IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received March 20, 2020, accepted April 3, 2020, date of publication April 9, 2020, date of current version April 27, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2986771

A New Benchmark for Instance-Level Image Classification

KAI KANG, GANGMING PANG, XUN ZHAO, JIABAO WANG, AND YANG LI
College of Command and Control Engineering, Army Engineering University of PLA, Nanjing 210007, China
Corresponding author: Jiabao Wang (jiabao_1108@163.com)

ABSTRACT Although fine-grained image classification is able to classify more fine-grained sub-categories compared to its coarse-grained counterpart, it often fails to identify individual instances. Therefore, we propose a new instance-level image classification task which further refines the granularity of fine-grained classification in order to identify unique instances rather than a sub-category containing multiple instances. In addition, we introduce an instance-level image classification dataset, AircraftCarrier, which contains 20 global aircraft carrier classes, as the first publically available dataset for instance-level image classification. The classification of instance-level aircraft carriers can prove to be a challenging task due to large intra-category differences as well as variations in the camera view, illumination, scale, and the presence of complex backgrounds. The AircraftCarrier dataset put forward here has the potential to improve the development of instance-level image classification. At the same time, we provide a Simple Classification Head (SCH) technique for the classification of aircraft carriers, with classical convolutional neural network models as the backbone network. The SCH has better performance than a direct classification head, and these results provide a benchmark performance result for researchers. Furthermore, we evaluate several fine-grained image classification methods and give their benchmark results. Finally, we present the challenges of instance-level classification and discuss further directions. This study provides the first publicly available instance-level image classification dataset and a performance benchmark for further research. The dataset and codes can be downloaded at https://github.com/tsingqsu/AircraftCarrier_Dataset/.

INDEX TERMS Image classification, aircraft carrier dataset, instance-level classification, fine-grained classification.

I. INTRODUCTION

In computer vision, image classification usually refers to the classification of coarse-grained objects (e.g. people, birds and ships) [1]. The accuracy of such classification techniques has been greatly promoted by the development of deep learning technology [2]. However, the coarse-grained image classification is not able to effectively address the task of fine-grained image classification [3], [4]. In particular, fine-grained image classification refers to a sub-category classification of a coarse-grained super-category, such as the classification between "destroyer", "frigate", "supply ship" and "aircraft carrier" in the "ship" category. In order to overcome the obstacles faced by coarse-grained image classification, in the current study, we present an aircraft carrier dataset, AircraftCarrier, in which each category corresponds to an instance of aircraft carriers, for example, "Liaoning (CV16)", "USS Nimitz (CVN68)" and "Charles de Gaulle (R91)". Each of the classes corresponds to an aircraft carrier instance, thus we denoted this process as instance-level classification. Figure 1 presents the semantic diagram of coarse-grained, fine-grained and instance-level classification.

The progress of deep neural networks has resulted in vast improvements in accuracy in coarse-grained image classification, as well as the development of visual object recognition techniques, such as AlexNet [1], VGGNet [5], ResNet [6] and DenseNet [7]. Furthermore, with the recent advancement of part detectors [8], [9] and attention mechanisms [10]–[12], the accuracy of fine-grained image classification has also improved. However, research on instance-level image classification is lacking. In particular, studies tend to focus more on the feature representation of face recognition [13] and person re-identification [14], overlooking instance-level image classification. Despite this, research on instance-level classification is highly important. For example,

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

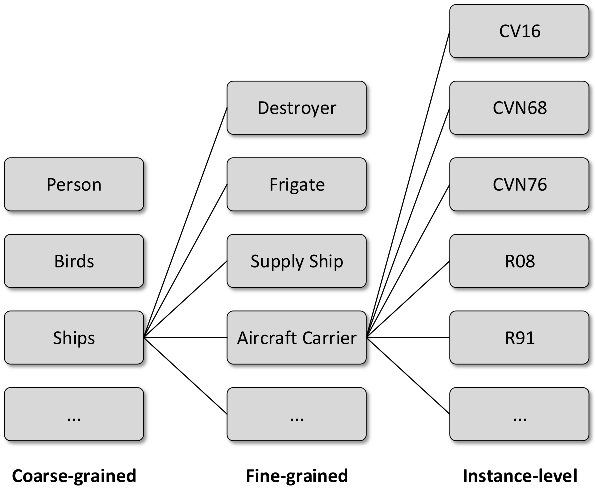

FIGURE 1. Semantic diagram of coarse-grained, fine-grained and instance-level classification. Left: Coarse-grained classification (e.g. "person", "birds" and "ships"). Middle: Fine-grained classification, for example, "destroyer", "frigate", "supply ship" and "aircraft carrier", in the "ship" category. Right: Instance-level classification of the "aircraft carrier" sub-category (e.g. "Liaoning (CV16)", "USS Nimitz (CVN68)" and "Charles de Gaulle (R91)").

aircraft carriers are key military objects across the globe, thus research in the instance-level classification of aircraft carrier images is of great significance.

Instance-level image classification can be considered to be the most fine-grained classification, and is also more difficult to execute compared to the standard fine-grained classification. This is because fine-grained classification categories can contain multiple instances, while each instance-level classification category contains a unique instance. In general, instance-level classification categories are made up of one fine-grained classification category, with limited inter-category variation. Moreover, each instance exhibits variations in perspective, light, shade and so on, making instance-level classification more complicated. In order to improve such instance-level classification methods, we created an instance-level image classification dataset and present the baseline results.

