Fine Grained Image Retrieval via Piecewise Cross Entropy loss

Xianxian Zeng, Yun Zhang, Xiaodong Wang, Kairui Chen, Dong Li, Weijun Yang

PII: S0262-8856(19)30145-3

DOI: https://doi.org/10.1016/j.imavis.2019.10.006

Reference: IMAVIS 3820

To appear in: *Image and Vision Computing*

Received date: 14 October 2019

Accepted date: 20 October 2019

Please cite this article as: X. Zeng, Y. Zhang, X. Wang, et al., Fine Grained Image Retrieval via Piecewise Cross Entropy loss, *Image and Vision Computing*(2019), https://doi.org/10.1016/j.imavis.2019.10.006

Fine Grained Image Retrieval via Piecewise Cross Entropy loss

Xianxian Zeng[a], Yun Zhang[a,*], Xiaodong Wang[a], Kairui Chen[b], Dong Li[a], Weijun Yang[c]

[a]*Automation, Guangdong University of Technology, Guangzhou, 510006, China*
[b]*School of Mechanical and Electrical Engineering, Guangzhou University, Guangzhou 510006, China.*
[c]*Department of Electromechanical Engineering, Guangzhou City Polytechnic, Guangzhou 510405, China.*

Abstract

Fine-Grained Image Retrieval is an important problem in computer vision. It is more challenging than the task of content-based image retrieval because it has small diversity within the different classes but large diversity in the same class. Recently, the cross entropy loss can be utilized to make Convolutional Neural Network (CNN) generate distinguish feature for Fine-Grained Image Retrieval, and it can obtain further improvement with some extra operations, such as Normalize-Scale layer. In this paper, we propose a variant of the cross entropy loss, named Piecewise Cross Entropy loss function, for enhancing model generalization and promoting the retrieval performance. Besides, the Piecewise Cross Entropy loss is easy to implement. We evaluate the performance of the proposed scheme on two standard fine-grained retrieval benchmarks, and obtain significant improvements over the state-of-the-art, with 11.8% and 3.3% over the previous work on CARS196 and CUB-200-2011, respectively.

Keywords: Fine-grained image retrieval, CNN, Piecewise cross entropy loss
2010 MSC: 00-01, 99-00

*Corresponding author
URL: yun@gdut.edu.cn (Yun Zhang)

1. Introduction

Image Retrieval (IR)[1, 2, 3, 4] is a popular problem in computer vision, and it need to retrieve images that contain object instances of the same variety. Furthermore, Fine-Grained Image Retrieval (FGIR) is to search image through subordinate in the same visual category, like cars [5], birds [6] and products [7]. FGIR is a challenging task because the classes are similar to each other but the images of intra-class can have large difference like pose, illumination and the view point, and it attracted increasing research focus. To solve this problem, a discriminative feature is demanded to distinguish the subtle differences among fine-grained categories. A recent trend is to adopt convolutional neural network (CNN) with metric learning to extract the discriminative and generative features, which aim to distinguish high-dimensional features within/outside fine-grained categories.

However, using metric learning method, like pairwise loss and triplet loss, to train a CNN for FGIR is usually low accuracy and slow training, due to the losses are local structure and the losses with mean square error (MSE) are proved to get stuck in local optimum [8]. To this end, Centralized Ranking Loss (CRL) [9] and Decorrelated Global Centralized Ranking Loss (DGCRL) [10] are proposed, respectively. CRL is a global structure loss that uses a centralized anchor to replace the anchor of the triplet loss. And DGCRL use a fully connect layer to replace the centralized anchor, and then use cross entropy loss to train the CNN.

In this paper, we propose the Piecewise Cross Entropy loss to enhance model generalization. To further improve the performances in FGIR, we use the proposed loss to replace the cross entropy loss in DGCRL [10] and then propose the Decorrelated Global Piecewise Centralized Ranking Loss. In our experiments, we find that the proposed Piecewise Cross Entropy loss not only can enhance model generalization and the performance in FGIR, but also in FGVC. Furthermore, our methods outperform the state-of-the-arts in FGIR, and achieve 86.7 Recall@1 and 70.1 Recall@1 on CARS196 [5] and CUB-200-2011 [6] for

Resnet50.

