IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 23, 2022, accepted March 20, 2022, date of publication March 30, 2022, date of current version April 7, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3163302

Weakly Supervised Semantic and Attentive Data Mixing Augmentation for Fine-Grained Visual Categorization

MENGQI HE[1], QILONG CHENG[2], AND GUANQIU QI[3]
[1]College of Engineering and Computer Science, The Australian National University, Canberra, ACT 2600, Australia
[2]Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, ON M5S 1A1, Canada
[3]Computer Information Systems Department, The State University of New York at Buffalo State, Buffalo, NY 14222, USA

Corresponding author: Guanqiu Qi (qig@buffalostate.edu)

ABSTRACT As a key factor, the availability of large-scale training samples determines the improvement of visual performance. However, the size of Fine-Grained Visual Categorization (FGVC) datasets is always limited. Therefore, overfitting as an issue in FGVC-related training needs to be solved. Data mixing augmentation is a widely-used data augmentation method. In most of the recently proposed data mixing augmentation methods, random patch selection may generate meaningless training samples and result in model instability during the training process. This paper proposes a data mixing augmentation strategy termed Semantic and Attentive Data Mixing (SADMix) to select semantic patches for the generation of new training samples. In SADMix, a certain number of critical regions are localized according to convolutional activations. An image patch is selected from these localized regions for the generation of new training samples. The size, aspect ratio, and center location of these image patches are changed according to the random values from a beta distribution. These image patches with semantic information are used to mix two training images. According to the class activation map (CAM), training images and their labels are mixed proportionally to generate new mixed training samples and the corresponding labels. The proposed SADMix is tested on three fine-grained datasets, which are CUB-200-2011, FGVC Aircraft, and Stanford Cars, respectively. The experimental results confirm the effectiveness of the proposed SADMix.

INDEX TERMS Data augmentation, fine-grained visual categorization, attention.

I. INTRODUCTION

In order to prevent overfitting caused by the limited training data, random data augmentation methods used in machine learning are applied to increase the size and diversity of training data. Additionally, random data augmentation methods can also improve the generalization ability of models. Therefore, data augmentation methods, such as random image cropping, flipping and rotation, are widely used in the training phase of visual tasks such as object recognition and object detection, etc. Recently proposed data augmentation methods are categorized into two groups. As the first group, region-erasing methods [1], [2] erase partial image regions to encourage models to identify more discriminative regions. As the second group, data mixing methods [2]–[7] generate new training data by combining multiple images and fusing their labels accordingly. Data mixing augmentation methods have been proven to be effective in both general and fine-grained image classification [6]–[8]. Therefore, they are receiving more and more attention. In Mixup [5], two images are combined linearly and their labels are mixed by using the same combination coefficients. In CutMix [7], an image region is first cut out and then pasted on another image. Labels of these two images are mixed according to the corresponding area proportion.

Data mixing augmentation methods have two major drawbacks. As the first drawback, the diversity of augmented data is limited due to the symmetrically blended image regions. As a result, the selected regions are restricted to be complementary. As the second drawback, label noise may be generated by the random selection of image regions for mixing. In Fig. 1, the randomly generated region may cover the meaningless region in image (a), and this region is copied and pasted on the critical regions in image (b) to generate

The associate editor coordinating the review of this manuscript and approving it for publication was Tai-Hoon Kim.

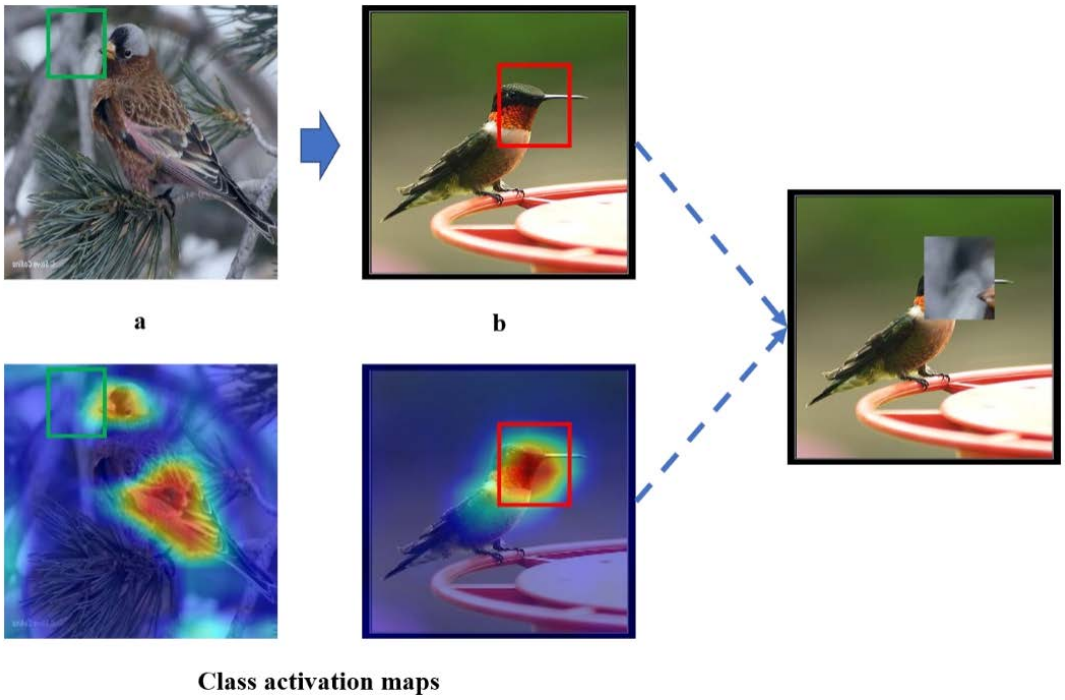

FIGURE 1. Meaningless training samples generated in SnapMix.

a meaningless training sample. Moreover, the meaningless region in image (a) may still have some semantic information in the class activation map (CAM). Due to the subtle differences in some small image regions, these two drawbacks limit the recognition performance, especially in the field of Fine-Grained Visual Categorization (FGVC).

Due to both high intra-class variances and low inter-class variances, FGVC is a challenging visual task. Additionally, the size of fine-grained datasets is usually small because the obtainment of partial manual annotations requires expert-level domain knowledge. In essence, FGVC mainly focuses on learning fine-grained features from the limited data. As discussed in [9], FGVC feature learning methods can be divided into three paradigms, fine-grained object recognition (1) with localization-classification subnetwork, (2) with end-to-end feature encoding, and (3) with external information. Due to the limited dataset size, data augmentation is also important for FGVC.

