Discriminative Suprasphere Embedding for Fine-Grained Visual Categorization

Shuo Ye, Qinmu Peng, Wenju Sun, Jiamiao Xu, Yu Wang, Xinge You, *Senior Member, IEEE*, and Yiu-Ming Cheung, *Fellow, IEEE*

***Abstract*— Despite the great success of the existing work in fine-grained visual categorization (FGVC), there are still several unsolved challenges, e.g., poor interpretation and vagueness contribution. To circumvent this drawback, motivated by the hypersphere embedding method, we propose a discriminative suprasphere embedding (DSE) framework, which can provide intuitive geometric interpretation and effectively extract discriminative features. Specifically, DSE consists of three modules. The first module is a suprasphere embedding (SE) block, which learns discriminative information by emphasizing weight and phase. The second module is a phase activation map (PAM) used to analyze the contribution of local descriptors to the suprasphere feature representation, which uniformly highlights the object region and exhibits remarkable object localization capability. The last module is a class contribution map (CCM), which quantitatively analyzes the network classification decision and provides insight into the domain knowledge about classified objects. Comprehensive experiments on three benchmark datasets demonstrate the effectiveness of our proposed method in comparison with state-of-the-art methods.**

***Index Terms*— Deep hypersphere embedding, discriminative localization, fine-grained visual categorization (FGVC), weakly supervised learning.**

I. Introduction

SAMPLES from different classes have a large variation in appearance, but ones from the subclasses [1], [2], [3], [4], [5] generally have a small divergence, which brings a great challenge to distinguishing samples from different subclasses. This is usually referred to as fine-grained visual categorization (FGVC). More extremely, samples from the intraclasses even possibly own large discrepancy than those from interclasses (see Fig. 1) due to different sample postures or camera viewpoints. To handle this problem, several works have been devoted to finding the discriminant part [6], [7], such as the head, body, legs, and so on, of a bird. To achieve accurate part detection, previous fine-grained works require manual labels (such as bounding boxes and part annotations) to train a discriminant part detection network. For example, region-convolutional neural networks (R-CNN)-based methods [8], [9] (such as part-RCNN [10], parts segmentation and alignment (PSA-CNN) [11], and multiple granularity convolutional neural networks (MG-CNN) [12]), which employed selective search [13] to generate multiple candidate patches, and then high overlap regions with part bounding box are selected for classifier training [14], [15]. Fully convolutional network (FCN) [16], [17], [18] is another method to capture discriminant parts, which obtain the approximate outline information of the samples through the semantic segmentation approach. While these bottom-up manners have witnessed empirical improvements in the algorithm performance, the resultant algorithms and methods may still fail to achieve satisfactory results in FGVC. One of the important reasons concerns the misalignment between manual labels and deep learning needed regions. In other words, such manually defined regions may not fit deep learning models [19].

Weakly supervised learning only uses label information, which can effectively avoid human subjective error [20], [21], [22], [23]. Since the manual information of the target is no longer used, locating the discriminative part is a challenging task. Current solutions can be roughly divided into two categories, learning the local and global features by employing the multistream network architecture [12], [24], [25], [26], [27], or aggregating sophisticated higher order statistics of convolutional features [19], [28], [29], [30], [31], [32], [33], [34]. The core of the former is to select those containing discriminant regions from the generated candidate boxes and zoomed-in view gradually. To avoid the influence of the background in the candidate boxes, a series of work using convolutional response is carried out, such as RA-CNN [35], WSCPM [36], and PA-CNN [37]. Although promising results have been reported, those methods involve alternating optimization and other training operations, which are computationally expensive. It hinders the networks to be trained end-to-end, resulting in information loss [38], [39], [40]. The latter represents an image as a pooled outer product of features derived from two CNNs and captures localized feature interactions in a translationally invariant manner. Although it is an end-to-end training framework, these methods heavily rely on intricate

Manuscript received 18 June 2021; revised 23 January 2022 and 24 May 2022; accepted 20 August 2022. Date of publication 15 September 2022; date of current version 5 April 2024. This work was supported in part by the Natural Science Foundation of China (NSFC) under Grant 61772220 and in part by the Key Research and Development Plan of Hubei Province under Grant 2020BAB027. *(Corresponding author: Qinmu Peng.)*

Shuo Ye, Wenju Sun, and Yu Wang are with the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan 430074, China.

Qinmu Peng and Xinge You are with the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: pqinmu@gmail.com).

Jiamiao Xu is with the Department of Deep Learning, Deeproute Company, Ltd., Shenzhen 518000, China.

Yiu-Ming Cheung is with the Department of Computer Science, Hong Kong Baptist University, Hong Kong, SAR, China (e-mail: ymc@comp.hkbu.edu.hk).

This article has supplementary downloadable material available at https://doi.org/10.1109/TNNLS.2022.3202534, provided by the authors.

Digital Object Identifier 10.1109/TNNLS.2022.3202534

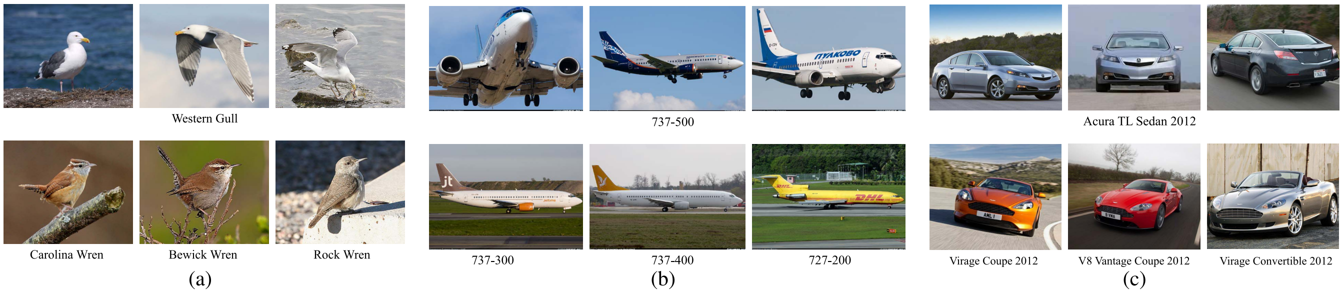


Fig. 1. Samples from three different datasets. The objects from different categories have a large difference in appearance, which provides discriminative clues to distinction. However, the same object in different images may have a strong appearance change due to the different postures or perspectives. The first row shows the large variances in the same subcategory, and the second row indicates the small variances among different subcategories. (a) CUB-200-2011. (b) FGVC-Aircraft. (c) Stanford Cars.

feature-encoding [41], which are less human-interpretable. Moreover, all the above-mentioned methods cannot give the quantitative contribution of the local part.

To address the limitations of poor interpretation and vagueness contribution, this article, therefore, proposes a novel discriminating suprasphere embedding (DSE) framework with three core modules, i.e., SE block module, phase activation map (PAM) module, and class contribution map (CCM) module. Among them, PAM can consistently highlight the important regions for predicted results. CCM can shed light on the domain knowledge learned by the network and quantitatively measure the contribution of each object part to the classification decision. To sum up, the contributions of this work are threefold.

1) The suprasphere model shows the feasibility to represent object features in weakly supervised FGVC, which enjoys efficient feature encoding with a clear geometric interpretation.
2) A novel weakly supervised FGVC framework is proposed, which contains two branches to localize the object and quantitative analysis of the discriminant area, respectively.
3) Comprehensive experiments on widely used benchmark datasets have been implemented, which justify that the proposed approach can achieve outperforming performance than other state-of-the-art approaches and baseline models.

The rest of this article is organized as follows. In Section II, we will introduce the related work including representation learning and discriminative localization for fine-grained visual recognition. We will describe the DSE framework and further discuss the unique benefits originating from its intuitive geometric interpretation in Section III. Experimental results will be presented in Section IV. Finally, Section V will draw a conclusion.

II. Related Work

In this section, we first present the work of weakly supervised fine-grained visual recognition. Then, the most relevant work on deep hypersphere embedding is briefly reviewed.

