VGG-19	92.8
iSQRT-COV [11]	2018	ResNet-50	92.8
DFL-CNN [42]	2018	ResNet-50	93.1
MAMC [10]	2018	ResNet-101	93.0
TASN [20]	2019	VGG-19	93.2
HBPASM [43]	2019	ResNet-34	93.8
TASN [20]	2019	ResNet-50	93.8
MGE-CNN [16]	2019	ResNet-50	93.9
DBTNet-50 [44]	2019	ResNet-50	94.1
DCL [17]	2019	ResNet-50	94.5
Cross-X [45]	2019	ResNet-50	94.6
MS-SRP-D [46]	2020	ResNet-50	92.9
BBPL [47]	2021	ResNet	94.08
MFF [48]	2021	ResNet-34	93.4
SMA-Net [49]	2021	ResNet-50	94.37
GHNS [51]	2021	ResNet-50	94.54
SSSNet [22]	2021	ResNet-50	95.0
CMSEA(Ours)	-	EfficientNetV2-S	**95.19**

to scale only one of the three dimensions of the network in model scaling. Although they improve the accuracy, they commonly introduce a large number of network parameters and thus increase the model complexity. However, our method carefully balances the three dimensions of the network. Therefore, our CMSEA surpasses VGG-19 [15] and ResNet-101 [16] by 4.13% and 1.23%, respectively. The results demonstrate that our approach has state-of-the-art accuracy on three fine-grained image datasets.

TABLE 10. Results of our CECAMBConv module for different kernel size (*k*) numbers on three datasets.

k	CUB-200-2011	FGVC-Aircraft	Stanford Cars
3	88.57%	92.65%	93.73%
5	88.81%	92.92%	93.76%
7	**89.39%**	**93.40%**	**94.59%**
9	89.07%	92.98%	94.03%

TABLE 11. Impact of different attention modules on accuracy and FLOPs.

Attention	CUB-200-2011	FGVC-Aircraft	Stanford Cars	FLOPs
None	88.02%	91.72%	93.15%	10.921G
Original	88.31%	91.96%	93.30%	10.925G
Ours	**89.39%**(+1.1%)	**93.40%**(+1.4%)	**94.59%**(+1.3%)	10.921G