2. Related works

Fine-Grained Image Retrieval (FGIR): Increasing research focus [11, 12, 13, 9, 10] has been attracted to FGIR in recently years. And there are two challenging problems in FGIR: 1) small inter-class variance; 2) large intra-class variance. Existing works in FGIR can be easily categorized into two main methods. The first one is based on classical hand-crafted features [11] and the second one utilizes deep metric learning [12, 7, 14] or attention module [12, 9] to make CNN extract discriminative features. Recently, the state-of-the-art in FGIR is DGCRL [10], which uses Normalize-Scale Layer to project the features on the hypersphere and then eliminates the gap between Euclidean distance and inner-product. And the details of FGIR will be introduced in Section 3.1.

Fine-Grained Visual Classification (FGVC): FGVC has been an active area of interest in computer vision, and it is another challenging problem that is attracted research focus. Numerous researches are proposed to improve the test performance of FGVC, and they can be categorized into three groups. The first one [15, 16, 17] is constructing a more complicated CNN structure to obtain better classification performance, and the second one [18, 19, 20, 21] utilizes attention module to locate the distinguishing part, which is helpful for improving classification performance. And the third one [22, 23, 24, 25] is adding some noises to alleviate the overfitting and then to improve the classification performance. In summary, the first two methods need extra test time to improve the classification performance, but the last one does not.

3. Method

As shown in Fig.1, the proposed method contains training stage and testing stage. In the training stage, the model can be considered as a general classification model with Normalize-Scale Layer and is trained with the Decorrelated

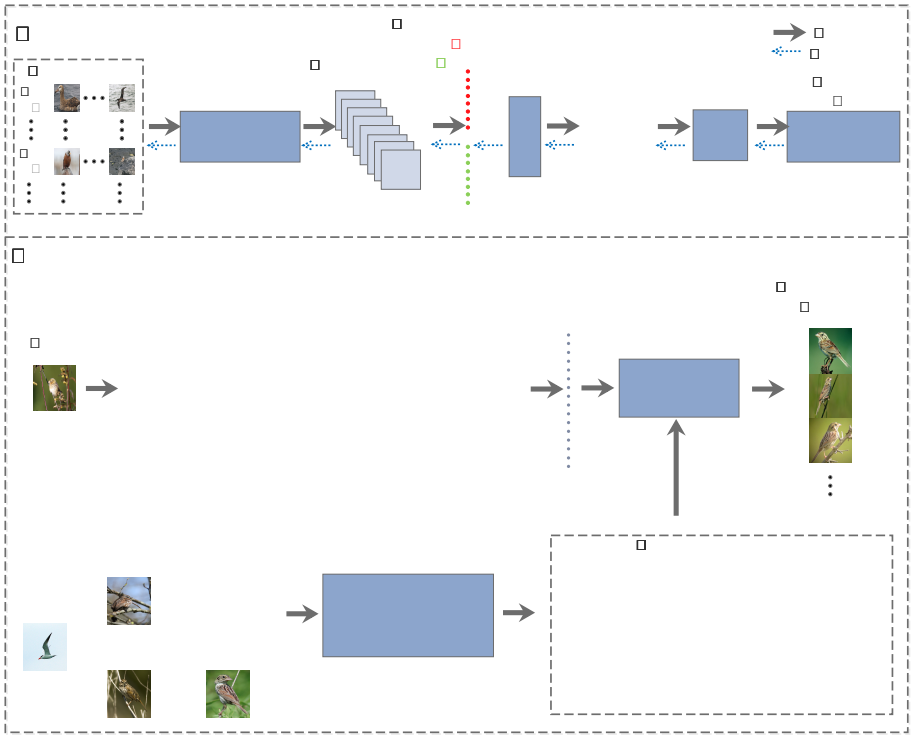


Figure 1: An illustration of the proposed framework. In training time, the framework is similar as classification, trained with the proposed piecewise cross entropy loss. And in testing time, a L2Norm layer is used to extract the norm discriminative features.

Global Piecewise Centralized Ranking Loss (DGPCRL). In testing stage, the model is a feature extractor and extracts image features for FGIR.

3.1. Problem definition and overview

Let $\mathcal{I} = \{I_1, I_2, ..., I_N\}$ be a labeled collection of fine-grained images and each images in training set has its corresponding label $y_i \in \{1, ..., K\}$ of K clusters/classes. In the task of image retrieval, we need a feature extractor $\mathcal{F}_R(\cdot;\theta)$ that can transform an image I_i to a vector $f_i = \mathcal{F}_R(I_i, \theta) \in \mathcal{R}^d$ in a d-dimensional embedding space, and θ is the CNN parameters of the feature extractor. Similarly, in the task of visual classification, we need a combination of a feature extractor and a classifier $F_C\big(F_R(\cdot;\theta);\omega\big)$ that can transform an

image I_i to a probability vector $p_i = Softmax\big(\mathcal{F}_C\big(\mathcal{F}_R(I_i;\theta);\omega\big)\big) \in \mathcal{R}^K$, and ω is the weight of Fully Connected (FC) layer.