A. Weakly Supervised Fine-Grained Visual Categorization

Previous fine-grained works heavily rely on the object/part annotations in categorization. To effectively characterize subtle differences between fine-grained categories when only label information is involved, sophisticated deep learning models are designed. Specifically, a CNN that is pretrained from ImageNet is first used as an object detector to detect the object from each image. Then, part features are extracted from objects.

In [12], the family and genus information hidden under the species label was utilized, and the three classifiers were trained to capture the region of interest at the corresponding granularity. Xiao *et al.* [24] and Peng *et al.* [25] found that the proposal region obtained by pretrained CNN contained many noisy patches. For this reason, they filtered out noisy patches at first, then, retrained the network with the filtered data and extracted features for classification. In [42], each object is rough positioning and segmentation, and then takes it as the prior information of subsequent segmentation. This article is different from [12], [24], [25], and [42]; in that, we do not use a multistage approach to capture the discriminant part but propose an efficient end-to-end method to extract the most discriminative information for categorization.

Recently, high-order information-based weakly supervised algorithms demonstrate superior performance. The most typical application is bilinear CNNs (B-CNNs) [19], which uses two independent CNNs to capture the image local differences and takes the outer product over the convolutional outputs to grasp the second-order information. Several recent works [43] attempt to use multistream architecture to enhance the network learning capability which simultaneously learns both local and global features. However, their network convergence requires elaborate layer initialization. Furthermore, considering low-dimensional features and high-dimensional features that are interrelated and mutually reinforcing, the hierarchical bilinear pooling (HBP) model [34] has achieved better results. Although the above-mentioned methods are effective, it is difficult to understand why higher order features can promote the expression of features. In contrast, our method encodes semantic differences between subcategories by using the phase information of convolutional activations, which enjoys efficient feature encoding as well as human interpretability.

B. Deep Hypersphere Embedding

Deep hypersphere embedding [44] has been widely used in face recognition, which aims to make the maximum intraclass distance of extracted features less than their minimum

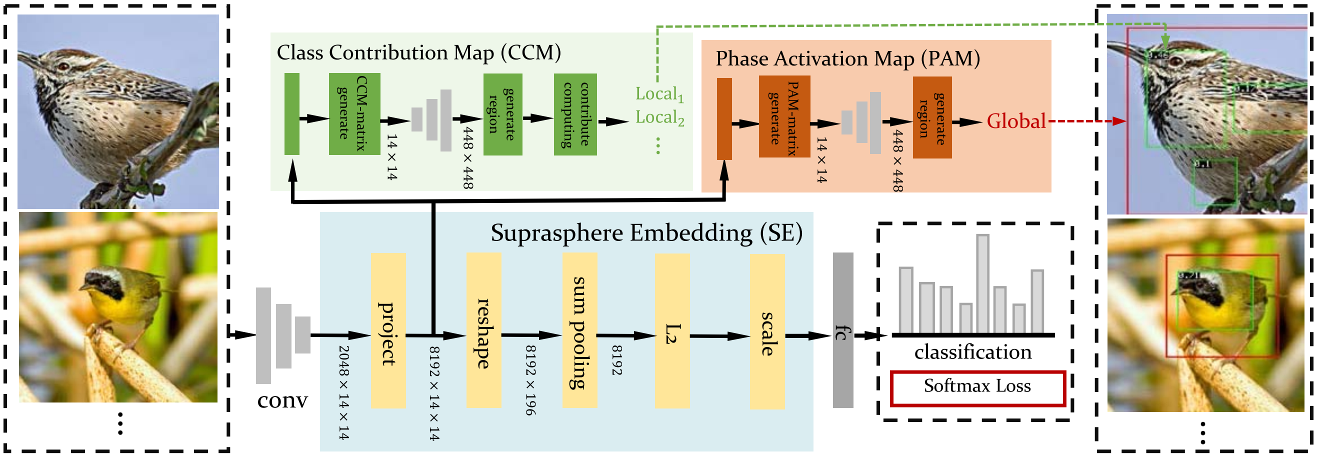


Fig. 2. Overview of the proposed DSE framework. The input image is fed into CNNs to get the feature map of the last convolutional layer. First, the feature is expanded into high-dimensional space to enhance its interclass separability, followed by sum pooling to obtain the compact feature representation, and then ℓ_2 normalization is deployed to weaken the intraclass variance and to output the unit suprasphere feature. Finally, a fully connected layer without bias is used to do the classification. The PAM and CCM obtained by combining the high-dimensional descriptors and the category attribute descriptors highlight the object region and the discriminative parts, respectively.

interclass distance. A simple example is given below to describe the general method. In binary classification, for a learned feature vector $\mathbf{x}$, a decision boundary can be obtained as

$$\mathbf{w}_1\mathbf{x} - \mathbf{w}_2\mathbf{x} = 0 \tag{1}$$

where $\mathbf{w}_i$ is the weight of the last fully connected layer corresponding to class i. Bias $\mathbf{b}$ is omitted to simplify the analysis. Equation (1) can be further formulated as

$$\|\mathbf{x}\|(\|\mathbf{w}_1\|\cos(\theta_{(\mathbf{w}_1,\mathbf{x})}) - \|\mathbf{w}_2\|\cos(\theta_{(\mathbf{w}_2,\mathbf{x})})) = 0 \tag{2}$$

where $\theta_{(\mathbf{w}_i,\mathbf{x})}$ is the angle between $\mathbf{w}_i$ and $\mathbf{x}$. It is clear that $\|\mathbf{w}_1\|\cos(\theta_{(\mathbf{w}_1,\mathbf{x})})$ and $\|\mathbf{w}_2\|\cos(\theta_{(\mathbf{w}_2,\mathbf{x})})$ determine the classification result [45]. Given $\mathbf{w}_1$ and $\mathbf{w}_2$, the predicted result only depends on the phase of $\mathbf{x}$. If we constrain $\|\mathbf{x}\|$ to be constant, all object attributes are encoded in the feature phase information [46]. As a result, the phase of $\mathbf{w}_i$ can be naturally regarded as an attribute descriptor of class i. The goal of training is then to make the phase of the learned feature as close as possible to the corresponding category attribute descriptor. Although the above-mentioned analysis is built on a binary-class case, its result can be generalized to the multiclass scenario. In general, forcing the model to use angle only to discriminate subclasses can effectively improve the performance but it can also increase the difficulty of training [44]. The larger the decision boundary, the more unstable the model will be. Different from the previous method, this article, therefore, normalizes the extracted features. The weights and phases are used to complete the classification, which reduces the difficulty of training and has high precision.

III. Proposed Method

In this section, we first detail the motivation and general idea of our proposed DSE and then discuss the unique benefits originating from its intuitive geometric interpretation.

A. Notation

Scalars are represented by lowercase letters (e.g., x), and vectors and matrices are denoted by the bold lowercase letter (e.g., $\mathbf{x}$) and the bold uppercase letter (e.g., $\mathbf{X}$), respectively. $\mathbf{X}^{\mathbf{T}}$ denotes the transpose of the matrix $\mathbf{X}$. Matrix norm, $\|\mathbf{X}\|_F = (\sum_i \sum_j |\mathbf{X}_{ij}|^2)^{1/2}$ is frequently used in this article, where $\mathbf{X}_{ij}$ denotes the element of $\mathbf{X}$ at the ith row and the jth column. $\|\cdot\|$ denotes the two-norm of vectors.

B. Suprasphere Embedding

The architecture of the proposed method is illustrated in Fig. 2. The goal of our method is to make the model lightweight and has an intuitive geometric interpretation. To this end, an improved metric method is proposed, which can effectively pull similar samples close and simultaneously push dissimilar samples apart from each other [47], [48], [49], [50], [51].