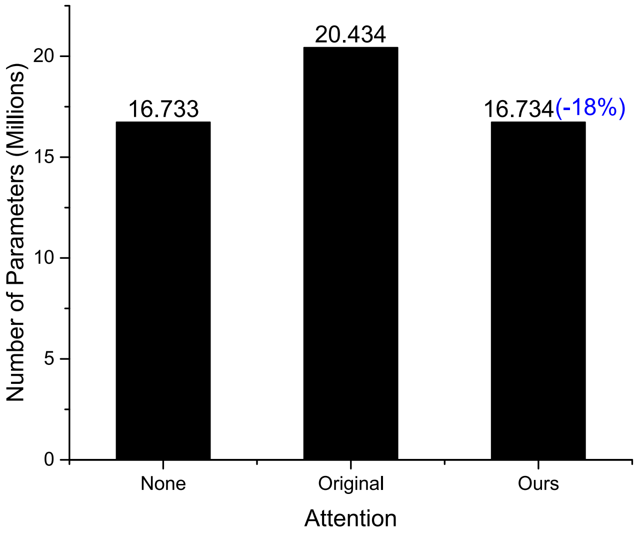
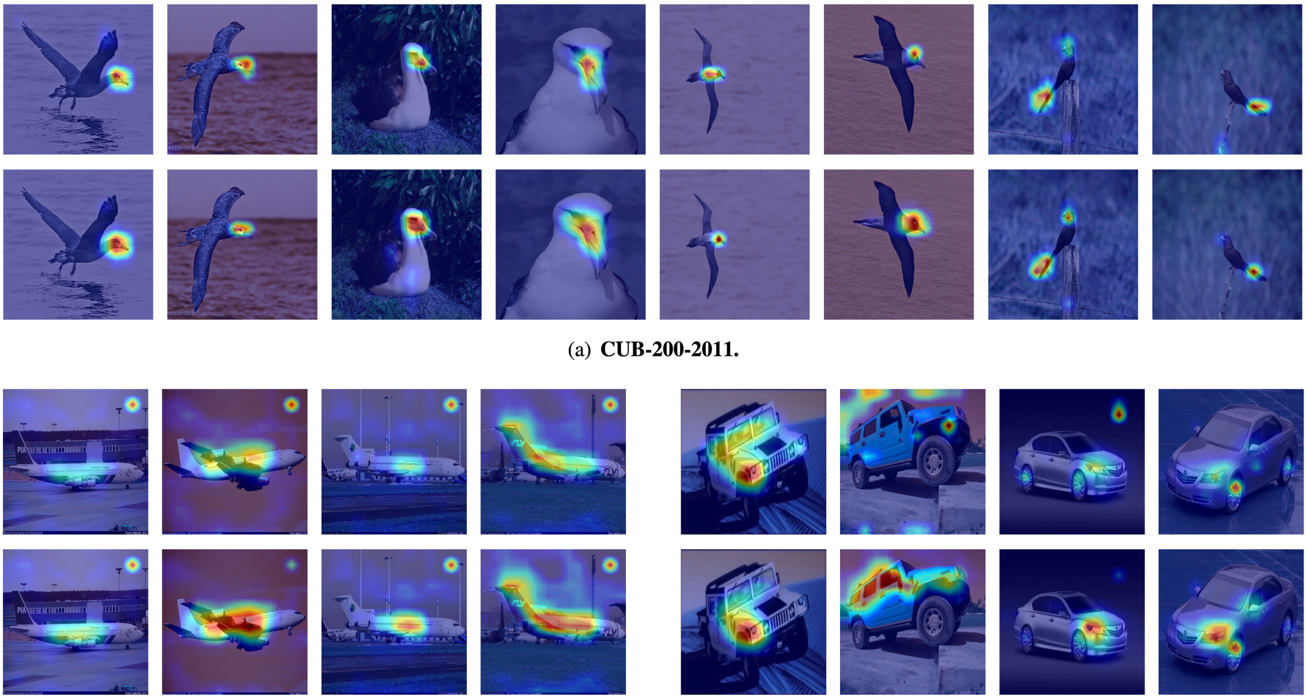


FIGURE 9. Parameter efficiency. Note that the network parameters of our method are reduced by 18% compared to the original method.

Consequently, our CMSEA achieves 90.63%, 94.51%, and 95.19% accuracy on CUB, AIR, and CAR datasets,

TABLE 12. Ablation study on training methods. The ✓ indicates that we use the corresponding method. EMA denotes exponential moving average.

ImageNet1k	ImageNet21k	RandAugment	Random erase	Dropout	EMA	CUB-200-2011	FGVC-Aircraft	Stanford Cars
✓						88.31%	91.96%	93.30%
	✓					88.66%	92.44%	93.81%
	✓	✓				89.06%	93.13%	93.99%
	✓		✓			88.85%	92.92%	93.96%
	✓			✓		89.14%	92.89%	94.07%
	✓				✓	89.69%	92.56%	94.17%
	✓	✓	✓	✓	✓	**89.87%**(+1.5%)	**93.55%**(+1.6%)	**94.75%**(+1.4%)


(a) **CUB-200-2011.**

(b) **FGVC-Aircraft.**

(c) **Stanford Cars.**

FIGURE 10. Visualization results. We randomly select one image from each subcategory of the three fine-grained datasets to Grad-CAM [53]. The first row is the attention map of the original method and the second row is the attention map of our CMSEA. The brighter an area is, the more important it is.

respectively. Moreover, Table 1 indicates that our method improves 2.3%, 2.5%, and 1.9% than the original method on CUB, AIR, and CAR datasets, respectively. Meanwhile, the network parameters of our approach are reduced by 18%. As a result, our method outperforms the previous most competitive methods on three fine-grained image datasets.

E. ABLATION STUDY

1) EFFECT OF KERNEL SIZE (k) ON ATTENTION MODULE

As presented in Eq. (17), our CECAMBConv module involves a parameter k, the kernel size of the 1D convolution. We evaluated its impact on our CECAMBConv module. We utilized EfficientNetV2-S as the backbone and trained it using our CECAMBConv module with k values ranging from 3 to 9. Table 10 and Figure 8 show the results of the study, and our CECAMBConv module achieves the best results at $k = 7$.