The main contributions of this study are as followings:

- A new instance-level image classification task is proposed and an instance-level aircraft carrier image classification dataset (AircraftCarrier) is presented. The dataset is the first published aircraft carrier image classification dataset and exhibits a more refined granularity compared to fine-grained image classification datasets.
- The Simple Classification Head (SCH) method is proposed in order to classify aircraft carriers via the use of classical convolutional neural network (CNN) models as the backbone network. Existing high-precision and light-weighted models and fine-grained image classification methods are evaluated using the AircraftCarrier dataset. The results provide a baseline for further research. Furthermore, future research directions for instance-level classification are discussed.

II. RELATED WORK

According to our knowledge, there is no work directly related to instance-level image classification, and the indirectly related works mainly include fine-grained classification and instance recognition.

A. FINE-GRAINED CLASSIFICATION

There are numerous datasets available for fine-grained image classification (e.g. CUB200,[1] Stanford Dog,[2] Stanford Car,[3] FGVC Aircraft,[4] NABirds [15], iNat2017 [16] and RPC2019).[5] However, such datasets are distinct to those required for instance-level classification. In particular, for fine-grained image classification, each category contains different object instances, rather than the same object instance as in instance-level classification. In terms of classification granularity, instance-level image classification is considered as the limitation of fine-grained classification, as it is a highly challenging task.

Over the recent years, many state-of-the-art methods have been developed for fine-grained image classification, and can be classified into regional feature-based methods and global feature-based methods. The former include detector-based and attention-based approaches, such as Part R-CNN [17], HSnet Search [18], RA-CNN [10], MA-CNN [11], DFL-CNN [12], WS-DAN [19] and TASN [20]. Global feature-based methods apply metric learning to increase accuracy, with examples including Bilinear CNN (B-CNN) [21] and its invariants [22], [23]. Furthermore, several triplet loss-based methods have been developed, such as MAMC [24] and MMLN [4].

B. INSTANCE RECOGNITION

Instance recognition refers to the identification of an instance [25], with each instance belonging to a different category. Different from instance-level classification, instance recognition is considered as a matching problem of specific instances, while instance-level classification is the classification of multiple similar instances in a sub-category. Various datasets already exist for instance recognition, containing different instances from different categories for identification (e.g. RGB-D[6] and BigBIRD [26]), thus making this problem distinct to instance-level classification. Instance recognition datasets are frequently applied for image retrievals, where the instance objects are retrieved from a large database via a query image. The key-point [27], image block [28] and deep feature [29] matching methods are frequently used to recognize a specific instance by matching local feature blocks, feature points or feature vectors. Face recognition [13] and person re-identification [14] are two types of instance-level datasets, yet researchers generally focus on the feature

[1] http://www.vision.caltech.edu/visipedia/CUB-200-2011.html
[2] http://vision.stanford.edu/aditya86/StanfordDogs/
[3] https://ai.stanford.edu/∼ jkrause/cars/car_dataset.html
[4] http://www.robots.ox.ac.uk/∼ vgg/data/fgvc-aircraft/
[5] https://rpc-dataset.github.io/
[6] http://www.cs.washington.edu/rgbd-dataset

representation of the identification of a new face or person, which differs from instance-level image classification. In order to overcome this, we present our instance-level image classification dataset.

The rest of the paper is organized as follows. First, we present the first publically available aircraft carrier dataset, containing a large number of warships from across the globe. Then, we introduce six classical convolutional neural networks as the backbone network, and propose our Simple Classification Head (SCH) technique to construct several classification models. This is followed by an evaluation of the models using the aircraft carrier dataset. Finally, we discuss the challenges and future directions of instance-level image classification tasks.

III. INSTANCE-LEVEL DATASETS

A. DATASET COLLECTION

To the best of our knowledge, regardless of face recognition and person re-identification, there are no publicly available datasets for instance-level image classification tasks. In order to fill in this gap, we built an aircraft carrier image classification dataset using images selected from the Internet via the Baidu and Google search engines. In addition to this, images were also manually collected from professional military websites (including navy.mil,[7] denfense.gov,[8] defense.gov).[9] The dataset contains the following 20 categories: "Liaoning (CV16)", "Cavour (CVH550)", "Giuseppe Garibaldi (CVH551)", "USS Nimitz (CVN68)", "USS Dwight David Eisenhower (CVN69)", "USS Carl Vinson (CVN70)", "USS Theodore Roosevelt (CVN71)", "USS Abraham Lincoln (CVN72)", "USS George Washington (CVN73)", "USS John C. Stennis (CVN74)", "USS Harry S. Truman (CVN75)", "USS Ronald Reagan (CVN76)", "USS George H.W. Bush (CVN77)", "USS Gerald R. Ford (CVN78)", "Juan Carlos I (L61)", "HMS Queen Elizabeth (R08)", "INS Viraat (R22)", "Charles de Gaulle (R91)", "HTMS Chakri Naruebet (R911)" and "Admiral Flota Sovetskogo Soyuza Kuznetsov (RN063)". Figure 2 presents examples of the 20 aircraft carriers.

B. DATASET CHARACTERISTICS

The AircraftCarrier dataset contains 2781 images, with the number of images in each category varying from 53 to 208, and an average of 150 images. Figure 3 presents the size distribution of the dataset. Note the CVH551, R22 and R911 categories contain relatively low numbers of images, due to the images of these categories are very few on the Internet.

Although the instance-level AircraftCarrier dataset is not directly captured by camera, it still exhibits a high amount of variation between images. The images collected from the Internet differed in terms of perspective, illumination, scale and complex background types. Figure 4 provides examples of the variation in image characteristics.