Some existing study focuses on data augmentation for FGVC [6], [9], [10]. A data mixing augmentation method called Semantically Proportional Mixing (SnapMix) was proposed [6]. Mixed labels are generated by exploiting Class Activation Map (CAM) to reduce negative influence [10]. The generated labels, that are normalized to sum to 1, are used to weigh the mixed images in SnapMix. Different from CutMix, cut-and-paste operations are set to be asymmetric in SnapMix, which can boost data diversity. Due to the importance of attribute-level features in discriminating subcategories, the discriminative and transferable attribute features are explored and used to scale up the fine-grained training samples in Attribute Mix [8]. A weakly supervised data augmentation network (WS-DAN) was proposed to generate attention maps for the representation of discriminative regions by weakly-supervised learning [10]. The generated attention maps were then used to guide image augmentation, such as attention cropping and attention dropping. The fine-grained features were extracted through an effective module called bilinear attention pooling (BAP).

This paper proposes a data mixing augmentation method called Semantic and Attention Data Mixing (SADMix) for FGVC. In SADMix, the regions selected for mixing are localized by using CNN's activation maps. According to the observation of CNN's activation maps, the regions with high activation values often correspond to the localized key parts. When the activation value is getting larger, more information is involved in the representation of the corresponding region. Since the regions with relatively large activation values are often adjacent to the region with the largest activation value, non-maximum suppression (NMS) is first adopted to select a fixed number of box regions with different scales. Then a box region is randomly selected from these box regions for mixing. In order to improve the corresponding performance, random adjustment is applied to change the size and aspect ratio of the selected box region according to a random value generated by Beta distribution. According to the conclusion in [11], the regions with relatively large activation values are often adjacent to the region with the largest activation value. Therefore, non-maximum suppression (NMS) is first adopted to select a fixed number of box regions with different scales and less redundancy.

The main contributions of this paper are summarized as follows: (1) A data mixing augmentation method called SADMix is proposed for fine-grained visual categorization. SADMix can mix images asymmetrically and generate the corresponding label of the mixed image according to the normalized CAM. A box region used for mixing is selected by a Semantic and Attentive Part Localization (SAPL) module. The box region for image mixing is always located in critical object parts. (2) Due to the fixed size of box regions selected from a SAPL module, it is necessary to randomly change the size and aspect ratio of the selected semantic and attentive box region for integrating more useful information. Experimental results confirm that even a simple network with the proposed SADMix can achieve competitive performance compared with the state-of-the-art methods.

II. RELATED WORK

This section briefly reviews FGVC methods and several recently proposed data augmentation methods, which are closely related to the proposed SADMix.

A. FGVC METHODS

This paper only focuses on the weakly supervised FGVC methods with localization-classification subnetwork [11]–[16]. Weakly supervised FGVC methods can localize the critical regions and learn fine-grained features only depending on the image-level labels. Navigator-Teacher-Scrutinizer Network (NTS-Net) was proposed and its localization subnetwork was trained to localize informative regions without any manual annotations [11]. The features extracted from the whole image and informative regions were concatenated for recognition. In multi-branch and multi-scale learning network (MMAL-Net) [11], three branches are used to learn features from the whole input image, an object and parts. Localization branches in MMAL-Net only involve a small number of parameters that need to be trained. A discriminative filter learning network (DFL-Net) was proposed [17] to learn discriminative mid-level patches in an end-to-end fashion. Both discriminative local and global features were learnt by DFL-Net. Destruction and construction learning (DCL) [18] encourages the CNN to learn fine-grained features by the destruction of the global image structure. WS-DAN [19] first generates attention maps to represent the discriminative parts by weakly-supervised learning. The generated attention maps are then used to guide image augmentation, such as attention cropping and attention dropping. The fine-grained features are extracted by an effective module called bilinear attention pooling (BAP).

B. DATA MIXING AUGMENTATION

In this subsection, the recently published data mixing augmentation methods are introduced. Mixup [5] was proposed to mix data to extend the training distribution. It generated images by linearly combining training images and fusing their labels with the same coefficients. CutMix [7] can generate a mixed image by cutting out one region and pasting it on another image. The labels are also mixed proportionally according to the area of the corresponding regions. To avoid the label noise generated by Mixup and CutMix and increase the diversity of training data, SnapMix [6] mixes images asymmetrically and generates the new labels for the mixed images by estimating semantic compositions according to the normalized CAM. The details of these related data augmentation methods will be discussed in subsection 3.1.

III. THE PROPOSED METHOD

This section discusses the proposed SADMix in detail. The subsection 3.1 reviews Mixup, CutMix and SnapMix. The subsection 3.2 specifies the details of the proposed SADMix.

A. BACKGROUND

The original training dataset is defined as $\{(\mathbf{I}_i, y_i), i = 1, 2, \ldots, N\}$, where $\mathbf{I}_i \in \mathbf{R}^{3\times H\times W}$ and y_i refer to an image and its label, respectively. For a data pair $((\mathbf{I}_a, y_a), (\mathbf{I}_b, y_b))$ and a random variable λ from Beta distribution $Beta(\alpha, \alpha)$, a mixed image $\hat{\mathbf{I}}$ and two label weights β_a and β_b are generated. The label weights β_a and β_b are used to generate the mixed label $\hat{y}$.

In Mixup [5], two images and their labels are combined linearly. The linear combination is expressed as follows.

$$\hat{\mathbf{I}} = \lambda \mathbf{I}_a + (1-\lambda)\mathbf{I}_b \tag{1}$$

$$\hat{y} = \beta_a y_a + \beta_b y_b \tag{2}$$

$$\beta_a = \lambda \tag{3}$$

$$\beta_b = 1 - \lambda \tag{4}$$

In Mixup, the whole image is applied to the linear combination. So, the robustness of CNN to adversarial examples is improved.

Different from Mixup, CutMix adopts cut-and-paste operations for mixing two images. The mixed labels are combined according to the area ratio. The combination in CutMix can be expressed as follows.

$$\hat{\mathbf{I}} = M_\lambda \bigodot \mathbf{I}_a + (1 - M_\lambda) \bigodot \mathbf{I}_b \tag{5}$$

$$\hat{y} = \lambda y_a + (1-\lambda) y_b \tag{6}$$

where $\bigodot$ denotes the element-wise multiplication and $M_\lambda \in \mathbf{R}^{H\times W}$ is the binary mask of a randomly selected box region whose area ratio to the image is λ. In addition to improving the robustness of CNN, CutMix can enhance the localization capacity of CNN.