Given an input image $\mathbf{I}$, let the output feature map of last convolution layer be $\mathbf{X} \in \mathbb{R}^{c\times h\times w}$ with height h, width w, and channels c. The convolutional activations at a single-spatial location can be considered as a local descriptor. Therefore, we denote $\mathbf{X}$ as a set of descriptors $\{\mathbf{x}_p\}_{p\in\boldsymbol{\Omega}}$, where $\mathbf{x}_p \in \mathbb{R}^c$ represents the descriptor at a particular position p and $\boldsymbol{\Omega}$ is the set of spatial locations. To model the subtle variance of different subcategories, the descriptors are first mapped to a high-dimensional space to enhance their representation capability, which is given by

$$\mathbf{y}_p = \mathbf{U}^T\mathbf{x}_p \tag{3}$$

where $\mathbf{U} \in \mathbb{R}^{c\times d}$ is the projection matrix, and d is a user-defined projected feature dimension that can be adjusted according to the specific task. Geometrically, it explicitly increases the interclass separability between learned features. Then, we reshape the features, and the high-dimensional descriptors across the spatial locations are aggregated into a

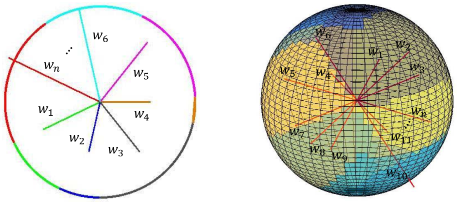


Fig. 3. Geometry Interpretation of suprasphere feature. The first is a 2-D feature constraint, and the second is a 3-D feature constraint. Each color represents a category. Features are mapped onto a hypersphere, since different subclasses have different weights, they are mapped inside or outside the hypersphere, which forms a suprasphere.

single-global feature by sum pooling

$$\mathbf{z} = \mathrm{Sum}P(\mathbf{Y}) = \sum_{p\in\Omega} \mathbf{y}_p. \tag{4}$$

Given the large variance in the same subcategory, the norm of the global feature is further constrained by ℓ_2 normalization, which can be computed as

$$\mathbf{f} = \frac{\alpha\mathbf{z}}{\|\mathbf{z}\|} = \frac{\alpha\sum_{p\in\Omega}\mathbf{y}_p}{\|\sum_{p\in\Omega}\mathbf{y}_p\|} \tag{5}$$

where $\mathbf{f} \in \mathbb{R}^d$ is the hypersphere feature. Geometrically, the learned features span on a hypersphere manifold, which encourages intraclass compactness. Moreover, object attributes are all encoded in the phase information of the hypersphere feature and our network can fully focus on optimizing the cosine distance between the learned feature and the corresponding category attribute descriptor, which is particularly useful for fine-grained visual recognition. The scale factor α is a model parameter that is learned through network training. Note that α is crucial here because it adaptively controls the gradient in backpropagation. Experimentally, α significantly improves the performance and promotes network convergence. Finally, an inner product classifier is used to obtain the classification score

$$S = \mathbf{w}\cdot\mathbf{f} \tag{6}$$

where $\mathbf{w} \in \mathbb{R}^{m\times d}$ and m is the number of categories.

Note that, unlike the previous methods [44], [45], [46], [49], we do not normalize $\mathbf{w}$, but normalize $\mathbf{f}$, as illustrated in Fig. 3. In other words, the suprasphere is the projection of the weights. This brings an advantage, even if the phase difference between similar subclasses is small, the weight is located at different positions in the suprasphere, which can also provide a basis for discrimination.

During the whole training process, a softmax-based loss is used. We remove the bias term and use $\mathbf{w^T x}_i = s\cos\theta_i$ to transform it as

$$L = -\log P_{y_i} = -\log\frac{e^{s\cos\theta_{y_i}}}{e^{s\cos\theta_{y_i}} + \sum_{j=1,j\neq y_i}^{C} e^{s\cos\theta_{y_i}}} \tag{7}$$

where $\mathbf{x_i}$ denotes the embedding feature of the ith training image, and $\mathbf{y_i}$ is the label of $\mathbf{x_i}$. P_{y_i} is the predicted probability of assigning $\mathbf{x_i}$ to class $\mathbf{y_i}$. C is the number of identities. Our goal is to minimize this loss function.

Our SE module is lightweight and effective. It can be easily embedded into the existing backbone networks and involves only a single-stage end-to-end training process.

C. Phase Activation Map

Analysis of the impact of discriminant regions on classification is of great significance for understanding the decision-making of fine-grained models [52], [53], [54]. Zeiler and Fergus [52] uses deconvolution to locate the target and realize the analysis of the discriminant region. However, it uses fully connected layers at the end, which ignores the spatial location information in the feature map. Class activation mapping (CAM) [53] is much more reasonable. It uses global average pooling to identify the entire object area and utilizes the weighted sum of convolutional feature maps to highlight the discriminative regions. But it only provides a qualitative analysis of the corresponding network, and we can only judge whether the results are reliable by feeling. The intuitive geometric interpretation of the DSE framework provides unique benefits for exploring classified objects. We will explain in the following how to achieve a quantitative analysis of parts contributions.

Equation (6) can be expanded as follows:

$$\begin{aligned} S &= \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1d} \\ w_{21} & w_{22} & \cdots & w_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ w_{m1} & w_{m2} & \cdots & w_{md} \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_d \end{bmatrix} \\ &= \begin{bmatrix} \alpha\|\mathbf{w}_1\|\cos\left(\theta_{(\mathbf{w}_1,\mathbf{f})}\right) \\ \alpha\|\mathbf{w}_2\|\cos\left(\theta_{(\mathbf{w}_2,\mathbf{f})}\right) \\ \vdots \\ \alpha\|\mathbf{w}_m\|\cos\left(\theta_{(\mathbf{w}_m,\mathbf{f})}\right) \end{bmatrix} \end{aligned} \tag{8}$$

where $\mathbf{w}_i$ denotes the ith row vector in the matrix $\mathbf{w}$. Notably, for a given hypersphere feature $\mathbf{f}$, the predicted score for class i is only determined by $\mathbf{w}_i$. Therefore, each row vector in the matrix $\mathbf{w}$ can be naturally regarded as an attribute descriptor of the corresponding subcategory. Besides, all the semantic differences, as stated earlier, are encoded in the phase information of the suprasphere feature. Based on the above-mentioned discussion, we proposed PAM to measure the contribution of local descriptors to the phase of suprasphere features. It is defined as

$$\mathrm{PAM}_p = \begin{cases} \|\mathbf{y}_p\|^{\beta}\cos\left(\theta_{(\mathbf{y}_p,\mathbf{w}_i)}\right)^{\gamma}, & \cos\left(\theta_{(\mathbf{y}_p,\mathbf{w}_i)}\right) > 0 \\ 0, & \cos\left(\theta_{(\mathbf{y}_p,\mathbf{w}_i)}\right) \le 0 \end{cases} \tag{9}$$

where $\mathbf{y}_p$ is the high-dimensional descriptor and $\mathbf{w}_i$ is the attribute descriptor of class i. β, γ are the tunable parameters. Equation (9) is intuitive since the hypersphere feature is determined by the norm and phase of the high-dimensional descriptors. In this process, a threshold is set and we calculate the maximum connected domain of the part above the threshold and then mark its bounding box.

D. Class Contribution Map

Moreover, both sum pooling and normalization do not change the phase of the high-dimensional descriptors at each spatial location. Thus, we can reformulate the predicted score in (8) into a more intuitive form

$$\begin{aligned}\alpha\|\mathbf{w}_i\|\cos\left(\theta_{(\mathbf{w}_i,\mathbf{f})}\right) &= \alpha\|\mathbf{w}_i\|\cos\left(\theta_{\left(\mathbf{w}_i,\frac{\alpha\sum_{p\in\Omega}\mathbf{y}_p}{\|\sum_{p\in\Omega}\mathbf{y}_p\|}\right)}\right)\\ &= \alpha\|\mathbf{w}_i\|\cos\left(\theta_{\left(\mathbf{w}_i,\sum_{p\in\Omega}\mathbf{y}_p\right)}\right)\\ &= \alpha\frac{\sum_{p\in\Omega}\mathbf{y}_p\mathbf{w}_i}{\|\sum_{p\in\Omega}\mathbf{y}_p\|}. \qquad (10)\end{aligned}$$

Equation (10) indicates that the contribution value of local descriptors to the prediction score can be specifically calculated. Without loss of generality, we define the contribution rate of location p to the predicted score of class i as

$$\mathrm{CCM}_p = \frac{\mathbf{y}_p\mathbf{w}_i}{\sum_{p\in\Omega}\mathbf{y}_p\mathbf{w}_i}. \qquad (11)$$

In an intuitive understanding, during this process, a threshold is set to capture important areas of the image. Different from PAM, we record its several connected domains, sum up all values of any connected domain, and define its local contribution as the value of the current and the whole image. An interesting finding is that some regions receive negative contribution rates, and this may prove that manually defined regions are not necessarily optimal for machine classification. CCM intuitively explains the network classification decision. In other words, our method can not only identify exactly which object parts are being used for classification but also figure out their contribution rates to the predicted subcategory in an intuitive, straightforward, and efficient manner.