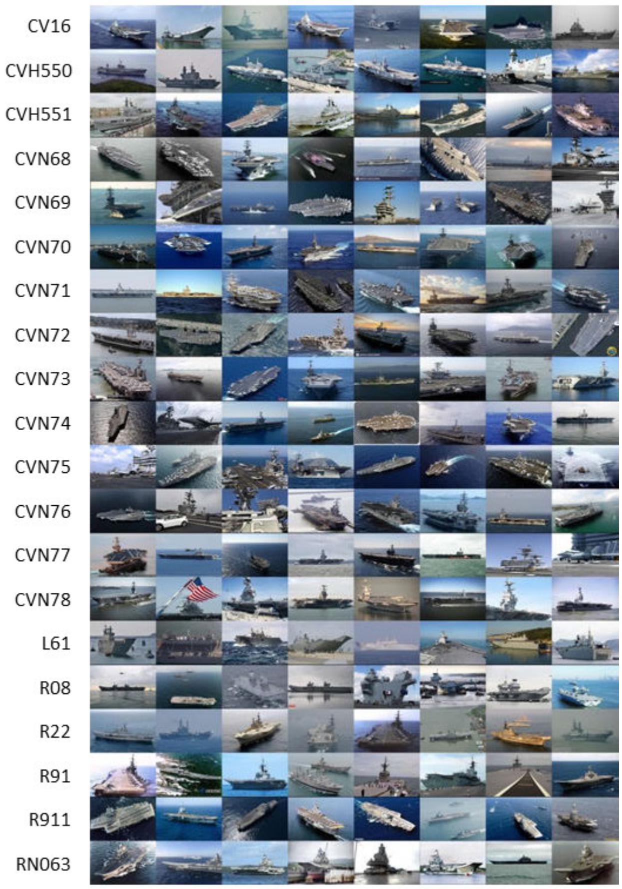

FIGURE 2. Examples of the 20 aircraft carriers used in our AircraftCarrier dataset. Each row corresponds to a category.

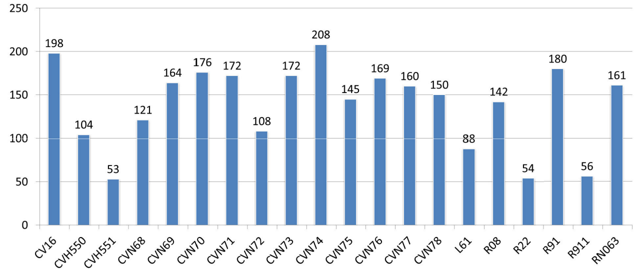

FIGURE 3. The number of images in each category of the AircraftCarrier dataset.

In particular, Figure 4 shows that images within the same category present variations in terms of viewing angle, scale, and lighting condition. Thus, it is evident that the aircraft carrier dataset exhibits great differences in multiple characteristics. Such intra-category variations in instance-level image classification make the classification task much more of a challenge.

C. DATASET USAGE

In order to evaluate the effectiveness of the instance-level classification on the AircraftCarrier dataset, we randomly

[7] https://www.navy.mil/
[8] https://www.usni.org/
[9] https://www.defense.gov/

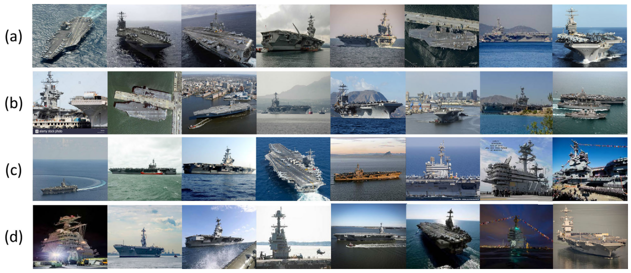

FIGURE 4. Intra-category differences of instance-level aircraft carrier images. (a) Different image perspectives from the CVN72 category, (b) images with complex backgrounds from the CVN75 category, (c) different image scales from the CVN77 category, and (d) different image illumination from the CVN78 category.

TABLE 1. The partitioning of the AircraftCarrier dataset. "#train", "#val" and "#test" refer to the number of training, validation, and testing images, respectively.

Category	#train	#val	#test	Category	#train	#val	#test
CV16	108	51	39	CVN75	94	22	29
CVH550	59	25	20	CVN76	106	33	30
CVH551	28	15	10	CVN77	98	31	31
CVN68	79	19	23	CVN78	80	39	31
CVN69	95	37	32	L61	48	20	20
CVN70	95	44	37	R08	82	24	36
CVN71	103	33	36	R22	35	8	11
CVN72	64	19	25	R91	108	45	27
CVN73	105	29	38	R911	31	13	12
CVN74	111	49	48	RN063	98	29	34

divided the dataset into three sections, the training set, the validation set and the test set, with a ratio of 3:1:1. The results are reported in Table 1.

IV. EVALUATION METHODS

A. BACKBONE NETWORKS

Three high-precision and three light-weighted models were used as the basic backbone network to build a classifier for the AircraftCarrier dataset. These backbone networks are briefly described as follows:

a) SqueezeNet [30]: SqueezeNet applies the squeeze operation in order to reduce the feature map channels replacing the 3 × 3 convolutional filters with 1 × 1 filters, and subsequently expands the channels by concatenating the results of the 3 × 3 and 1 × 1 filters. SqueezeNet is able to achieve AlexNet-level accuracies on ImageNet with 50× fewer parameters, amounting to less than 1.0 M.

b) ShuffleNetV2 [31]: ShuffleNet proposes a new type of channel shuffle operation for group convolution. This group operation is able to reduce computation costs, while the channel shuffle operation exchanges information among different groups. Accuracy levels are maintained when the computation is reduced. Furthermore, in order to avoid the large amount of group convolutions of ShuffleNetV1, ShuffleNetV2 adopts a shuffle split strategy, thus achieving a more competitive performance.

c) MobileNetV2 [32]: MobileNet proposes a depthwise separable convolution with depthwise and pointwise layers in order to compress the size of the model. The architecture is based on an inverted residual structure where the channels are initially increased and subsequently decreased. MobileNetV2 builds on MobileNet by including a pointwise convolution before the depthwise convolution to increase the channels, and also removes the second activation function to form a line bottleneck.