Area ratio used in CutMix cannot effectively reflect the intrinsic semantic composition of the mixed image. A large area may contain less semantic information, but it takes a large proportion in the label mixing process. This results in noise mixing labels. In order to solve this problem, class activation maps (CAM) [10] is applied to estimate the label composition of mixed images in SnapMix. Different from mixing images at symmetric locations in CutMix, an area is first cropped at a randomly selected location and then transformed and pasted to a randomly selected location on

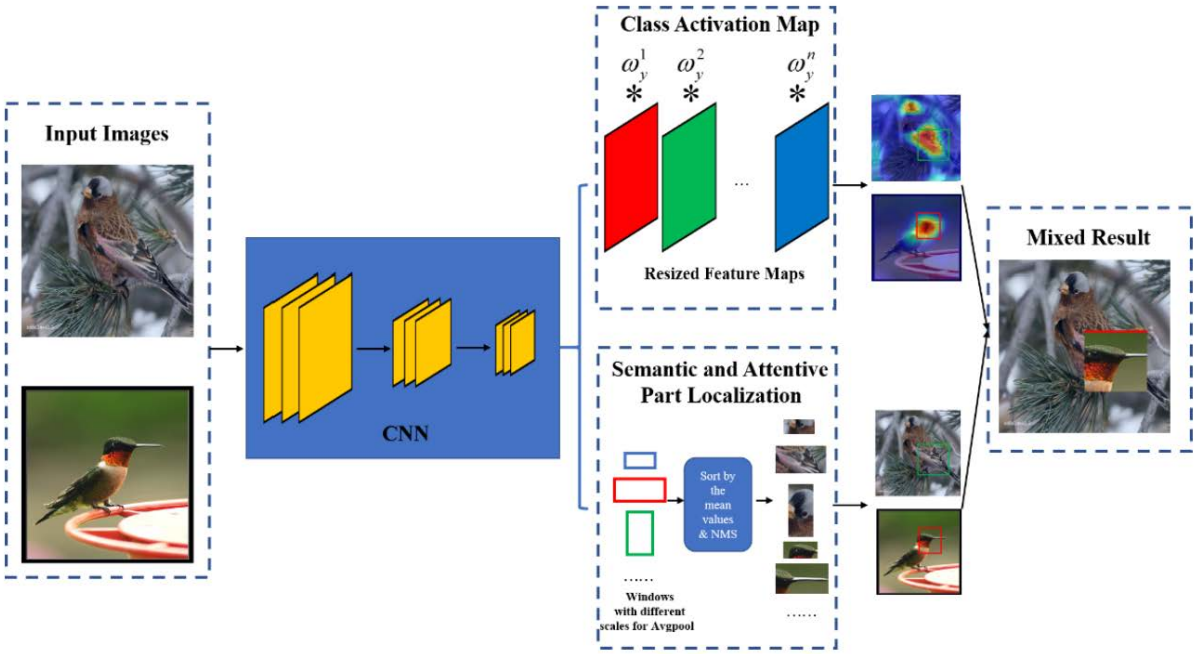

FIGURE 2. An overview of SADMix.

another image in SnapMix. The mixing of two images can be expressed as follows.

$$\hat{\mathbf{I}} = T_\theta(M_{\lambda_a} \bigodot \mathbf{I}_a) + (1 - M_{\lambda_b}) \bigodot \mathbf{I}_b \tag{7}$$

where M_{λ_a} and M_{λ_b} are two binary masks containing the randomly selected box regions with area ratios λ_a and λ_b, and T_θ is a transformation that makes the cutout region size of $\mathbf{I}_a$ to match the corresponding box region size of $\mathbf{I}_b$. In the process of label generation, the CAM of the input image is calculated as follows.

$$Cam(\mathbf{I}_i) = Upsample(\sum_{j=1}^{C} \omega_{y_i}^j F_j(\mathbf{I_i})) \tag{8}$$

where $\mathbf{I}_i$ is the input image, $F(\mathbf{I_i}) \in \mathbf{R}^{c\times h\times w}$ is the output of the last convolutional layer, $F_j(\mathbf{I_i}) \in \mathbf{R}^{h\times w}$ is the j-th feature map of $F(\mathbf{I_i})$, and $\omega_{y_i}^j \in \mathbf{R}^c$ is the weight corresponding to class y_i in the FC layer. The CAM of the input image should be normalized to sum-to-1. The normalized CAM of the input image $\mathbf{I}_i$, which is defined as $N(\mathbf{I}_i)$, is calculated as follows.

$$N(\mathbf{I}_i) = \frac{Cam(\mathbf{I}_i)}{sum(Cam(\mathbf{I}_i)))} \tag{9}$$

The mixed label in SnapMix is generated as follows.

$$\hat{y} = \beta_a y_a + \beta_b y_b \tag{10}$$

$$\beta_a = sum(M_{\lambda_a} \bigodot N(\mathbf{I}_a)) \tag{11}$$

$$\beta_b = 1 - sum(M_{\lambda_b} \bigodot N(\mathbf{I}_b)) \tag{12}$$

In SnapMix, two images are mixed asymmetrically to generate a mixed image and the target labels of the mixed image are generated by estimating its intrinsic compositions through the normalized CAM. Experimental results confirm SnapMix can consistently outperform existing data mixing augmentation methods such as Cutout, Mixup and CutMix.

B. SADMix

Different from SnapMix, the random box region generation is replaced by Semantic and Attentive Part Localization (SAPL) module in SADMix.

According to the observation of the feature map $\mathbf{F} \in \mathbf{R}^{c\times h\times w}$ of the last convolutional layer in CNN, where c is the channel of a feature map, h and w ere the height and width of the feature map, windows with different sizes and aspect ratios are used to perform average pooling operation on the feature map, which is similar to [20]. The activations mean value of each window's feature map $\mathbf{F}_\omega$ is calculate as follows.

$$\hat{f}_\omega = \frac{\sum_{x=0}^{W_\omega - 1} \sum_{y=0}^{H_\omega - 1} \mathbf{F}_\omega(x, y)}{H_\omega \times W_\omega} \tag{13}$$

where H_ω and W_ω are the height and width of a window's feature map. The activations mean maps are summed in channel dimension. When the $\hat{f}_\omega$ is getting larger, the region corresponding to this mean value becomes more informative. The regions with relatively high informativeness are selected for data mixing. However, they are usually adjacent to the largest $\hat{f}_\omega$ windows and almost contain the same part of the object. To reduce region redundancy, non-maximum suppression (NMS) is used to select a number of regions with different scales.