In summary, our model can not only automatically recognize and localize the fine-grained objects but also show users how to identify subordinate categories by highlighting discriminative parts. For humans, this is very difficult because it involves expert knowledge about classified objects. Apparently, our DSE framework performs well without the participation of auxiliary annotations.

IV. Experiments

We conduct comprehensive experiments in this section to demonstrate the effectiveness of our method. Particularly, we describe benchmark datasets in Section IV-A. Next, we demonstrate the details and configuration of the model in Sections IV-B and IV-C. We present the classification results in Section IV-D. Finally, in Section IV-E, we provide discriminative localization and visualization which shed light on the domain knowledge learned by the network.

A. Datasets

Exeriments are conducted on three challenging datasets, including Caltech-UCSD Birds (CUB)-200-2011 [55], Stanford Cars [56], and FGVC-Aircraft [57], which have been widely used to evaluate FGVC methods. All datasets provide fixed train and test splits. The details are as follows.

CUB-200-2011 is the most widely used dataset for FGVC. It includes 11 788 images from 200 different bird species. Images are split into the training and test sets with 5994 images for training and 5794 images for testing. Each image in this dataset is associated with detailed annotations including image-level labels, object bounding boxes, part locations, and binary attributes.

FGVC-Aircraft contains 10 000 images from 100 aircraft variants, which are split into training, validation, and testing sets. And each image is provided with a bounding box annotation and an image-level class label. Unlike the CUB-200-2011 dataset, some aircraft variants have extremely subtle differences that only can be distinguished by the number of windows in the model. Thus, discriminative part localization would play a more significant role here.

Stanford Cars consists of 16 185 images from 196 classes of cars, which are divided into 8144 images for training and 8041 images for testing. Compared with CUB-200-2011, the background of cars is relatively clear. However, there are cases of adding equipment or changing the appearance color of the same car. This requires the network could capture the most essential features of the current category.

Note that, in all our experiments, we only use the category label without any part or bounding box annotation provided by the datasets. We adopt top-1 accuracy as the evaluation metric to comprehensively evaluate the classification performance of our DSE-based FGVC method and baseline methods.

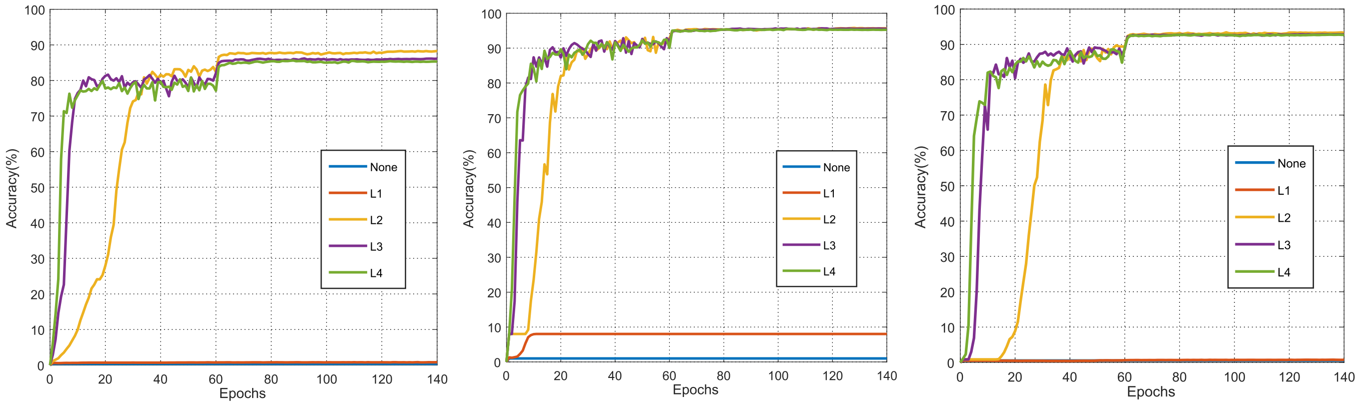
B. Implementation Details

To make a fair comparison with the SOTA methods, we implement the proposed DSE method based on the widely used CNN model ResNet-101 [58], which is pretrained on ImageNet [59] dataset. We remove the last average pooled and fully connected layers and insert the DSE module. For all three datasets [60], the DSE takes images with a size of 448 × 448 pixel. Our preprocessing follows the commonly used configurations. Specifically, random sampling and horizontal flipping are adopted for data augmentation in the training phase and center cropping is only applied during inference. For the sake of faster convergence and better performance, the original ResNet layers are initialized from the pretrained model and the scale factor α is initialized as 1. The other layers are randomly initialized. During training, we directly optimize the whole network using stochastic gradient descent [61] with a batch size of 16, momentum of 0.9, weight decay of 10^{-4}, and learning rate of 10^{-3}, periodically annealed by 0.1. Our implementation is based on Torch [62] framework with a Titan X GPU.

C. Model Configuration

Model configuration experiments are conducted on the CUB-200-2011 dataset to verify the validity of the individual component and to determine the optimal hyperparameters.

1) Feature Projection: To verify the effectiveness of feature projection and investigate the effect of projected dimension d, we conduct extensive experiments on d and the results are given in Table I. As expected, feature projection significantly enhances the network representation capability, which might be explained by the fact that CNNs output activations are all positive, which greatly constrains the range of the hypersphere feature space. Besides, we observe that increasing d leads



(a) (b) (c)

Fig. 4. Accuracies of different normalization methods on the three datasets using the Resnet-101 as backbone. The blue line is the result of the network without regularization and the red, yellow, purple, and green lines show the network training with ℓ_1, ℓ_2, ℓ_3, and ℓ_4 regularization, respectively. (a) CUB-200-2011. (b) FGVC-Aircraft. (c) Stanford Cars.

TABLE I

EXPERIMENTAL RESULTS USING VARIED PROJECTION DIMENSIONS. "W/O" DENOTES "WITHOUT"

d	w/o	512	1024	2048	4096	8192	16384
Accuracy(%)	85.0	87.5	87.7	88.0	88.1	88.4	88.0

TABLE II

EXPERIMENTAL RESULTS USING DIFFERENT NORMALIZATION METHODS. "W/O" DENOTES "WITHOUT." NOTE THAT ACCURACY WITH "-" REPRESENTS THE NETWORK FAILED TO CONVERGE

Normalization	w/o	ℓ_1	ℓ_2	ℓ_3	ℓ_4
Accuracy(%)	-	-	88.4	86.0	85.7

to higher accuracy. However, the performance slightly drops when the projected d increases from 8192 to 16 384, which infers that the 8192-d hypersphere feature is saturated in comprehensively encoding object attributes. Therefore, we use $d = 8192$ in all the following experiments for its ideal balance between computational complexity and accuracy.

2) *Normalization:* To analyze the effect of the feature normalization, we conduct experiments using different normalization methods. The results are summarized in Table II. It can be seen that the network without feature normalization cannot converge. To further illustrate the universality of this conclusion, we visualize the results of different regularizations over three datasets in Fig. 4. The results indicate that feature normalization helps to enhance network stability. It is worth noting that ℓ_3 and ℓ_4 regularization can make our model converge faster, but ℓ_2 regularization is more accurate in the CUB dataset. One possible reason considers the visual differences between aircraft and cars often come from a perspective, while the differences in birds come from not only perspective but also posture, such as resting and flying [63], and ℓ_2 regularization can better depict such changes in detail. So the ℓ_2 regularization is used in the following experiments.