2) EFFECT OF ATTENTION MODULE ON THE NETWORK

In this part, we discuss the impact of different attention modules on the network. As presented in Table 11, the accuracy of the original approach improved by 0.29%, 0.24%, and 0.15% over the no-attention approach on CUB, AIR, and CAR datasets, respectively. Specifically, our method surpassed the original method by 1.1%, 1.4%, and 1.3% on CUB, AIR, and CAR datasets, respectively. Therefore, the attention mechanism can improve the model performance. As presented in Figure 9, our method significantly surpasses the original method in terms of network parameters. As a result, our CECAMBConv module verifies that avoiding dimensionality reduction has a positive effect on learning channel attention.

3) EFFECTIVENESS OF REGULARIZATION AND DATA AUGMENTATION

In this part, we discuss training strategies to learn CMSEA in a data-efficient way. We build upon PyTorch and the timm library. We analyze the impact of each choice. Table 12 indicates the hyperparameters we use by default for training in all experiments unless otherwise stated. Experiments confirm that our CMSEA requires strong data augmentation, such as RandAugment [36] and Random erase [37]. Almost all of the data augmentation methods we evaluated proved to be useful. Regularization like Dropout [38] improves performance. Dropout is a network-level regularization that reduces co-adaptation by randomly dropping channels. Meanwhile, we evaluate some enhancements in the network Exponential Moving Average (EMA) obtained after training.

F. VISUALIZATION RESULTS

To more intuitively show the sensitivity of our CMSEA to subtler parts. We show the visualization results of our CMSEA on fine-grained image datasets in Figure 10. We randomly sample sixteen images from three fine-grained datasets as observation objects. Through Grad-CAM [53] visualization technology to show that the attention part of different regions. The lighter a region is, the more important it is. Consequently, our CMSEA can facilitate the model to learn more detailed features of the object.

V. CONCLUSION

This paper presents a Compound Model Scaling with Efficient Attention (CMSEA) for fine-grained image classification. Specifically, our CMSEA carefully balances each dimension of network width, depth, and image resolution in model scaling. Moreover, our attention module replaces the two fully connected layers with a fast one-dimensional convolution with the kernel size (k), which avoids dimensionality reduction and effectively learns the discriminative features. In addition, regularization and data augmentation were employed to improve accuracy in the training. Experimental results demonstrate that our CMSEA achieves 90.63%, 94.51%, and 95.19% accuracy on CUB-200-2011, FGVC-Aircraft, and Stanford Cars datasets, respectively. In particular, our CMSEA on CUB-200-2011 obtains 2.3% higher accuracy with 18% fewer network parameters than the original approach. Consequently, our method has better accuracy and parameter efficiency compared to most existing methods. We will investigate better attention mechanism modules to improve the model performance in the future.

REFERENCES

[1] X.-S. Wei, J. Wu, and Q. Cui, "Deep learning for fine-grained image analysis: A survey," 2019, *arXiv:1907.03069*.

[2] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The Caltech-UCSD birds-200-2011 dataset," California Inst. Technol., Pasadena, CA, USA, Tech. Rep. CNS-TR-2011-001, 2011. [Online]. Available: http://www.vision.caltech.edu/visipedia/CUB-200-2011.html

[3] S. Maji, E. Rahtu, J. Kannala, M. Blaschko, and A. Vedaldi, "Fine-grained visual classification of aircraft," 2013, *arXiv:1306.5151*. [Online]. Available: https://www.robots.ox.ac.uk/~vgg/data/fgvc-aircraft/

[4] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3D object representations for fine-grained categorization," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, Dec. 2013, pp. 554–561.

[5] H. Zhang, T. Xu, M. Elhoseiny, X. Huang, S. Zhang, A. Elgammal, and D. Metaxas, "SPDA-CNN: Unifying semantic—Part detection and abstraction for fine-grained recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1143–1152.

[6] Z. Wang, S. Wang, P. Zhang, H. Li, W. Zhong, and J. Li, "Weakly supervised fine-grained image classification via correlation-guided discriminative learning," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 1851–1860.

[7] Z. Wang, S. Wang, S. Yang, H. Li, J. Li, and Z. Li, "Weakly supervised fine-grained image classification via Guassian mixture model oriented discriminative learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9749–9758.