The overall process of SADMix is shown in Fig. 2 Two original training images $((\mathbf{I}_a, y_a), (\mathbf{I}_b, y_b))$ are processed through the backbone network. The feature maps of the last convolutional layer in the backbone network are applied to both CAM and SAPL. The normalized CAMs $N(\mathbf{I}_a)$ and $N(\mathbf{I}_b)$ of training images are calculated through the methods reviewed in subsection 3.1. The image mixing is expressed as follows.

$$\hat{\mathbf{I}} = T_\theta(M_{\lambda_a}^{SAPL} \bigodot \mathbf{I}_a) + (1 - M_{\lambda_b}^{SAPL}) \bigodot \mathbf{I}_b \qquad (14)$$

where $M_{\lambda_a}^{SAPL}$ and $M_{\lambda_b}^{SAPL}$ are two binary masks containing semantic and attentive box regions with the area ratios λ_a and λ_b. In order to improve the performance, the aspect ratio and size of the semantic and attentive box regions are determined by a random value between 0 and 1 that also follows a Beta distribution.

IV. EXPERIMENTS AND RESULTS

In this section, the performance of the proposed SADMix is evaluated on three fine-grained datasets. Two network architectures (ResNet34 and ResNet50) are used as backbones. The implementation details of baselines and SADMix are introduced in subsection 4.3. The performance comparison between the proposed SADMix and the related data augmentation methods is shown in subsection 4.4. In order to further show the effectiveness of SADMix, the class activation maps (CAMs) of all the methods are compared and analyzed in subsection 4.5.

A. DATASETS

In comparative experiments, three fine-grained datasets are used for evaluation.

- CUB-200-2011 [21] is the most widely used fine-grained object dataset, which contains 11,788 images spanning 200 bird species. There are 5,994 training images and 5,794 testing images, respectively.
- Stanford-Cars [22] contains 16,185 images involving 196 manufacturer classes. There are 8,144 training images and 8,041 testing images, respectively.
- FGVC-Aircraft [23] consists of 10,000 images, which can be divided into 100, 30 and 70 categories by Variants, Manufacturers and Family, respectively. There are 6,667 training images and 3,333 testing images, respectively. The comparative experiments only use the images from 100 categories of variants.

B. BACKBONE NETWORKS AND BASELINES

Two backbone networks of ResNet34 and ResNet50 pretrained on ImageNet dataset [24] are applied to the proposed SADMix and other four representative data augmentation methods including CutOut [2], Mixup [5], CutMix [7] and SnapMix [6]. These two networks pretrained on ImageNet are also used as baselines in the experiments. Similar to [6], a strong baseline termed as mid-level baseline that incorporates mid-level features is used. There are two branches in the mid-level baseline. One is the original classification branch in baseline. The other is mid-level classification branch, which is composed of $Conv_{1\times1}$, MaxPooling and FC layer. The mid-level classification branch is placed after the last convolutional layer in the network. In the inference phase, the sum of the outputs of original classification branch and mid-level classification branch is used for prediction.

1) DATA AUGMENTATION METHODS

According to the experiment setup in [6], the probability of performing data augmentation is set to 0.5 for CutOut and Mixup and 1.0 for CutMix, SnapMix and SADMix. The α in Beta distribution $Beta(\alpha, \alpha)$ is set to 1.0 for MixUp and 3.0 for the remaining methods.

2) FGVC METHODS

In order to further evaluate the performance of SADMix, five state-of-the-art FGVC methods including NTS-Net [12], MMAL-Net [11], DFL-CNN [17], DCL [18] and WS-DAN [19] are used for comparison in the experiments.

C. IMPLEMENTATION DETAILS OF SADMIX

In this subsection, the implementation of SADMix is discussed in detail. In the experiments, the input images are first resized to 512 × 512 and then randomly cropped to 448 × 448. Similar to the experiments setup in [11], windows with three broad scale categories are constructed: [4 × 4, 3 × 5, 5 × 3], [6 × 6, 5 × 7, 7 × 5], [8 × 8, 6 × 10, 10 × 6, 7 × 9, 9 × 7, 7 × 10, 10 × 7] for the feature maps of 14 × 14 output by the last convolutional layer in ResNet. After using the NMS, different number of windows need to be selected on each broad scale category. The selection of window sizes is consistent with that of [1], which presents superior results in our experiments, indicating that this selection can obtain the regions with good discriminability. Avg. pooing operation is required for each window size, and multiple feature maps obtained will be sorted from large to small values. Also, NMS is used to select the region with the largest mean value. In the experiments, $N_1 = 2$, $N_2 = 3$ and $N_3 = 2$ are set. One window is randomly selected from these windows as the final semantic and attentive box region. In order to improve the performance, the selected semantic and attentive box regions are then randomly adjusted in size and aspect ratio according to a randomly generated value.

The experiments are implemented by PyTorch and trained on a PC with four TITAN-X GPUs. Similar to [6], stochastic gradient descent (SGD) with momentum 0.9 is used, and learning rate is set to 0.001 for the pre-trained parameters and 0.01 for new parameters respectively. All models are trained for 200 epochs and the learning rate is decayed by factor 0.1 every 80 epochs.

D. PERFORMANCE EVALUATION

In this subsection, the results of SADMix and performance comparisons with comparative approaches are presented. In order to show the effectiveness of random adjustment

TABLE 1. Performance comparison of SADMix with/without random adjustment.

	CUB-200-2011		Stanford-Cars		FGVC-Aircraft	
	ResNet34	ResNet50	ResNet34	ResNet50	ResNet34	ResNet50
SADMix	86.96%	88.01%	93.81%	94.25%	92.55%	92.96%
SADMix + rand adjustment	87.37%	88.23%	94.08%	94.43%	92.92%	93.13%

TABLE 2. Performance comparison of data augmentation methods on fine-grained object datasets.