To facilitate the understanding, we should summarize the general learning-based pipeline for fine-grained image retrieval. Triplet loss function [26, 27, 28] is a popular loss function for image retrieval, and it is shown as Eq.1. It usually uses the siamese network to generate the feature $\{f_q, f_p, f_n\}$ of the input triplet data $\{I_q, I_p, I_n\}$, where I_q, I_p are different images of the same class, and I_q, I_n are different classes images. And then it computes the Euclidean distances between positive pair and negative pair as $D_{q,p}$ and $D_{q,n}$, respectively. And m is the margin value.

$$L_{trip} = \frac{1}{2}\max(0, m + D_{q,p} - D_{q,n}). \tag{1}$$

The triplet loss function is designed to lessen the distance between positive pair and enlarge the distance between negative pair. However, it is slow training due to the MSE [8]. Besides, Schroff et al. [28] find that the performance of triplet loss depends on the sampling strategy and need very large minibatches to generate discriminative features. To overcome the weakness of the triplet loss, Zheng et al. [10] propose Normalize-Scale Layer as Eq.2 to project the features on a hypersphere.

$$\hat{f}_i = \frac{f_i}{\|f_i\|_2} \cdot \alpha. \tag{2}$$

And then they propose Global Centralized Ranking Loss (GCRL) [9, 10] to improve the performance of FGIR. The loss function is designed to lessen the distance between the feature $\hat{f}_i$ and its corresponding center v_{y_i} and enlarge the distances between $\hat{f}_i$ and negative centers.

$$\begin{aligned} L_{GCRL} = &\max\big(\sum_{i=1}^{N}(m + \|\hat{f}_i - v_{y_i}\| - \frac{1}{K-1}\sum_{j\neq y_i}^{K}\|\hat{f}_i - v_j\|)\big) \\ s.t. \quad &\|\hat{f}_i\| = \alpha. \end{aligned} \tag{3}$$

To accelerate GCRL, Zheng et al. [10] define center v_j as the jth vector weight ω_j of the Fully Connected (FC) layer and propose the Decorrelated Global

Centralized Ranking Loss (DGCRL) as follows:

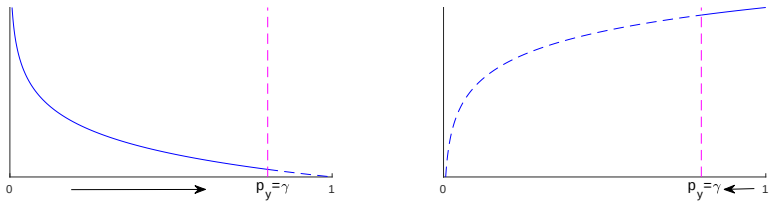
$$L_{DGCRL} = -\log \frac{\exp(\omega_{y_i}^T \hat{f}_i)}{\exp(\omega_{y_i}^T \hat{f}_i) + \sum_{j \neq y_i} \exp(\omega_j^T \hat{f}_i)} + \frac{\lambda}{|\Omega|} \sum_{k \neq j} |\omega_k^T \omega_j| \quad (4)$$
$$s.t. \quad \|\hat{f}_i\| = \alpha,$$

where ω_{y_i} is the y_ith vector weight of the FC layer and Ω denotes different pairwise sets of centers. The former part of DGCRL, which obtains softmax and cross entropy loss, aims at optimize the intra-class compactness and the latter part requires the centers to be perpendicular, which promotes the feature discriminability. Zheng et al. [10] find that the normalized value of center ω is not important, and use Gram-Schmidt Optimization [29] to further improve the FGIR performances.

3.2. Piecewise Cross Entropy Loss and DGPCRL

From another perspective, using DGCRL, a softmax loss with extra restraints, to train the network for FGIR is similar to the training stage in FGVC. Therefore, using some FGVC techniques maybe bring the improvement for feature extraction. In the task of FGVC, one of the most important problem is that the CNN is overfitting, and then can not achieve good performance. To solve this problem, numerous researches are proposed to enhance model generalization: Krause et al. [23] add some data noise from Web data to train the network to obtain better performance; Dubey et al. propose the Pairwise Confusion loss [24] and Maximum Entropy Loss [25] to alleviate the CNN overfitting phenomenon and enhance model generalization; Zheng et al. [21] and Hu et al. [30] add weakly supervised data augmentation to make CNN obtain better generalization. In summarize, we can get better CNN to solve the task of FGVC if we add some noises from data or from the loss function. Inspired by [24, 25], we propose the Piecewise Cross Entropy Loss, which is a variant of cross entropy loss, for training CNN.