3) *Scale Factor* (**α**)*:* Scale transformation is used to expand the space of feature expression [46] and accelerate network convergence. The reason is that the denominator in the fraction of the ℓ_2 layer is relatively large, which seriously hinders the gradient backpropagation. α can be used in two methods, including fixing as a constant and learning during end-to-end training. The results are given in Table III. Learning the scale factor during end-to-end training can adaptively optimize the network. In light of this, we let the scale factor be learned through backpropagation in all the following experiments.

TABLE III

EXPERIMENTAL RESULTS WITH DIFFERENT α

α	1	5	10	15	20	25	learn
Accuracy(%)	47.7	86.2	87.7	87.5	87.5	87.4	88.4

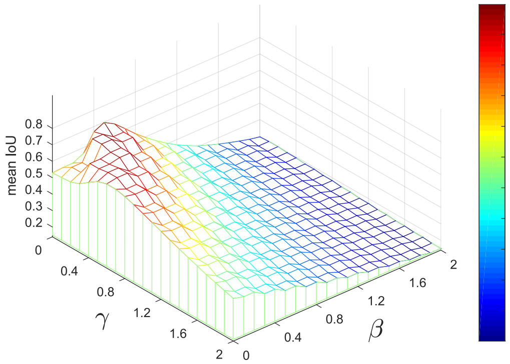


Fig. 5. Mean IoU with different value of β and γ.

4) **β** *and* **γ***:* Experiments to evaluate the sensitivity of hyperparameters β and γ on the classification performance of the PAM are conducted. β and γ are in the range of [0, 0.2, 0.4, 0.6, 0.8, 1, 1.2, 1.4, 1.6, 1.8, 2]. Experimental results are shown in Fig. 5. As can be seen, the performance increases first and then decreases when β or γ increases. The best performance is achieved at $\beta = 0.4$ and $\gamma = 0.1$. It is worth noting that the results are poor when $\beta = 0$, but when $\gamma = 0$, the results are best, which indicates that β is more important. Therefore, the norm of the high-dimensional descriptor is more important for the construction of the suprasphere feature phase.

TABLE IV
COMPARISON IN TERMS OF CLASSIFICATION ACCURACY ON THE CUB-200-2011, FGVC-AIRCRAFT, AND STANFORD CARS DATASET

Method	Accuracy(%)		
	CUB	Aircraft	Cars
FCAN [65]	84.7	-	91.3
DeepLAC [15]	80.3	-	-
Mask-CNN [17]	85.4	-	-
B-CNN [19]	84.1	86.6	91.3
LRBP [32]	84.2	87.3	90.9
HBP [34]	87.1	90.3	93.7
HIHCA [28]	85.3	88.3	91.7
GP [33]	85.8	89.8	92.8
DeepKSPD [30]	86.5	91.5	93.2
DFL-CNN [43]	87.4	91.7	93.1
RA-CNN [35]	85.3	-	92.5
MA-CNN [66]	86.5	89.9	92.8
P-CNN [27]	87.3	90.6	93.3
PA-CNN [37]	87.8	91.0	93.3
MAMC [67]	86.5	-	93.0
MCL [68]	87.3	92.6	-
MOMN [69]	88.3	90.3	93.2
MSEC [70]	88.3	92.4	-
DSE(OURS)	**88.4**	**93.2**	**93.8**

It is reasonable since the larger the norm of the descriptor, the higher the contribution to the suprasphere feature phase information. Nevertheless, combining the norm and phase of the high-dimensional descriptor can achieve the best result.

D. Performance Evaluation

The results of our proposed DSE method and the comparisons with the SOTA methods are reported in Table IV. It can be seen that DSE outperforms most of the existing methods on all three datasets. The detailed analysis of these three datasets is as follows. In the CUB-200-2011 dataset, the three earlier methods, FCAN, Deep LAC, and Mask-CNN, fail to provide satisfactory classification results, although they utilize additional annotations of the bounding box and parts to learn discriminative features. One possible reason is that manually defined regions are not necessarily optimal for machine classification. In contrast, our proposed DSE is trained without any additional annotations but outperforms them with a large margin. Furthermore, we analyze the part localization methods, such as RA-CNN and MA-CNN. Although they are annotation free, the accuracy of those methods is also beyond previous methods. But compared with our results, there is still a gap. One possible reason is that those methods need to generate a large number of region proposals at first, but most of the regions inevitably contain environmental noise. Moreover, those approaches involve complex alternate optimization or multistage training strategies, which may cause information loss. Compared with bilinear-based methods, such as the B-CNN [19], low-rank bilinear pooling (LRBP) [32], and HBP [34], our method does not use the second-order information and still achieves a modest accuracy improvement of 1.3%. In the Aircraft and Stanford Cars datasets, the proposed DSE also achieves excellent performance, while we also have the same observation as on the CUB-200-2011 dataset. We still get competitive performance against the latest methods [68], [69], [70]. Compared with the best results among them, we achieved an improvement of 0.1%, 0.6%, and 0.6% in CUB, Aircraft, and Cars, respectively. Specifically, compared with mutual-channel loss (MCL) [68], our DSE improves 1.1% and 0.6% in CUB and Aircraft datasets, respectively. MCL designed two branches before full connections to implement class-aligned channel profiles and calculates the similarity of all channels. However, this alignment may be more difficult for birds, which have wide attitudinal differences. For multi-objective matrix normalization (MOMN) [69], we achieved an improvement of 0.1%, 2.9%, and 0.6% in CUB, Aircraft, and Cars, respectively. For multi-scale erasure and confusion (MSEC) [70], we achieved an improvement of 0.1% and 0.8% in CUB and Aircraft, respectively. MSEC divides and confuses the subregions with higher confidence scores to generate an image with multiscale information. However, there is a risk of ignoring the structural relationships between feature regions. These results further validate the effectiveness and efficiency of our method. Specifically, our method outperforms most of the baseline methods with a large margin, which indicates that DSE is particularly efficient to encode the subtle semantic differences between subcategories using feature phase information. Apart from the above-mentioned quantitative evaluation, we present some qualitative results by visualizing the feature distributions in the 2-D space in Fig. 6 to show the superiority of suprasphere features, where the dots with the same color denote the same categories. It can be seen that the clutter original feature becomes compact after suprasphere mapping. This reflects that the SE method tends to capture the essence of the corresponding categories and maintain better distinguishing capacity. We also calculated the space utilization of the model (interested readers can find more information in [51]). It can be seen that DSE demonstrates the highest utilization of hypersphere space on the Cars dataset with the highest accuracy, which is consistent with our expectations.

E. Discriminative Localization and Visualization

The parts of some individual birds, aircraft, and car examples located by the proposed part localization network are shown in Fig. 7. It can be seen that despite the birds appearing in different poses and viewpoints with a cluttered background, our PAM and CCM can learn discriminative part detectors to consistently localize the parts of the head, breast, wing, and feet. For aircraft and car categories, consistently discriminative part areas are also successfully detected. This is mainly because both sum pooling and normalization do not change the phase of the high-dimensional descriptors at each spatial location, and the end-to-end training method does not lose discriminative information. We further evaluate the object localization performance in the context of fine-grained visual analysis. For a fair comparison, we follow the same experimental settings with CAM [53] that bounding boxes are constructed by choosing the optimal threshold and returning