	CUB-200-2011		Stanford-Cars		FGVC-Aircraft	
	ResNet34	ResNet50	ResNet34	ResNet50	ResNet34	ResNet50
Baseline	84.98%	85.49%	92.02%	93.04%	89.92%	91.07%
CutOut	83.36%	83.55%	92.84%	93.76%	89.90%	91.23%
MixUp	85.22%	86.23%	93.28%	93.96%	91.02%	92.24%
CutMix	85.69%	86.15%	93.61%	94.18%	91.26%	92.23%
SnapMix	87.06%	87.75%	93.95%	94.30%	92.36%	92.80%
SADMix	87.37%	88.23%	94.08%	94.43%	92.92%	93.13%

TABLE 3. Performance comparison with the state-of-the-art FGVC methods on fine-grained object datasets.

Method	**Backbone**	**CUB-200-2011**	**Stanford-Cars**	**FGVC-Aircraft**
NTS-Net	ResNet50	87.50%	93.90%	91.40%
DFL-CNN	VGG-16	86.70%	93.80%	92.00%
DCL	ResNet50	87.80%	94.50%	93.00%
MMAL-Net	ResNet50	89.60%	95.00%	94.70%
WS-DAN	Inception v3	89.40%	94.50%	93.00%
Baseline	ResNet50	85.49% (85.85%)	93.04% (93.17%)	91.07% (91.30%)
Mid-level baseline	ResNet50	87.13%	93.80%	91.68%
Baseline + SnapMix	ResNet50	87.75% (88.01%)	94.30% (94.59%)	92.80% (93.16%)
Mid-level baseline + SnapMix	ResNet50	88.70% (88.97%)	95.00% (95.16%)	93.24% (93.49%)
Baseline + SADMix	ResNet50	88.23% (88.75%)	94.43% (94.55%)	93.13% (93.43%)
Mid-level baseline + SADMix	ResNet50	89.01% (89.23%)	95.19% (95.33%)	93.53% (93.71%)

on the selected semantic and attentive box region, the performance comparison of SADMix with/without random adjustment is shown in Table 1. Further, the performance comparison of data augmentation methods is shown in Table 2. Then, SADMix is tested using two baselines and compared with the state-of-the-art FGVC methods, and the corresponding results are shown in Table 3. Top-1 accuracy is used as a performance evaluation indicator. Both the best accuracy and average accuracy of SADMix are provided.

Each data augmentation method is reimplemented in the experiments. When the experimental results are presented, the original performance is shown if the reimplemented performance is lower than the original performance.

Table 1 shows the results of SADMix with/without random adjustment on three fine-grained object datasets. SADMix with random adjustment outperforms the one without random adjustment. As a main reason, the random adjustment on the selected semantic and attentive box region can introduce more useful information in the training phase. Table 1 shows the average accuracy of the last 10 epochs. Since the validity has been verified, all subsequent experimental results are obtained by SADMix with random adjustment.

Table 2 shows the performance comparison of data augmentation methods on three fine-grained object datasets. The proposed SADMix consistently outperforms the comparative data augmentation methods. The experimental results show that deeper CNNs have better performance in all data augmentation methods. The results listed in Table 2 are obtained by using the baselines without mid-level features. Table 2 shows the average accuracy of the last 10 epochs.

Table 3 shows the performance comparison with the state-of-the-art FGVC methods. In Table 3, the results of FGVC methods are directly cited from the original papers. The results of SnapMix and SADMix are calculated by standard baselines and mid-level baselines, respectively. Moreover, the average accuracy of the last 10 epochs is reported and the best accuracy is shown in the brackets. As shown in Table 3, MMAL-Net performs better on CUB-200-2011 dataset than all other comparative methods. WS-DAN achieves sub-optimal performance on CUB-200-2011. SADMix with mid-level baseline achieves 89.23% accuracy, which is lower than MMAL-Net and WS-DAN. As a possible reason, CUB-200-2011 is more complex, which requires the model to have more powerful feature representation ability. MMAL-Net has three branches to describe global and local features simultaneously, while WS-DAN has a BAP module for feature representation. In FGVC-Aircraft, MMAL-Net also achieves the best performance. The proposed SADMix with mid-level baselines achieves the second-best performance. In Stanford-Cars, SADMix with mid-level baselines outperforms than all comparative methods.

The performance improvements of SADMix over the data augmentation methods are relatively low on Stanford-Cars

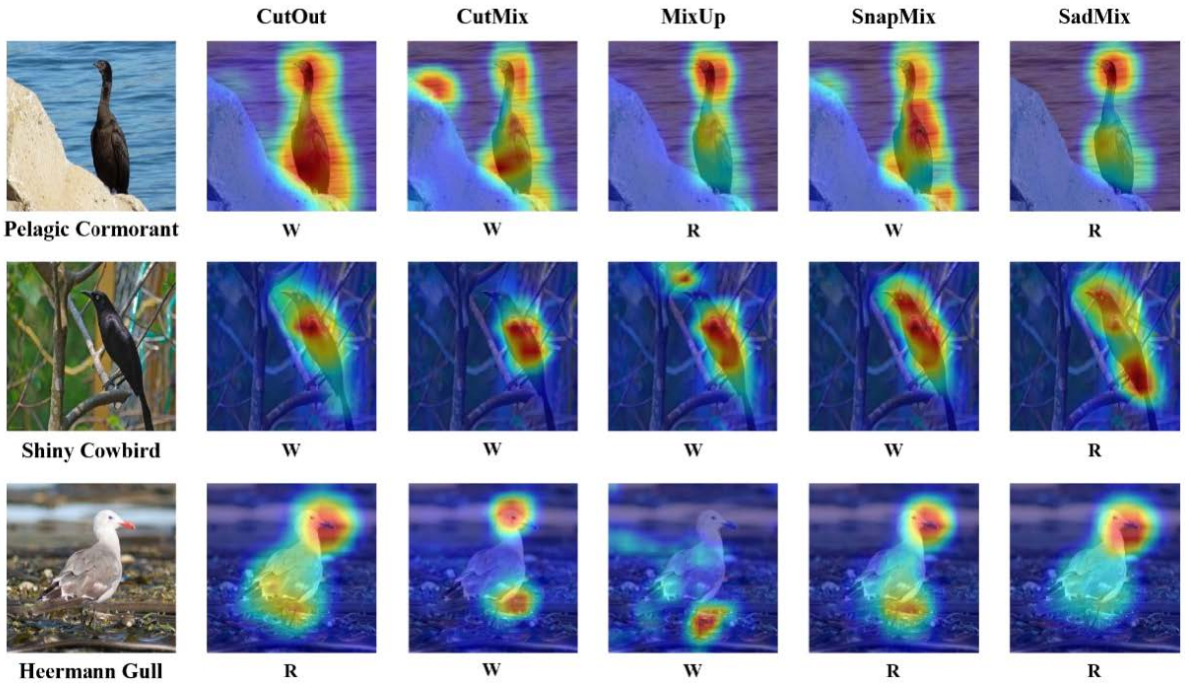

FIGURE 3. Class activation maps visualization of different data augmentation methods on CUB-200-2011.