d) VGGNet-16 [5]: VGGNet is the first network to use 3 × 3 convolution filters in order to build networks with a depth of dozens of weight layers. VGGNet-16 has 13 convolution layers and 3 fully connection layers. It achieves a better performance than former deep convolutional neural networks, at the cost of, however, greater computation and an increased number of parameters.

e) ResNet-50 [6]: In ResNet, the deep network is built via a residual block. The identity mapping creates a direct connection from the input to the output, thus reducing the back propagation path and avoiding the vanishing and exploding of the gradient. Furthermore, compared with the original block of two 3 × 3 convolutions, the improved block of 1 × 1, 3 × 3, 1 × 1 convolutions reduces the parameters and computations without causing any drop in performance. ResNet-50 is the most widely used CNN for classification purposes.

f) DenseNet-121 [7]: DenseNet is a CNN with dense connections. In particular, there is a direct connection between any two layers. More specifically, the input of each layer of the network is the union of the output of all the previous layers, and the feature map learned by a particular layer is also directly transmitted to all the subsequent layers as an input. Due to this dense connection, DenseNet is able to enhance the back propagation gradient, thus making the network easier to train.

B. SIMPLE CLASSIFICATION HEAD

In order to unify their application, we delete all full connection and classification layers in the above networks, to propose a unified Simple Classification Head (SCH) method for image classification (Figure 5).

The SCH is principally composed of global average pooling (GAP), a dimension reduction module (DRM) and a softmax classifier. Note that the DRM is composed of a convolution (Conv) with the 1 × 1 filter, a batch normalization (BN) and a rectified linear unit (ReLU). Due to the small size of the dataset, during training, we include a dropout layer prior to the softmax classifier [33] with a discarding rate of 0.5. This avoids over-fitting while training.

Given an input image, the output of the backbone is a feature tensor $\mathbf{F} = [\mathbf{F}_1, \mathbf{F}_2, \ldots, \mathbf{F}_C] \in \mathbb{R}^{H \times W \times C}$, where $\mathbf{F}_k, k = 1, \ldots, C$ refers to the k-th feature map and C refers to the number of the channels of $\mathbf{F}$. A feature vector $\mathbf{x}$ can then be determined via the GAP operation, with the c-th element of $\mathbf{x}$ calculated as follows:

$$\mathbf{x}_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} \mathbf{F}_c(i, j), \tag{1}$$

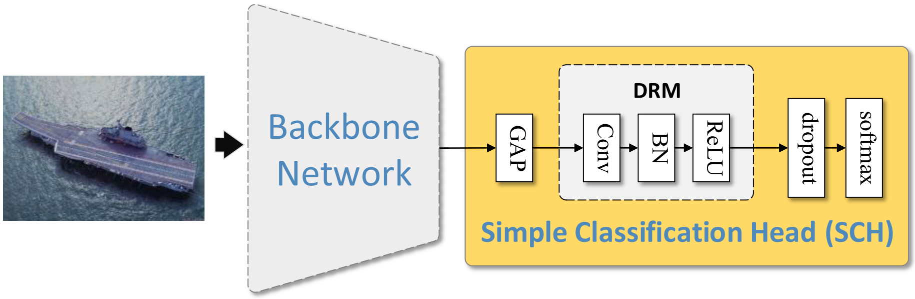

FIGURE 5. The proposed unified Simple Classification Head (SCH) method, composed of a global average pooling (GAP), a dimension reduction module (DRM) and a softmax classifier. DRM consists of a 1 × 1 convolution (Conv), a batch normalization (BN) and rectified linear unit (ReLU). A dropout layer is added before the softmax classifier to avoid over-fitting.

where H and W denote the height and width of the c-th feature map $\mathbf{F}_c$, respectively. Following this, in the DRM, the 1×1 convolution is applied in order to reduce the channel dimension of $\mathbf{x}$ to 512. Batch normalization and the ReLU function are then executed. The output $\mathbf{f} \in \mathbb{R}^{1\times1\times512}$ is determined by

$$\mathbf{f} = \delta(BN(\mathbf{K} \otimes \mathbf{x})), \tag{2}$$

where δ denotes the ReLU function, BN is the batch normalization operation, $\otimes$ is the convolution operation and $\mathbf{K} \in \mathbb{R}^{1\times1\times C\times512}$ is a convolutional kernel parameter for the dimensionality-reduction.

In order to improve the generalization performance, the label smoothing softmax classifier [34] is then applied, with the loss is computed as follows:

$$L = \sum_{i=1}^{C} -q_i \log p_i, \tag{3}$$

where

$$q_i = \begin{cases} 1 - \dfrac{C-1}{C}\varepsilon, & i = y \\ \varepsilon / C, & i \neq y, \end{cases} \tag{4}$$

$$p_i = \frac{exp(\mathbf{W}_y^T \mathbf{f} + b_y)}{\sum_{j=1}^{C} exp(\mathbf{W}_j^T \mathbf{f} + b_j)}, \tag{5}$$

and y is the ground truth of the input image, C is the number of classes, $\mathbf{W}_j$ and b_j are the parameters to learn, and ε is a hyper-parameter set as 0.1.

V. EVALUATION AND ANALYSIS

A. IMPLEMENTATION DETAILS

Our experiments were performed on a deep learning workstation with the following specifications: 3.5 GHz Intel Core E5-2637v4 CPU, Nvidia GTX 1080Ti GPU with 11GB RAM, and the Ubuntu 16.04 operating system. The programming environment is based on the Python language, while the PyCharm integrated development environment and Pytorch deep learning toolkit were used for GPU accelerated training.