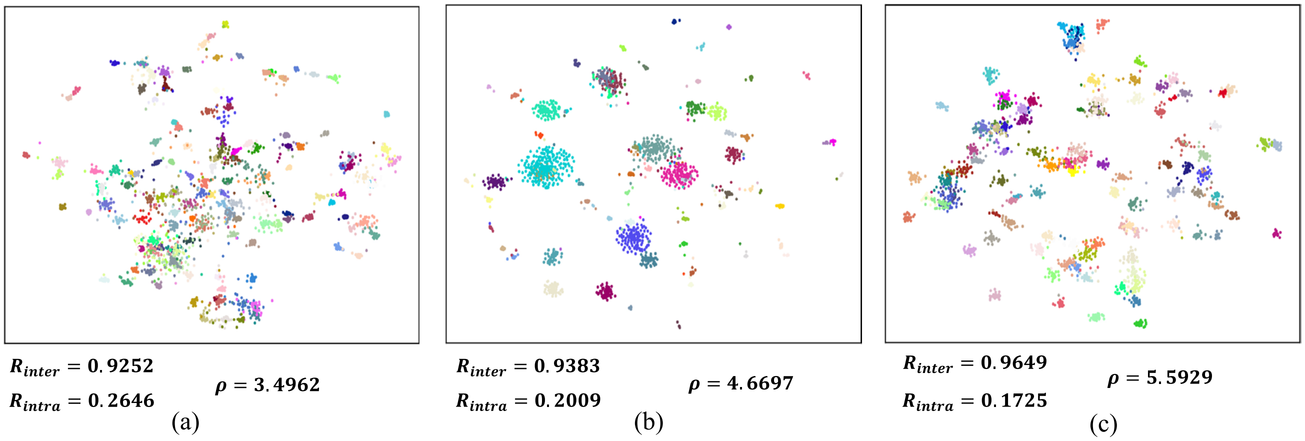


Fig. 6. Visualization of the category distribution by SE mapping in 2-D space with t-distributed stochastic neighbor embedding (t-SNE) [64]. Each dot denotes a sample and different colors represent different categories. In each dataset, several classes are randomly selected for visualization. In (a), 140 classes are selected, in (b), 67 classes are selected, and in (c), 97 classes are selected. R_{inter} and R_{intra} indicate the degree of compactness of interclass and intraclass, respectively. ρ means space utilization [51]. (a) CUB-200-2011. (b) FGVC-Aircraft. (c) Stanford Cars.

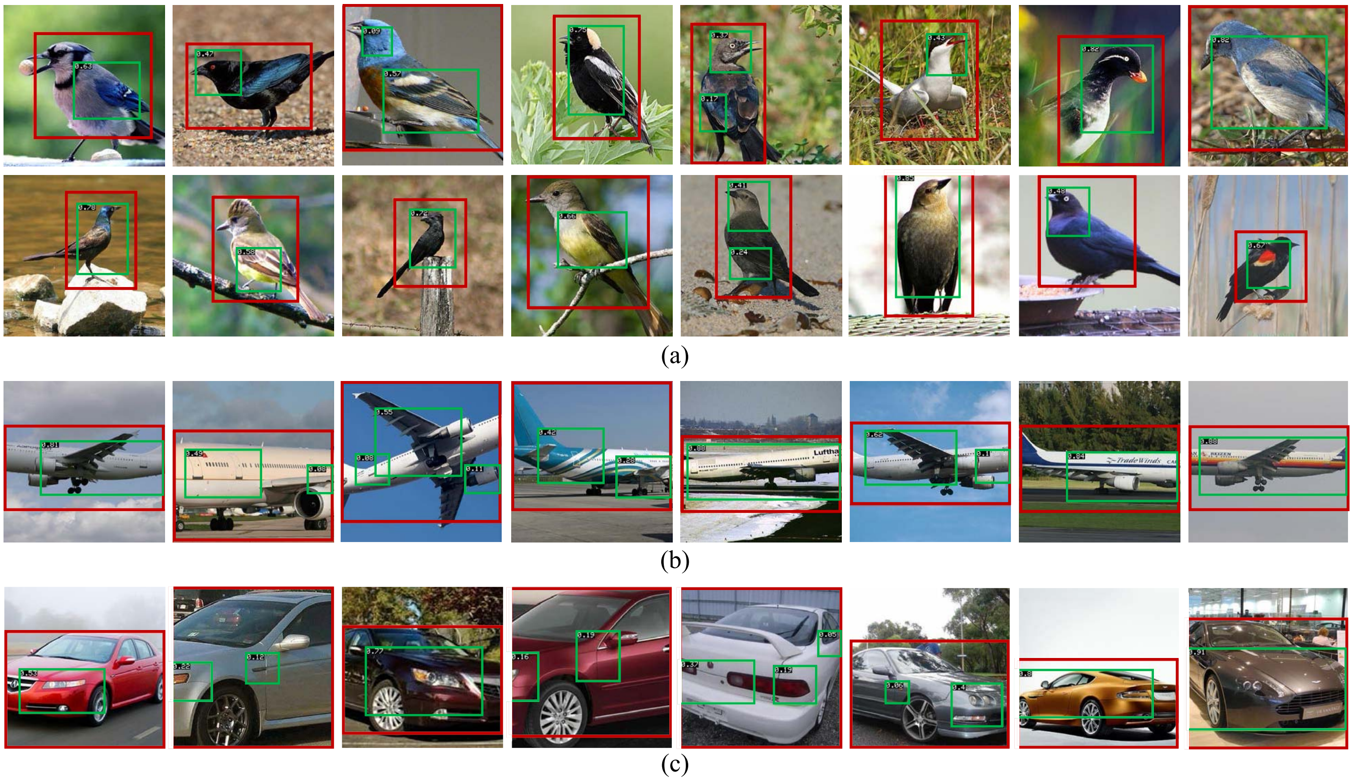


Fig. 7. Part localization results on the CUB-200-2011 dataset, FGVC-Aircraft dataset, and the Stanford Cars dataset. The part detectors can localize consistently discriminative parts on all these three datasets. (a) CUB-200-2011. (b) FGVC-Aircraft. (c) Stanford Cars.

the largest box. We then calculate the mean intersection-over-union (IoU) of the predicted box with the provided object bounding box. The results are summarized in Table V, and an average 4.5% improvement is achieved across three different datasets, which demonstrates its localization capability.

To further demonstrate how PAM and CCM quantitatively analyze the network classification decision. We visualize some examples in the three datasets. Fig. 8 shows that PAM consistently highlights the object region and CCM highlights the discriminative parts. The discriminative localization results intuitively explain the network classification decision. Let us take Fig. 8(a) as an example, which contains two subclasses with large changes within subclasses and similar appearances between subclasses. It can be seen that our network tends to focus primarily on specific discriminative parts, such as the crown of Sparrow, the leg of Tern, and the beak of Kittiwake. Such observations exactly reflect the nature of our approach which aims to learn the discriminative object attributes by enhancing the interclass variance and weakening the intraclass variance. Similar conclusions are also observed

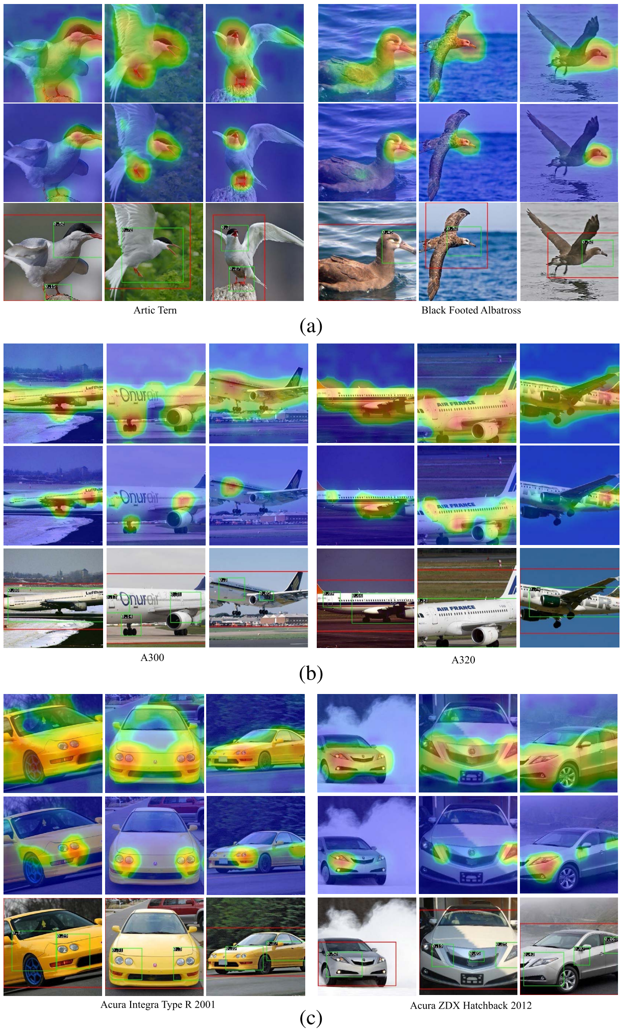


Fig. 8. Visualization of three datasets. For each image, PAM, CCM, localization result, and the predicted subcategory are presented from top to bottom. The red box indicates the object localization and the green box indicates the discriminative localization. It can be seen that PAM consistently highlights the object region and CCM highlights the discriminative parts. (a) CUB-200-2011. (b) FGVC-Aircraft. (c) Stanford Cars.

in other datasets and a vivid demonstration is available in the Supplementary Material. In summary, PAM and CCM show that our network can not only identify subclasses but also locate objects and quantitatively evaluate the contribution of the discerning region. This is attractive for use in real-world scenarios and useful for guiding further improvements for fine-grained visual analysis in the future.