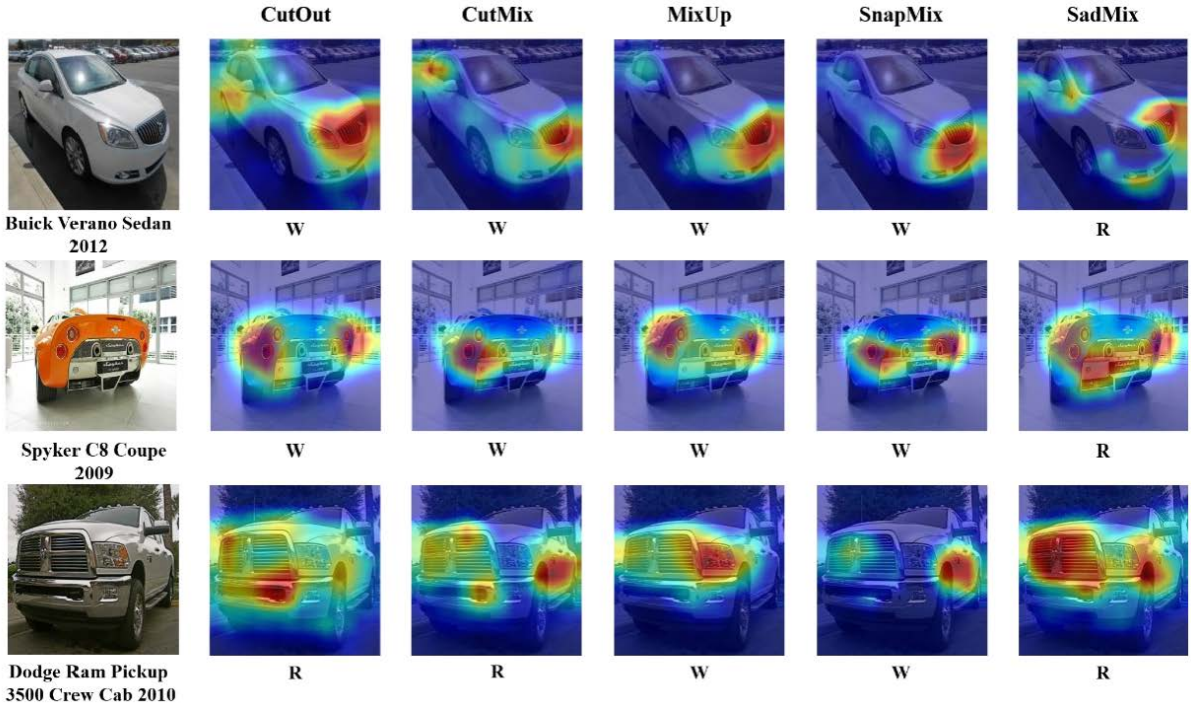

FIGURE 4. Class activation maps visualization of different data augmentation methods on Stanford-Cars.

and FGVC-Aircraft than on CUB-200-2011. The main reason is that the structure of vehicles and aircraft is relatively fixed. Compared with birds, the structure variation of vehicles and aircrafts is limited.

E. VISUALIZATION ANALYSIS

Class activation map visualization of different data augmentation methods is shown in Figures 3, 4, and 5. Three images are selected from each dataset used in the experiments.

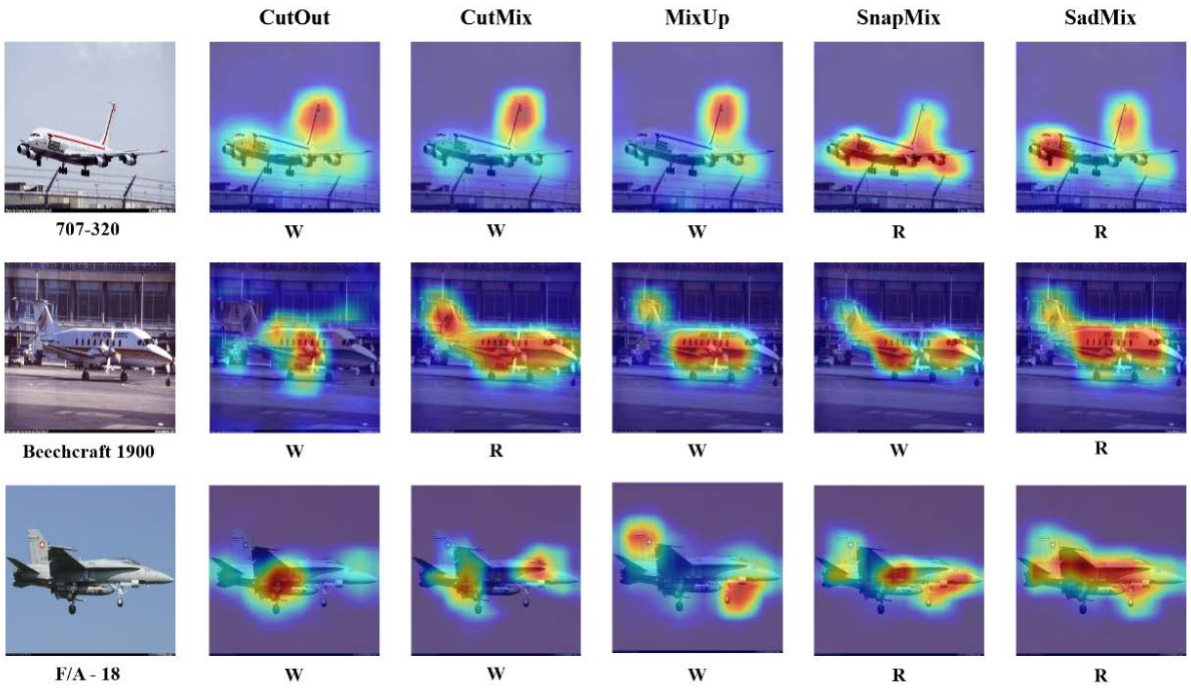

FIGURE 5. Class activation maps visualization of different data augmentation methods on FGVC-Aircraft.