[8] Z. Wang, S. Wang, H. Li, Z. Dou, and J. Li, "Graph-propagation based correlation learning for weakly supervised fine-grained image classification," in *Proc. AAAI*, vol. 34, no. 7, 2020, pp. 12289–12296.

[9] A. Recasens, P. Kellnhofer, S. Stent, W. Matusik, and A. Torralba, "Learning to zoom: A saliency-based sampling layer for neural networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 51–66.

[10] M. Sun, Y. Yuan, F. Zhou, and E. Ding, "Multi-attention multi-class constraint for fine-grained image recognition," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 805–821.

[11] P. Li, J. Xie, Q. Wang, and Z. Gao, "Towards faster training of global covariance pooling networks by iterative matrix square root normalization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 947–955.

[12] M. Tan and Q. V. Le, "EfficientNetV2: Smaller models and faster training," 2021, *arXiv:2104.00298*.

[13] X. He, Y. Peng, and J. Zhao, "Which and how many regions to gaze: Focus discriminative regions for fine-grained visual categorization," *Int. J. Comput. Vis.*, vol. 127, no. 9, pp. 1235–1255, Sep. 2019.

[14] J. Fu, H. Zheng, and T. Mei, "Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4438–4446.

[15] H. Zheng, J. Fu, T. Mei, and J. Luo, "Learning multi-attention convolutional neural network for fine-grained image recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5209–5217.

[16] L. Zhang, S. Huang, W. Liu, and D. Tao, "Learning a mixture of granularity-specific experts for fine-grained categorization," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8331–8340.

[17] Y. Chen, Y. Bai, W. Zhang, and T. Mei, "Destruction and construction learning for fine-grained image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5157–5166.

[18] J. Zhao, Y. Peng, and X. He, "Attribute hierarchy based multi-task learning for fine-grained image classification," *Neurocomputing*, vol. 395, pp. 150–159, Jun. 2020.

[19] T. Kim, H. Kim, and H. Byun, "Localization-aware adaptive pairwise margin loss for fine-grained image recognition," *IEEE Access*, vol. 9, pp. 8786–8796, 2021.

[20] H. Zheng, J. Fu, Z.-J. Zha, and J. Luo, "Looking for the devil in the details: Learning trilinear attention sampling network for fine-grained image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5012–5021.

[21] Y. Ding, Y. Zhou, Y. Zhu, Q. Ye, and J. Jiao, "Selective sparse sampling for fine-grained image recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6599–6608.

[22] S. Rong, Z. Wang, and J. Wang, "Separated smooth sampling for fine-grained image classification," *Neurocomputing*, vol. 461, pp. 350–359, Oct. 2021.

[23] W. Dai, W. Diao, X. Sun, Y. Zhang, L. Zhao, J. Li, and K. Fu, "CAMV: Class activation mapping value towards open set fine-grained recognition," *IEEE Access*, vol. 9, pp. 8167–8177, 2021.

[24] H. Wei, M. Zhu, B. Wang, J. Wang, and D. Sun, "Two-level progressive attention convolutional network for fine-grained image recognition," *IEEE Access*, vol. 8, pp. 104985–104995, 2020.

[25] O. T. Nartey, G. Yang, J. Wu, and S. K. Asare, "Semi-supervised learning for fine-grained classification with self-training," *IEEE Access*, vol. 8, pp. 2109–2121, 2020.

[26] X. He, Y. Peng, and J. Zhao, "Fast fine-grained image classification via weakly supervised discriminative localization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 5, pp. 1394–1407, May 2019.

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

[28] S. Zagoruyko and N. Komodakis, ''Wide residual networks,'' 2016, *arXiv:1605.07146*.

[29] Y. Huang, Y. Cheng, A. Bapna, O. Firat, D. Chen, M. Chen, H. Lee, J. Ngiam, Q. V. Le, and Y. Wu, ''GPipe: Efficient training of giant neural networks using pipeline parallelism,'' in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 103–112.

[30] M. Tan and Q. Le, ''EfficientNet: Rethinking model scaling for convolutional neural networks,'' in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6105–6114.

[31] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, ''MobileNetV2: Inverted residuals and linear bottlenecks,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.

[32] J. Hu, L. Shen, and G. Sun, ''Squeeze-and-excitation networks,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[33] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, ''ECA-Net: Efficient channel attention for deep convolutional neural networks,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2235–2239.