TABLE 2. Evaluation results of the AircraftCarrier dataset. "Prec(%)" refers to the test accuracy, "FLOPs(G)" refers to the floating point calculation, and "Params(M)" refers to the number of parameters. The **bold** and <u>underlined</u> numbers denote the optimal and the second optimal results of the methods in Section IV-A, respectively.

Methods	Prec(%)	FLOPs(G)	Params(M)
SqueezeNet	51.67	<u>0.03</u>	**1.00**
ShuffleNetV2	53.78	**0.02**	<u>1.79</u>
MobileNetV2	58.35	<u>0.03</u>	2.89
VGGNet-16	58.35	1.71	14.99
ResNet-50	**63.44**	0.41	24.57
DenseNet-121	<u>62.92</u>	0.29	7.49

1) TRAINING

Our network, which applies the ImageNet pre-training model, is trained using batch stochastic gradient descent (SGD) with a batch size of 32. The initial learning rate of the model backbone network is 0.001, with the exception of the SCH, where 0.01 is used. This is done in order to increase the learning rate of the newly added SCH and to increase the convergence of the network. We train the network for 120 epochs and decay the learning rate by 0.1 following the end of epochs 70 and 100 respectively. We apply this multi-stage learning rate in order to increase the learning speed via a large learning step at the start of the training, and subsequently to reduce the learning step at the later stage. This aims to improve the convergence of the optimization object.

2) DATA PREPROCESSING

All input images are uniformly normalized to 224 × 224 pixels. The training data was first expanded to 256 × 256 pixels, and then randomly cropped to 224 × 224 pixels and flipped horizontally with a probability of 0.5. The test data is scaled directly and uniformly to 224 × 224 pixels.

B. RESULTS AND ANALYSIS

The evaluation results for the AircraftCarrier dataset of the methods described in Section IV-A are presented in Table 2. In terms of classification accuracy, ResNet-50 achieved the highest accuracy of 63.44%, which was 11.77% higher than that of SqueezeNet (51.67%). In terms of FLOPs, ShuffleNetV2 exhibited the fastest computation speed and only 0.02 G FLOPs, while VGGNet-16 had the slowest computation speed, requiring 1.71 G FLOPs. In terms of model size, SqueezeNet required the lowest number of parameters (1.0 M), while ResNet-50 used the greatest (24.57 M), at approximately 25 times greater than that of SqueezeNet. In summary, greater FLOP values and larger model sizes were observed for the high-precision models (ResNet-50, DenseNet-121 and VGGNet-16), while fewer FLOPs and smaller model sizes were associated with relatively low classification accuracies.

Note that SqueezeNet and ShuffleNetV2 have the worst accuracies. In terms of model size, they are the smallest and the second smallest models. Compared with ResNet-50 and DenseNet121, they have fewer network layers and parameters. Therefore, the ability of the model is relatively weak and the performance is relatively poor. It is worth noting that the MobileNetV2 has a relatively better performance, because it uses the reverse residual structure to increase the number of channels of features, which has stronger feature representation ability. Besides, the same accuracy level (58.35%) was determined for VGGNet-16 and MobileNetV2. This is because VGGNet-16 is an early and relatively basic approach, and lacks further improvements, such as batch normalization (BN).

Our results show that, in general, for the practical use of such methods, the classification accuracy, calculation speed and model size need to be considered comprehensively. For example, the SqueezeNet model is very small in size, making it suitable for embedded mobile platform applications with low precision requirements.

In order to further determine the errors and accuracies of the six classification models, we presented their confusion matrices (Figure 6).

In particular, the classification confusion matrices present the detailed classification accuracies of the 20 categories from the AircraftCarrier test set. The presence of non-zeros (except diagonal) indicates a misclassification between corresponding categories, yet no misclassifications were observed between the two categories here. The black diagonal denotes the accuracy of the classification. For example, Figure 6(e) demonstrates that all L61 classifications were correctly classified, while CVN72 only achieved an accuracy of 20%. Furthermore, Figure 6(a) shows that misclassifications were present for each category for various models, with misclassification rates varying across models. In particular, CVN68 exhibited an accuracy of just 13%, while the categories CVN68-CVN78 were easily misclassified. These prediction errors are attributed to the limited variations between the instances of the U.S. nuclear powered aircraft carriers in these categories. In contrast, the classification accuracy of aircraft carriers from China, Russia, India, Italy, England and France, amongst other countries, is relatively high. In addition to their hull numbers, other features are also easily distinguished, including the aircraft carrier tower.

Figure 7 presents some misclassified images of the ResNet-50 model, where T:XXX and P:XXX denote the truth and predicted labels, respectively. A total of 207 images were misclassified during the prediction.

From Figure 7, we can see that the categories CVN68-CVN78 are easily misclassified because they have the similar appearances. Moreover, most of the carriers in the images are small, and the hull number is invisible or unrecognizable. Although the hull number in some images is large enough (the fifth image in row 2), the category is still misclassified. It may be because the hull number feature is not learned by the model. In a same category, the images with hull number are few and have big difference from other images. Therefore, the instance-level aircraft carrier image classification is a very challenging task due to the large intra-category difference and small inter-category difference.

C. ABLATION ANALYSIS

To further evaluate the SCH, we have conducted ablation experiments about SCH by removing DRM. We named the SCH without DRM as DCH, in which the features output by GAP is directly connected to classifier. The results are reported in Table 3.

TABLE 3. Ablation analysis. "Prec(%)" refers to the test accuracy.