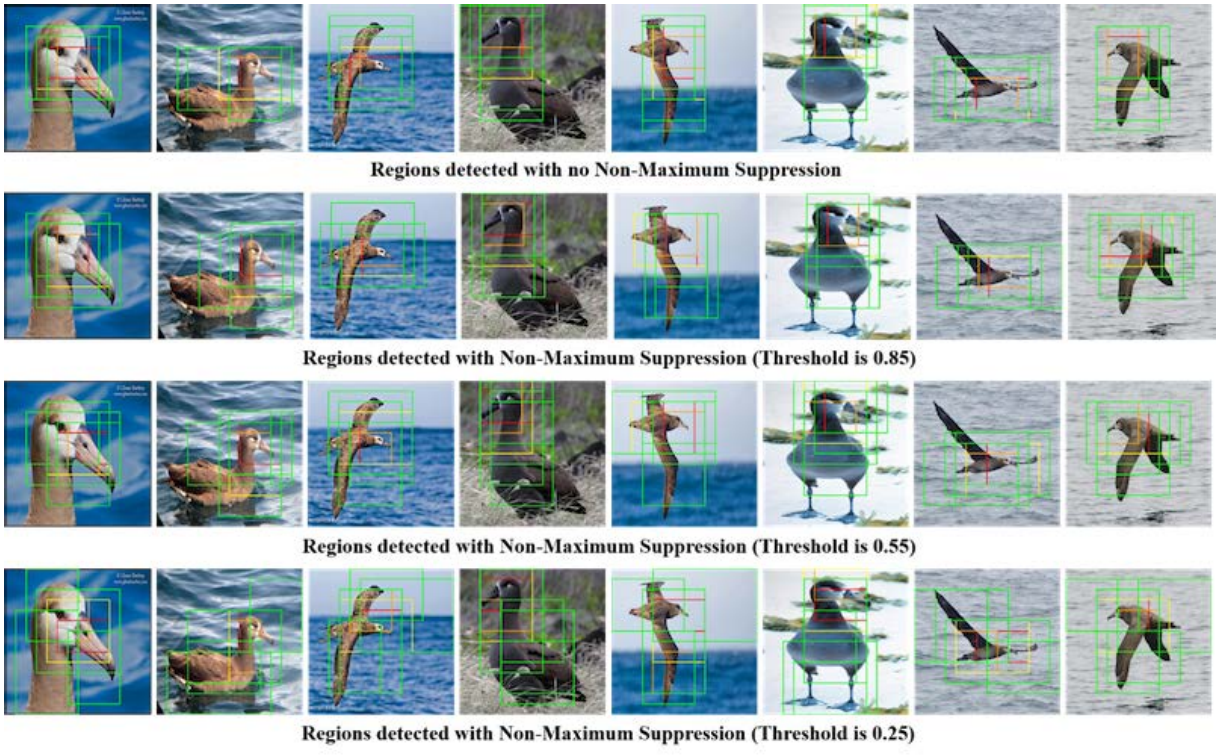

FIGURE 6. The effect of NMS with different threshold values.

Each image is correctly predicted by SADMix and maybe misclassified by several other methods. If an image is correctly classified, the image is marked in a capital R. If an image is misclassified, the image is marked in a capital W.

In order to reflect the ability of the network models trained by different data augmentation methods in the localization of critical regions, class activation map (CAM) [10] is used to make visualization analysis of the regions concerned by

TABLE 4. Performance comparison of SADMix with different NMS thresholds.

	CUB-200-2011		Stanford-Cars		FGVC-Aircraft	
Thresholds	ResNet34	ResNet50	ResNet34	ResNet50	ResNet34	ResNet50
0.25	87.16%	87.98%	93.78%	94.02%	92.85%	92.88%
0.55	87.37%	88.23%	94.08%	94.43%	92.92%	93.13%
0.85	87.03%	87.85%	93.57%	93.84%	92.60%	92.71%
No NMS	86.88%	87.52%	93.43%	93.65%	92.07%	92.35%

the networks. In CAM, the output feature maps of the final convolutional layer are fed into global average pooling. The global average pooling results are used as the features for classification. The importance of the image regions can be identified by projecting the weights of the fully connected layer on the final convolutional feature maps. In visualization analysis, the weights corresponding to the predicted class with the highest values in the fully connected layer are projected on the convolutional feature maps. Resnet50 is used in visualization analysis.

As shown in Figures 3, 4, and 5, the proposed SADMix, which outperforms the related data augmentation methods on three fine-grained datasets, not only localizes the critical regions more accurately, but also explores more critical region information. For example, the network trained by SADMix can localize both heads and tails of Shiny Cowbird for recognition. Networks trained by the remaining related data augmentation methods only localize parts of critical regions and cause misclassification.

F. EFFECT OF NMS

Figure 6 shows the effect of the NMS method on region detection at different thresholds. Each region is selected based on the activation mean value in the feature map. The boxes marked with different colors have different window sizes. It is observed that when NMS is not used, the detected regions are clustered together with a large area of overlapping. As the threshold value in the NMS decreases from 0.85 to 0.25, the selected regions gradually disperse and tend to cover different semantic parts of the object. This observation allows us to understand the purpose of NMS, which effectively reduces the redundancy and enhances the diversity of the highlighted semantic regions, leading to a performance gain.

To further examine the effect of NMS on the model performance, we evaluated SADMix with three NMS threshold values, including 0.25, 0.55, 0.85. Also, a baseline model without the usage of NMS was evaluated (i.e., a threshold of 1.0). Results on the three aforementioned datasets are reported in Table 4. it is observed that with an NMS threshold of 0.55, the resulting model consistently outperforms models with other NMS thresholds. This result verifies our hypothesis that as the threshold increases, a more diversified set of semantic regions can be selected and used for training sample augmentation, leading to performance gains. However, as the threshold keeps decreasing, the selected regions start to disperse and more background contents are included, which lowers the quality of the patches used for augmentation. It is expected that the model without NMS has the lowest accuracy due to the excessive redundancy among the selected regions.

V. CONCLUSION

This paper proposes a data mixing augmentation method termed as SADMix for fine-grained visual categorization. New training samples are generated by SADMix with the normalized CAM and SAPL modules. Different from existing solutions, random box region generation is replaced by a SAPL module. Adding a certain amount of random adjustment to windows with the fixed size and aspect ratio during localization is conducive to improving performance. As shown in subsection 4.4 and 4.5, the networks trained by SADMix can achieve better classification performance and stronger localization capacity.