Finally, we analyze the complexity of the DSE model. Two widely used indicators are adopted, including Params and FLOPs, which represent the total number of parameters to be trained in the model and the number of floating point operations (i.e., the theoretical calculation amount of the model) during inference, respectively. We report our results in Table VI and compare them with some classical models.

For a fair comparison, all models in Table VI use ResNet-50 as the backbone. As can be seen, the number of parameters in DSE is significantly lower than the B-CNN and HBP models and has roughly the same as the latest MOMN model. This is attributed to the design of our supersphere embedding process does not require too many parameters to control. For FLOPs, DSE is significantly lower than the latest model MOMN, and the result is roughly the same as BCNN. An important reason for the high FLOPs of HBP and MOMN is the design of interactive operations between the low-level and mid-level. Although this can improve the accuracy of the model, writing increases the cost of calculation.

TABLE V

Comparison in Terms of Mean IoU on the CUB-200-2011, FGVC-Aircraft, and Stanford Cars Datasets

Method	mean IoU		
	CUB-200-2011	FGVC-Aircraft	Stanford Cars
CAM [53]	0.674	0.739	0.815
PAM	0.706	0.801	0.855

TABLE VI

Model Complexity Analysis

Model	$Params(M)$	$FLOPs(G)$
B-CNN [19]	76.99	16.70
HBP [34]	76.37	26.31
MOMN [69]	30.62	25.35
DSE(OURS)	41.93	19.93

V. Conclusion

In this article, we have presented the DSE for FGVC, which does not need bounding box/part annotation for training and only involves a single-stage end-to-end optimization. Extensive experiments are conducted on three real-world datasets. Compared with the SOTA methods, our method can effectively promote CNN to learn discriminative suprasphere features, thus significantly improving the classification accuracy. Moreover, our method also has an intuitive geometric interpretation as well against all other competitors. In the future, we will extend this work in two directions. First, PAM and CCM can be used for data augmentation to further improve classification accuracy. Second, we will explore how to integrate suprasphere learning into the feature extraction stage to enhance the effect of phase encoding. In the FLOPs category, DSE also has a lower computation time than the examples listed. Note that visual modules for quantitative analysis of discriminant regions are also integrated into the DSE. The result is consistent with our lightweight design philosophy.

Acknowledgment

Codes are available at: https://github.com/wywd/DSE.

References

[1] X.-S. Wei, Q. Cui, L. Yang, P. Wang, and L. Liu, "RPC: A large-scale retail product checkout dataset," 2019, *arXiv:1901.07249*.

[2] M. Merler, C. Galleguillos, and S. Belongie, "Recognizing groceries *in situ* using *in vitro* training data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8.

[3] P. Jund, N. Abdo, A. Eitel, and W. Burgard, "The freiburg groceries dataset," 2016, *arXiv:1611.05799*.

[4] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, "DeepFashion: Powering robust clothes recognition and retrieval with rich annotations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1096–1104.

[5] C. Zhang, C. Liang, L. Li, J. Liu, Q. Huang, and Q. Tian, "Fine-grained image classification via low-rank sparse coding with general and class-specific codebooks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 7, pp. 1550–1559, Jul. 2017.

[6] Z. Li, Z. Zhang, J. Qin, Z. Zhang, and L. Shao, "Discriminative Fisher embedding dictionary learning algorithm for object recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 3, pp. 786–800, Mar. 2020.

[7] X. Chen *et al.*, "Sample balancing for deep learning-based visual recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 10, pp. 3962–3976, Oct. 2020.

[8] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[9] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[10] N. Zhang, J. Donahue, R. Girshick, and T. Darrell, "Part-based R-CNNs for fine-grained category detection," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 834–849.

[11] J. Krause, H. Jin, J. Yang, and L. Fei-Fei, "Fine-grained recognition without part annotations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5546–5555.

[12] D. Wang, Z. Shen, J. Shao, W. Zhang, X. Xue, and Z. Zhang, "Multiple granularity descriptors for fine-grained categorization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2399–2406.

[13] J. Uijlings, K. van de Sande, and T. Gevers, "Selective search for object recognition," *Int. J. Comput. Vis.*, vol. 104, pp. 154–171, Oct. 2013.

[14] S. Branson, G. Van Horn, S. Belongie, and P. Perona, "Bird species categorization using pose normalized deep convolutional nets," 2014, *arXiv:1406.2952*.

[15] D. Lin, X. Shen, C. Lu, and J. Jia, "Deep LAC: Deep localization, alignment and classification for fine-grained recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1666–1674.

[16] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[17] X.-S. Wei, C.-W. Xie, and J. Wu, "Mask-CNN: Localizing parts and selecting descriptors for fine-grained image recognition," 2016, *arXiv:1605.06878*.

[18] S. Huang, Z. Xu, D. Tao, and Y. Zhang, "Part-stacked CNN for fine-grained visual categorization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1173–1182.

[19] T.-Y. Lin, A. RoyChowdhury, and S. Maji, "Bilinear CNN models for fine-grained visual recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1449–1457.

[20] D. Zhang, J. Han, L. Zhao, and D. Meng, "Leveraging prior-knowledge for weakly supervised object detection under a collaborative self-paced curriculum learning framework," *Int. J. Comput. Vis.*, vol. 127, no. 4, pp. 363–380, 2018.

[21] D. Zhang, W. Zeng, J. Yao, and J. Han, "Weakly supervised object detection using proposal- and semantic-level relationships," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 6, pp. 3349–3363, Jun. 2022.

[22] G. Cheng *et al.*, "High-quality proposals for weakly supervised object detection," *IEEE Trans. Image Process.*, vol. 29, pp. 5794–5804, 2020.

[23] D. Zhang, J. Han, L. Zhao, and T. Zhao, "From discriminant to complete: Reinforcement searching-agent learning for weakly supervised object detection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 12, pp. 5549–5560, Dec. 2020.

[24] T. Xiao, Y. Xu, K. Yang, J. Zhang, Y. Peng, and Z. Zhang, "The application of two-level attention models in deep convolutional neural network for fine-grained image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 842–850.

[25] 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.

[26] H. Yao, S. Zhang, C. Yan, Y. Zhang, J. Li, and Q. Tian, "AutoBD: Automated bi-level description for scalable fine-grained visual categorization," *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 10–23, Jan. 2018.

[27] J. Han, X. Yao, G. Cheng, X. Feng, and D. Xu, "P-CNN: Part-based convolutional neural networks for fine-grained visual categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 2, pp. 579–590, Feb. 2022.

[28] 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.

[29] 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.

[30] M. Engin, L. Wang, L. Zhou, and X. Liu, "DeepKSPD: Learning kernel-matrix-based SPD representation for fine-grained image recognition," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 612–627.

[31] Y. Gao, O. Beijbom, N. Zhang, and T. Darrell, "Compact bilinear pooling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 317–326.

[32] S. Kong and C. Fowlkes, "Low-rank bilinear pooling for fine-grained classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 365–374.