Backbone	SCH Prec(%)	DCH Prec(%)
SqueezeNet	51.67	44.99
ShuffleNetV2	53.78	31.45
MobileNetV2	58.35	58.34
VGGNet-16	58.35	56.06
ResNet-50	63.44	62.21
DenseNet-121	62.92	60.63

It can be seen from Table 3 that SCH has better performance than DCH. It is found that the performance of SqueezeNet and ShuffleNetV2 is greatly reduced after DRM is removed. The main reason is that these two models are very light-weighted models, and there are very few redundant parameters to adapt the changed objective. Furthermore, the output of GAP layer is directly connected to the classifier, and the adaptability of parameters is reduced during training. From the convergence curves of training losses (as shown in Figure 8), it can also be seen that SCH has better training convergence than DCH for SqueezeNet and ShuffleNetV2.

D. EVALUATION WITH FINE-GRAINED IMAGE CLASSIFICATION METHODS

In order to provide a more comprehensive evaluation, we evaluate the latest fine-grained image classification methods (B-CNN [21], DFL-CNN [21], WS-DAN [19], DCL [9], CrossX [3]) on the dataset. In order to compare with our existing methods, we set the input image size of these fine-grained image classification methods to 224 × 224 for training and testing, and the corresponding accuracies are shown in Table 4:

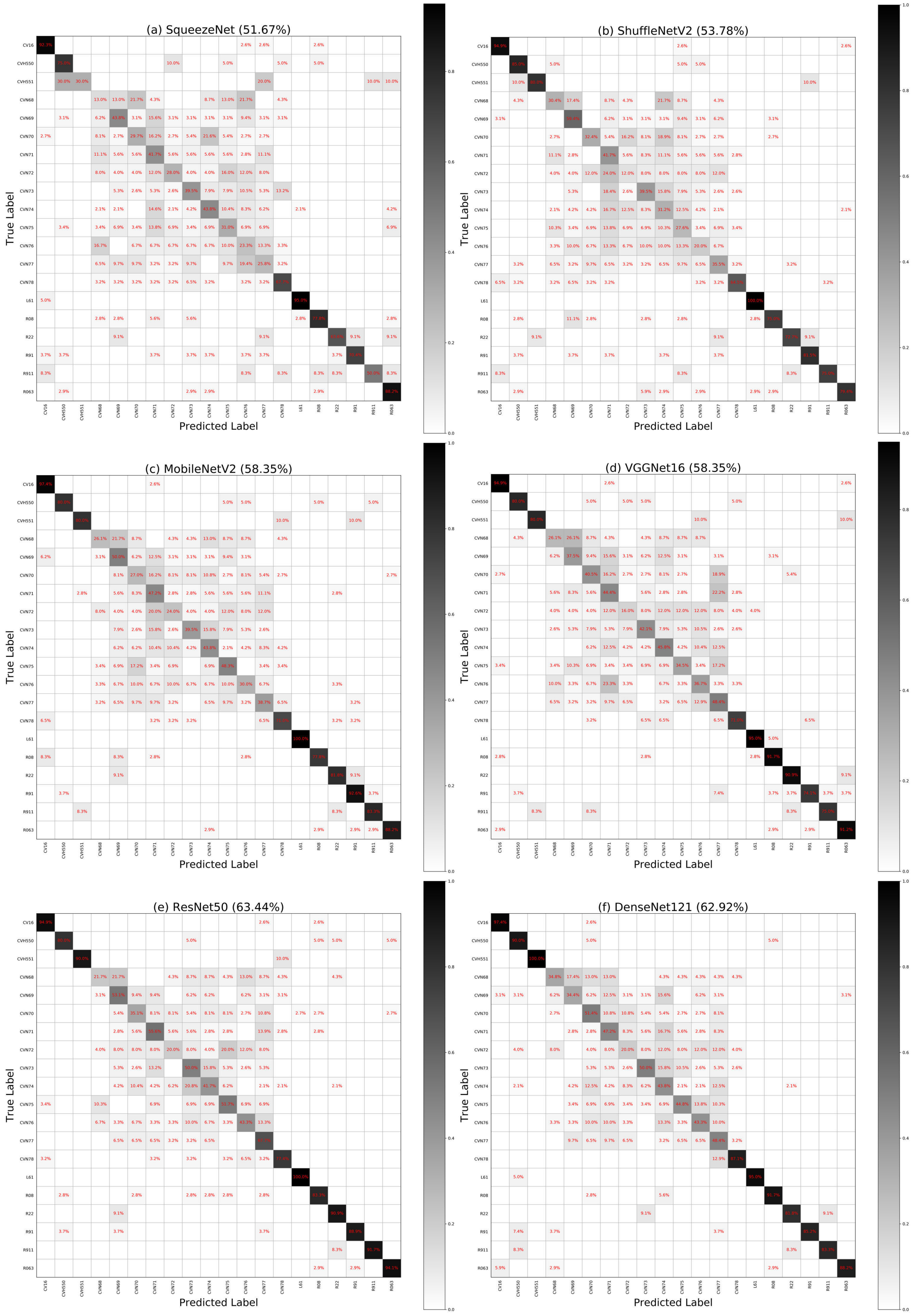

FIGURE 6. The confusion matrices of the studied models for the AircraftCarrier dataset. (a) SqueezeNet, (b) ShuffleNetV2, (c) MobileNetV2, (d) VGGNet-16, (e) ResNet-50, (f) DenseNet-121. (Zoom in to see better).