It is worth pointing out that the computational complexity of SADMix is higher than related data augmentation methods, such as MixUp and SnapMix, due to the added process of localizing discriminative regions. Although the complexity has increased, the proposed SADMix has achieved consistent performance gains on three datasets. In real-world deep learning applications, there is always a trade-off between performance and speed. The latter can be improved by upgraded hardware, while the former can only be improved with a better model design or an enhanced dataset. Compared to its peers, SADMix aims to maximize a model's ability to mine discriminative patterns via data augmentation, outperforming the SOTA. Therefore, the superiority of SADMix can be justified and validated through the conducted experiments. Meanwhile, our investigation shows that SADMix and its peers lack comprehensive algorithm complexity analysis. This interesting future direction can offer theoretical support for certain building blocks utilized in this study.

The conclusions of this paper can be summarized as follows. 1) Random region generation in previous data augmentation methods, such as Cutout, Cutmix, Mixup and Snapmix, may introduce meaningless training samples. 2) The region generated by the attention mechanism module for mixing data can avoid generating meaningless training samples, and the generalization of the network and useful information can be improved by the random modification of the regions generated by the attention mechanism module.

Two research directions are worth further investigation. First, is it better to use contours, as shown in the activated areas in the Figure 5, rather than bounding boxes to perform SADMix? Intuitively, the contours contain more descriptive fine-grained information with less background noise. However, we anticipate two challenges: 1) different contours vary

in shape and size, leading to a difficulty in mixing them; 2) extracting contours is less straightforward compared to the extraction of bounding boxes. With proper strategies to resolve these challenges, contours can be more promising. Second, the usage of a generative model such as a generative adversarial network (GAN) in the FGVC task is also interesting. One benefit of GAN is its ability to generate synthetic samples that are not seen by the learning algorithm, leading to a more diverse training set. The challenge of GAN is that for FGVC, the learning algorithm should exploit the fine-grained features, which should be emphasized during training. Regular GANs, however, are more focused on the global features rather than the fine-grained ones. The proposed SADMix method allows the learning algorithm to focus on the fine-grained patterns, which may be highlighted by the semantic regions used for data mixing. Both methods have their pros and cons. It would be interesting to design a custom GAN in a way that both global and fine-grained features are both well captured to generate high quality training samples specific for the FGVC task. A more in-depth study is needed to apply GAN for this purpose.

ACKNOWLEDGMENT

The authors thank the anonymous reviewers for their careful reading of their manuscript and their insightful comments, which are essential to the improvement of the manuscript quality.

REFERENCES

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

[2] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," 2017, *arXiv:1708.04552*.

[3] Y. Tokozume, Y. Ushiku, and T. Harada, "Between-class learning for image classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5486–5494.

[4] H. Inoue, "Data augmentation by pairing samples for images classification," 2018, *arXiv:1801.02929*.

[5] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond empirical risk minimization," 2017, *arXiv:1710.09412*.

[6] S. Huang, X. Wang, and D. Tao, "SnapMix: Semantically proportional mixing for augmenting fine-grained data," in *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35, no. 2, pp. 1628–1636.

[7] S. Yun, D. Han, S. Chun, S. J. Oh, Y. Yoo, and J. Choe, "CutMix: Regularization strategy to train strong classifiers with localizable features," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6023–6032.

[8] H. Li, X. Zhang, Q. Tian, and H. Xiong, "Attribute mix: Semantic data augmentation for fine grained recognition," in *Proc. IEEE Int. Conf. Vis. Commun. Image Process. (VCIP)*, Dec. 2020, pp. 243–246.

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

[10] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2921–2929.

[11] F. Zhang, M. Li, G. Zhai, and Y. Liu, "Multi-branch and multi-scale attention learning for fine-grained visual categorization," in *Proc. 27th Int. Conf. MultiMedia Modeling (MMM)*, Prague, Czech Republic, vol. 12572, in Lecture Notes in Computer Science. Berlin, Germany: Springer, Jun. 2021, pp. 136–147.

[12] Z. Yang, T. Luo, D. Wang, Z. Hu, J. Gao, and L. Wang, "Learning to navigate for fine-grained classification," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 420–435.

[13] B. Zhao, X. Wu, J. Feng, Q. Peng, and S. Yan, "Diversified visual attention networks for fine-grained object classification," *IEEE Trans. Multimedia*, vol. 19, no. 6, pp. 1245–1256, Jun. 2017.

[14] 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.

[15] 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)*, 2018, pp. 805–821.

[16] 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.

[17] 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.

[18] 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.

[19] T. Hu, H. Qi, Q. Huang, and Y. Lu, "See better before looking closer: Weakly supervised data augmentation network for fine-grained visual classification," 2019, *arXiv:1901.09891*.

[20] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated recognition, localization and detection using convolutional networks," 2013, *arXiv:1312.6229*.

[21] 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.

[22] 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.

[23] S. Maji, E. Rahtu, J. Kannala, M. Blaschko, and A. Vedaldi, "Fine-grained visual classification of aircraft," 2013, *arXiv:1306.5151*.

[24] O. Russakovsky, J. Deng, H. Su, and J. Krause, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

MENGQI HE is currently pursuing the degree in advance computing with The Australian National University, specializing in machine learning. He is expected to graduate, in 2022, and continue his master's degree in machine learning and computer vision. He is experienced and passionate in algorithm and software development. He is interned as an Algorithm Engineer at PwC, JD.com, and CAS. In 2021, during his intern at CAS, he has contributed a reinforcement learning-based PID control algorithm for the Institute of Launch Vehicle.

QILONG (JERRY) CHENG is currently pursuing the degree in mechanical engineering with the University of Toronto. He is expected to graduate, in 2022, and keep on pursuing his M.Eng. degree. He was selected as the University Campus Ambassador and a Developer by Autodesk. He had been the President of Fusion Design Association. From 2020 to 2021, he has contributed a high pressure spray nozzle design patent for China State Shipbuilding Corporation during his intern.

GUANQIU QI received the Ph.D. degree in computer science from Arizona State University, in 2014. He is currently an Assistant Professor with the Computer Information Systems Department, The State University of New York at Buffalo. His primary research interests include deep learning, machine learning, and image processing, and also span many aspects of software engineering, such as software-as-a-service (SaaS), testing-as-a-service (TaaS), big data testing, combinatorial testing, and service-oriented computing.