[33] X. Wei, Y. Zhang, Y. Gong, J. Zhang, and N. Zheng, "Grassmann pooling as compact homogeneous bilinear pooling for fine-grained visual classification," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 355–370.

[34] C. Yu, X. Zhao, Q. Zheng, P. Zhang, and X. You, "Hierarchical bilinear pooling for fine-grained visual recognition," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 574–589.

[35] 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.

[36] W. Ge, X. Lin, and Y. Yu, "Weakly supervised complementary parts models for fine-grained image classification from the bottom up," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3034–3043.

[37] H. Zheng, J. Fu, Z.-J. Zha, J. Luo, and T. Mei, "Learning rich part hierarchies with progressive attention networks for fine-grained image recognition," *IEEE Trans. Image Process.*, vol. 29, pp. 476–488, 2019.

[38] M. Jaderberg *et al.*, "Spatial transformer networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2017–2025.

[39] X.-S. Wei, P. Wang, L. Liu, C. Shen, and J. Wu, "Piecewise classifier mappings: Learning fine-grained learners for novel categories with few examples," *IEEE Trans. Image Process.*, vol. 28, no. 12, pp. 6116–6125, Dec. 2019.

[40] X.-S. Wei, J.-H. Luo, J. Wu, and Z.-H. Zhou, "Selective convolutional descriptor aggregation for fine-grained image retrieval," *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2868–2881, Jun. 2017.

[41] Z. Gao, Y. Wu, X. Zhang, J. Dai, Y. Jia, and M. Harandi, "Revisiting bilinear pooling: A coding perspective," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 4, 2020, pp. 3954–3961.

[42] X. Zhang, H. Xiong, W. Zhou, and Q. Tian, "Fused one-vs-all features with semantic alignments for fine-grained visual categorization," *IEEE Trans. Image Process.*, vol. 25, no. 2, pp. 878–892, Feb. 2016.

[43] 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.

[44] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "SphereFace: Deep hypersphere embedding for face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 212–220.

[45] W. Liu, Y. Wen, Z. Yu, and M. Yang, "Large-margin softmax loss for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, vol. 2, no. 3, 2016, p. 7.

[46] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive angular margin loss for deep face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4690–4699.

[47] K. Song, J. Han, G. Cheng, J. Lu, and F. Nie, "Adaptive neighborhood metric learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 9, pp. 4591–4604, Sep. 2022.

[48] G. Cheng, C. Yang, X. Yao, L. Guo, and J. Han, "When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative CNNs," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 5, pp. 2811–2821, May 2018.

[49] Y. Zhong, W. Deng, J. Hu, D. Zhao, X. Li, and D. Wen, "SFace: Sigmoid-constrained hypersphere loss for robust face recognition," *IEEE Trans. Image Process.*, vol. 30, pp. 2587–2598, 2021.

[50] X. Zhang, F. X. Yu, S. Kumar, and S.-F. Chang, "Learning spread-out local feature descriptors," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4595–4603.

[51] Y. Tian, X. Yu, B. Fan, F. Wu, H. Heijnen, and V. Balntas, "SOSNet: Second order similarity regularization for local descriptor learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11016–11025.

[52] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 818–833.

[53] 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.

[54] 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.

[55] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, *The CALTECH-UCSD birds-200–2011 Dataset*. Pasadena, CA, USA: California Institute of Technology, 2011.

[56] 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.

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

[58] 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.

[59] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[60] G. Wang, L. Lin, R. Chen, G. Wang, and J. Zhang, "Joint learning of neural transfer and architecture adaptation for image recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Apr. 19, 2021, doi: 10.1109/TNNLS.2021.3070605.

[61] S. R. Dubey, S. Chakraborty, S. K. Roy, S. Mukherjee, S. K. Singh, and B. B. Chaudhuri, "DiffGrad: An optimization method for convolutional neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 11, pp. 4500–4511, Nov. 2020.

[62] R. Collobert, K. Kavukcuoglu, and C. Farabet, "Torch7: A MATLAB-like environment for machine learning," in *Proc. BigLearn, NIPS Workshop*, 2011, pp. 1–6.

[63] Z. Miao, X. Zhao, J. Wang, Y. Li, and H. Li, "Complemental attention multi-feature fusion network for fine-grained classification," *IEEE Signal Process. Lett.*, vol. 28, pp. 1983–1987, 2021.

[64] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

[65] X. Liu, T. Xia, J. Wang, Y. Yang, F. Zhou, and Y. Lin, "Fully convolutional attention networks for fine-grained recognition," 2016, *arXiv:1603.06765*.

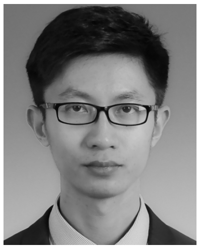
[66] H. Zheng, J. Fu, T. Mei, and J. Luo, "Learning multi-attention convolutional neural network for fine-grained image recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5209–5217.

[67] 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.*, Sep. 2018, pp. 805–821.

[68] D. Chang *et al.*, "The devil is in the channels: Mutual-channel loss for fine-grained image classification," *IEEE Trans. Image Process.*, vol. 29, pp. 4683–4695, 2020.

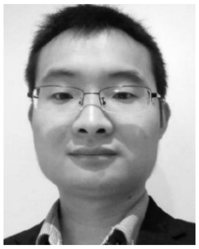
[69] S. Min, H. Yao, H. Xie, Z.-J. Zha, and Y. Zhang, "Multi-objective matrix normalization for fine-grained visual recognition," *IEEE Trans. Image Process.*, vol. 29, pp. 4996–5009, 2020.

[70] Y. Zhang *et al.*, "MSEC: Multi-scale erasure and confusion for fine-grained image classification," *Neurocomputing*, vol. 449, pp. 1–14, Aug. 2021.



Shuo Ye is currently pursuing the Ph.D. degree with the School of Electronic Information and Communications, Huazhong University of Sciences and Technology, Wuhan, China.

His current research interests include computer vision and signal processing, such as fine-grained visual categorization and automatic speech recognition.



Qinmu Peng received the Ph.D. degree from the Department of Computer Science, Hong Kong Baptist University, Hong Kong, in 2015.

He is currently an Assistant Professor with the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan, China. His current research interests include medical image processing, pattern recognition, machine learning, and computer vision.

Wenju Sun is currently pursuing the bachelor's degree with the School of Electronic and Communications, Huazhong University of Sciences and Technology, Wuhan, China.

His current research interests include computer vision and machine learning.

Jiamiao Xu received the B.S. and M.S. degrees from the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan, China, in 2016 and 2019, respectively.

He is currently a Researcher with the Department of Deep Learning, Deeproute Company, Ltd., Shenzhen, China.

Yu Wang received the B.S. degree from the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan, China, in 2019, where he is currently pursuing the M.S. degree.

His current research interests include machine learning and computer vision.

Xinge You (Senior Member, IEEE) received the B.S. and M.S. degrees in mathematics from Hubei University, Wuhan, China, in 1990 and 2000, respectively, and the Ph.D. degree from the Department of Computer Science, Hong Kong Baptist University, Hong Kong, in 2004.

He is currently a Professor with the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan. His current research interests include image processing, wavelet analysis and its applications, pattern recognition, machine learning, and computer vision.

Yiu-Ming Cheung (Fellow, IEEE) received the Ph.D. degree from the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong, in 2000.

He is currently a full Professor with the Department of Computer Science, Hong Kong Baptist University, Hong Kong, SAR, China. His research interests include machine learning, computer vision, pattern recognition, data mining, multiobjective optimization, and information hiding.

Dr. Cheung is a fellow of the Institution of Engineering and Technology (IET) and British Computer Society (BCS). He serves as an Associate Editor for the IEEE TRANSACTIONS ON CYBERNETICS, the IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTATIONAL INTELLIGENCE, the IEEE TRANSACTIONS ON COGNITIVE AND DEVELOPMENTAL SYSTEMS, the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, *Pattern Recognition*, and *Neurocomputing*, to name a few. For details, please refer to: http://www.comp.hkbu.edu.hk/ymc.