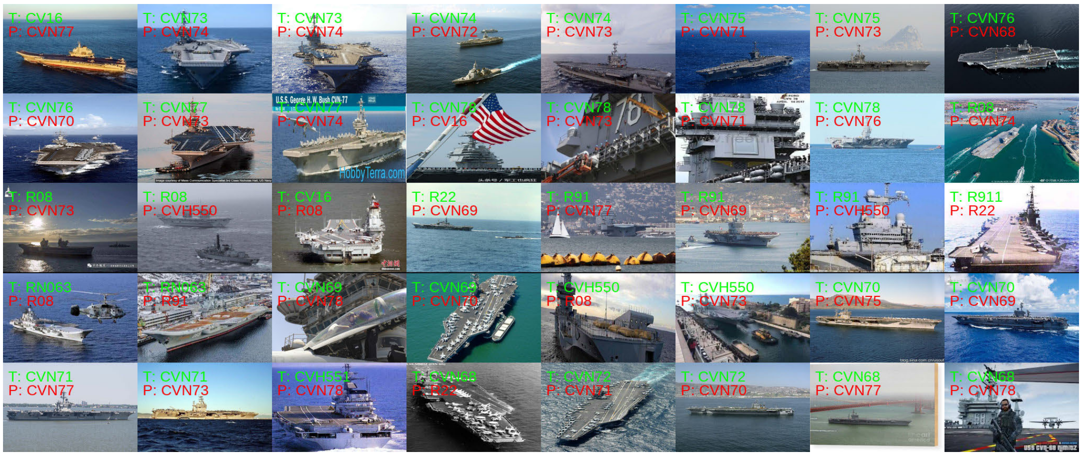

FIGURE 7. The images misclassified by ResNet-50, where T:XXX and P:XXX are the truth and predicted labels, respectively.

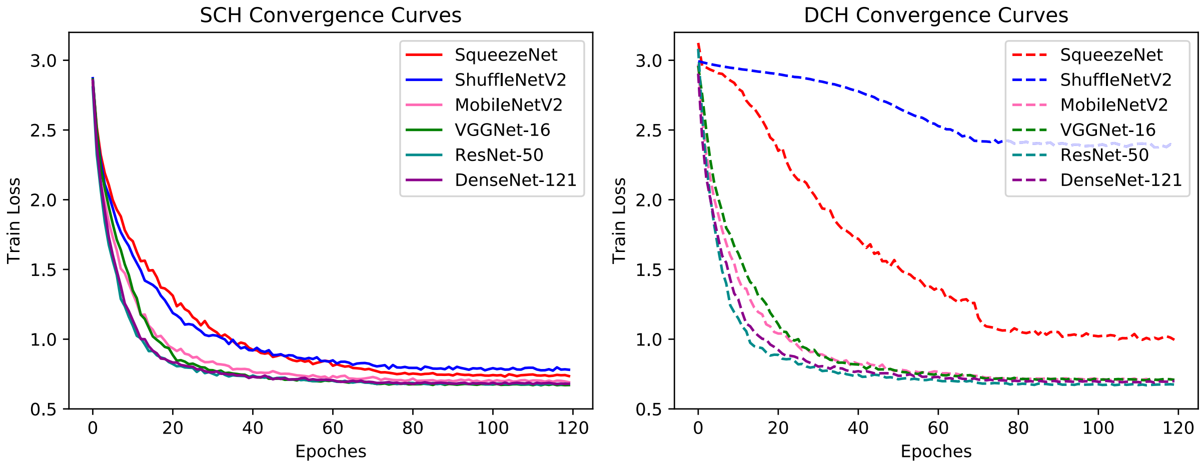

FIGURE 8. The convergence curves of training loss of SCH and DCH.

TABLE 4. The results of fine-grained classification methods on AircraftCarrier dataset. "Prec(%)" refers to the test accuracy.

Methods	Backbone	Prec(%)
SCH(our)	ResNet-50	63.4
B-CNN(PAMI2018) [21]	VGG-16	64.67
DFL-CNN(CVPR2018) [12]	ResNet-50	64.32
WS-DAN(ArXiv2019) [19]	ResNet-50	64.15
DCL(CVPR2019) [9]	ResNet-50	65.03
CrossX(ICCV2019) [3]	ResNet-50	64.50

From Table 4, we can find that these fine-grained image classification methods have better accuracy. At the same time, we analyzed these models, and found that these methods are basically based on ResNet-50 backbone, but they improve the model effect by using metric learning [19], [21], additional branches [12], adversarial generative learning [9], attention mechanism [3]. The performance of our unified SCH is worse than those of these methods. However, it is still meaningful to provide these evaluation results for researchers, because the original intention of our research is to provide a benchmark performance.

VI. DISCUSSION

In practice, object recognition can frequently be considered as instance-level image classification. However, due to the limited differences at the instance-level, it is difficult to distinguish the target objects. For example, the biggest difference between aircraft carrier instances lies in the hull number, tower and runway. Thus, an effective image object classification method is required for instance objects.

Based on the existing methods (e.g. fine-grained image classification), the following directions can be further explored:

- Numerous fine-grained image classification methods have attempted to locate and utilize differential regions based on part detectors, such as Part R-CNN [17] and HSnet Search [18] used in fine-grained classification. The differences in instance-level images categories can also be attributed to the local regions, e.g. hull number region. How to find the discriminative local regions can be explored in future.
- Recently, visual attention mechanisms have been applied to the classification, detection and segmentation

of images. Such mechanisms are able to pay more attention to the important regions or channels of the feature maps extracted by convolutional neural networks, such as WS-DAN [19] and TASN [20] used in fine-grained classification. Hence, attention mechanisms can be transferred to instance-level classification techniques.

- Deep convolutional neural networks have a strong feature representation ability, which allows for the application of metric learning (e.g. triplet loss in MMLN [4] and CDML [35]) to extract discriminant features by compressing the distance of intra-category features, and to expand the distance of the extra-category features.

VII. CONCLUSION

In the current study, a new instance-level object classification task is proposed, and a new aircraft carrier instance-level classification dataset is presented. We detail the collection method, characteristics and usage of our AircraftCarrier dataset. The publication of our dataset can promote research into instance-level image classification. At the same time, we propose the unified SCH method, which applies the classical convolutional neural network model as the backbone, to analyze the dataset. This provides a comparison baseline for future research. Finally, instance-level image classification is a challenging image classification task, thus more large-scale datasets and pointed methods are required.