[34] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, ''ImageNet large scale visual recognition challenge,'' *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[35] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, ''Training data-efficient image transformers & distillation through attention,'' in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 10347–10357.

[36] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le, ''Randaugment: Practical automated data augmentation with a reduced search space,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2020, pp. 702–703.

[37] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, ''Random erasing data augmentation,'' in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 7, 2020, pp. 13001–13008.

[38] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, ''Dropout: A simple way to prevent neural networks from overfitting,'' *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, Jun. 2014.

[39] Y. Peng, X. He, and J. Zhao, ''Object—Part attention model for fine-grained image classification,'' *IEEE Trans. Image Process.*, vol. 27, no. 3, pp. 1487–1500, Mar. 2018.

[40] W. Zhang, J. Yan, W. Shi, T. Feng, and D. Deng, ''Refining deep convolutional features for improving fine-grained image recognition,'' *EURASIP J. Image Video Process.*, vol. 2017, no. 1, pp. 1–10, Dec. 2017.

[41] Y. Cui, F. Zhou, J. Wang, X. Liu, Y. Lin, and S. Belongie, ''Kernel pooling for convolutional neural networks,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2921–2930.

[42] Y. Wang, V. I. Morariu, and L. S. Davis, ''Learning a discriminative filter bank within a CNN for fine-grained recognition,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4148–4157.

[43] M. Tan, G. Wang, J. Zhou, Z. Peng, and M. Zheng, ''Fine-grained classification via hierarchical bilinear pooling with aggregated slack mask,'' *IEEE Access*, vol. 7, pp. 117944–117953, 2019.

[44] H. Zheng, J. Fu, Z.-J. Zha, and J. Luo, ''Learning deep bilinear transformation for fine-grained image representation,'' 2019, *arXiv:1911.03621*.

[45] W. Luo, X. Yang, X. Mo, Y. Lu, L. Davis, J. Li, J. Yang, and S.-N. Lim, ''Cross-X learning for fine-grained visual categorization,'' in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8242–8251.

[46] M. Luo, G. Wen, Y. Hu, D. Dai, and Y. Xu, ''Stochastic region pooling: Make attention more expressive,'' *Neurocomputing*, vol. 409, pp. 119–130, Oct. 2020.

[47] J. Chen, J. Hu, and S. Li, ''Learning to locate for fine-grained image recognition,'' *Comput. Vis. Image Understand.*, vol. 206, May 2021, Art. no. 103184.

[48] L. Wang, K. He, X. Feng, and X. Ma, ''Multilayer feature fusion with parallel convolutional block for fine-grained image classification,'' *Appl. Intell.*, vol. 52, pp. 2872–2883, 2022, doi: 10.1007/s10489-021-02573-2.

[49] C. Liu, L. Huang, Z. Wei, and W. Zhang, ''Subtler mixed attention network on fine-grained image classification,'' *Appl. Intell.*, vol. 51, no. 11, pp. 7903–7916, 2021.

[50] Y. Zhang, Y. Sun, N. Wang, Z. Gao, F. Chen, C. Wang, and J. Tang, ''MSEC: Multi-scale erasure and confusion for fine-grained image classification,'' *Neurocomputing*, vol. 449, pp. 1–14, Aug. 2021.

[51] T. Kim, K. Hong, and H. Byun, ''The feature generator of hard negative samples for fine-grained image recognition,'' *Neurocomputing*, vol. 439, pp. 374–382, Jun. 2021.

[52] S. Cai, W. Zuo, and L. Zhang, ''Higher-order integration of hierarchical convolutional activations for fine-grained visual categorization,'' in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 511–520.

[53] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, ''Grad-CAM: Visual explanations from deep networks via gradient-based localization,'' in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 618–626.

JINZHENG GUANG was born in Guangxi, China, in 1995. He received the bachelor's degree in automation from the Shanghai University of Engineering Science, in 2018, where he is currently pursuing the master's degree. His research interests include digital image processing and computer vision.

JIANRU LIANG was born in Shanghai, China, in 1961. He received the bachelor's degree in electrical engineering from Shanghai University, in 1984. He is currently a Senior Engineer and a Master's Supervisor with the Shanghai University of Engineering Science. His research interests include intelligent detection and control.