ACKNOWLEDGMENT

The authors would like to thank TopEdit (www.topeditsci.com) for the English language editing of this manuscript.

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.

[2] M. Tan and V. Quoc Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. ICML*, 2019, pp. 6105–6114.

[3] 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.

[4] J. Wang, Y. Li, Z. Miao, X. Zhao, and Z. Rui, "Multi-level metric learning network for fine-grained classification," *IEEE Access*, vol. 7, pp. 166390–166397, 2019.

[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. ICLR*, 2015, pp. 1–14.

[6] 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.

[7] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2261–2269.

[8] Z. Yang, T. Luo, D. Wang, Z. Hu, J. Gao, and L. Wang, "Learning to navigate for fine-grained classification," in *Proc. ECCV*, 2018, pp. 438–454.

[9] 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.

[10] 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. 4476–4484.

[11] 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. 5219–5227.

[12] 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.

[13] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 815–823.

[14] J. Wang, Y. Li, S. Jiao, Z. Miao, and R. Zhang, "Grafted network for person re-identification," *Signal Process., Image Commun.*, vol. 80, Feb. 2020, Art. no. 115674.

[15] G. Van Horn, S. Branson, R. Farrell, S. Haber, J. Barry, P. Ipeirotis, P. Perona, and S. Belongie, "Building a bird recognition app and large scale dataset with citizen scientists: The fine print in fine-grained dataset collection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 595–604.

[16] G. Van Horn, O. Mac Aodha, Y. Song, Y. Cui, C. Sun, A. Shepard, H. Adam, P. Perona, and S. Belongie, "The iNaturalist species classification and detection dataset," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8769–8778.

[17] N. Zhang, J. Donahue, B. Ross Girshick, and T. Darrell, "Part-based R-CNNs for fine-grained category detection," in *Proc. ECCV*, 2014, pp. 834–849.

[18] M. Lam, B. Mahasseni, and S. Todorovic, "Fine-grained recognition as HSnet search for informative image parts," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6497–6506.

[19] T. Hu and H. Qi, "See better before looking closer: Weakly supervised data augmentation network for fine-grained visual classification," 2019, *arXiv:1901.09891*. [Online]. Available: https://arxiv.org/abs/1901.09891https://arxiv.org/abs/1901.09891

[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] T.-Y. Lin, A. RoyChowdhury, and S. Maji, "Bilinear convolutional neural networks for fine-grained visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 6, pp. 1309–1322, Jun. 2018.

[22] 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.

[23] C. Yu, X. Zhao, Q. Zheng, P. Zhang, and X. You, "Hierarchical bilinear pooling for fine-grained visual recognition," in *Proc. ECCV*, 2018, pp. 595–610.

[24] M. Sun, Y. Yuan, F. Zhou, and E. Ding, "Multi-attention multi-class constraint for fine-grained image recognition" in *Proc. ECCV*, 2018, pp. 834–850.

[25] D. Held, S. Thrun, and S. Savarese, "Deep learning for single-view instance recognition," 2015, *arXiv:1507.08286*. [Online]. Available: https://arxiv.org/abs/1507.08286

[26] A. Singh, J. Sha, K. S. Narayan, T. Achim, and P. Abbeel, "BigBIRD: A large-scale 3D database of object instances," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 509–516.

[27] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[28] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2, Jun. 2006, pp. 2169–2178.

[29] M. Portaz, M. Kohl, J.-P. Chevallet, G. Quénot, and P. Mulhem, "Object instance identification with fully convolutional networks," *Multimedia Tools Appl.*, vol. 78, no. 3, pp. 2747–2764, Feb. 2019.

[30] N. F. Iandola, W. M. Moskewicz, K. Ashraf, S. Han, J. W. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size," 2016, *arXiv:1602.07360*. [Online]. Available: https://arxiv.org/abs/1602.07360

[31] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "Shufflenet V2: Practical guidelines for efficient CNN architecture design," in *Proc. ECCV*, 2018, pp. 122–138.

[32] M. Sandler, G. Andrew Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation," 2018, *arXiv:1801.04381*. [Online]. Available: https://arxiv.org/abs/1801.04381

[33] N. Srivastava, E. 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, no. 1, pp. 1929–1958, 2014.

[34] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.

[35] J. Zhao and Y. Peng, "Cost-sensitive deep metric learning for fine-grained image classification," in *Proc. Int. Conf. Multimedia Modeling*, 2018, pp. 130–141.

KAI KANG received the master's degree in science of military command from the PLA University of Science and Technology, Nanjing, China, in 2016. He is currently a Staff Officer of the Army Engineering University of PLA, Nanjing. His current research interests include military data analysis and the theory of command and control.

GANGMING PANG is currently pursuing the B.S. degree with the College of Command and Control Engineering, Army Engineering University of PLA, Nanjing, China. His current research interests include computer vision and deep learning.

XUN ZHAO received the B.S. degree from the Army Engineering University of PLA, Nanjing, China, in 2019, where he is currently pursuing the M.S. degree with the College of Command and Control Engineering. His current research interests focus on image classification and deep learning.

JIABAO WANG received the Ph.D. degree in computational intelligence from the PLA University of Science and Technology, Nanjing, China, in 2013. He is currently an Assistant Professor with the Army Engineering University of PLA, Nanjing. His current research interests include computer vision and machine learning.

YANG LI received the M.S. degree from the PLA University of Science and Technology, Nanjing, China, in 2010, and the Ph.D. degree from the Army Engineering University of PLA, Nanjing, China, in 2018. He is currently an Associate Professor with the Army Engineering University of PLA. His current research interests include computer vision, deep learning, and image processing.