$$p_{y_i} = \frac{\exp(\omega_{y_i}^T \hat{f}_i)}{\exp(\omega_{y_i}^T \hat{f}_i) + \sum_{j \neq y_i} \exp(\omega_j^T \hat{f}_i)} \quad (5)$$



(a) $loss = -log(p_y)$ (b) $loss = log(p_y)$

Figure 2: The curves of the proposed loss function.

$$loss_{CE} = -\frac{1}{N}\sum_{i=1}^{N}\log p_{y_i}, \tag{6}$$

The cross entropy loss, as shown in Eq.6, is designed to make $p_{y_i} \to 1$, so the network can overconfidently classify the data from the training set. However, in the task of FGVC, it easily becomes overfitting. Therefore, we propose the Piecewise Cross Entropy loss function to alleviate overconfidence, and it is shown as follows:

$$\begin{aligned} loss_{PCE} &= \frac{1}{N}\sum_{i=1}^{N}\eta\log p_{y_i}, \\ s.t. \quad \eta &= \begin{cases} 1, & p_{y_i} >= \gamma, \\ -1, & p_{y_i} < \gamma. \end{cases} \end{aligned} \tag{7}$$

where $\gamma \in (0, 1]$. If $\gamma = 1$, the Piecewise Cross Entropy loss is the same as the cross entropy loss. To facilitate the understanding, Fig.2 shows the two curves of the different status of the proposed loss. We can find that if $p_y < \gamma$, $loss_{PCE}$ will make $p_y \to 1$, while make $p_y \to 0$ if $p_y >= \gamma$. It means that when $p_y >= \gamma$, the proposed loss is a noise to reduce the confidence of the model. And p_{y_i} will be shaking around γ. In back propagation, the gradient of the proposed loss function can be easily obtained as:

$$\frac{\partial Loss_{PCE}}{\partial p_{y_i}} = \eta \cdot \frac{1}{p_{y_i}} = \begin{cases} \frac{1}{p_{y_i}}, & p_{y_i} >= \gamma, \\ -\frac{1}{p_{y_i}}, & p_{y_i} < \gamma. \end{cases} \tag{8}$$

In our experiments, we find that when the model is shaking, the model will find other stable states, which can enhance model generalization and achieve better performance in the task on FGIR and FGVC.

In here, we use the Piecewise Cross Entropy loss to substitute for the cross entropy loss in DGCRL to obtain better feature extractor in FGIR. Therefore, the DGPCRL for FGIR can be obtained as:

$$\begin{aligned} loss &= \frac{1}{N}\sum_{i=1}^{N}\eta \log p_{y_i} + \frac{\lambda}{\|\Omega\|}\sum_{k\neq j}\|\omega_k\omega_j^T\|, \\ s.t. \quad \eta &= \begin{cases} 1, & p_{y_i} >= \gamma, \\ -1, & p_{y_i} < \gamma. \end{cases} \end{aligned} \tag{9}$$

4. Experiments

4.1. Datasets and Evaluation Protocols

We evaluate the retrieval performances on two widely-used benchmarks: CARS196 [5] and CUB-200-2011 [6]. CARS196[1] contains 196 car classes with $16,185$ images and CUB-200-2011[2] contains 200 bird classes with $11,788$ images. Following the previous works [31, 14, 7, 9, 10], we employ the first 98 classes and first 100 classes for training in CARS196 and CUB-200-2011, respectively. And then we use the ramaining 96 classes and 100 classes for testing.

As same as the previous works, we evaluate the retrieval by the standard Recall@**K**. Strictly following the setting in [7], Recall@**K** is the average recall scores over all query images in the test set. For each query, the top K similar images are returned. The recall score will be 1 if there are at least one positive image in the top K returned images, and 0 otherwise.

4.2. Implementation Details

We apply the widely-used Resnet50 and Resnet101 [32] in our experiment, and the networks are the pretrained models on ImageNet ILSVRC-2012 [33]. We

[1] `https://ai.stanford.edu/~jkrause/cars/car_dataset.html`

[2] `http://www.vision.caltech.edu/visipedia/CUB-200-2011.html`

Table 1: Recall@**K** on CARS196 and CUB-200-2011 with baseline methods. The best results are in **bold**.

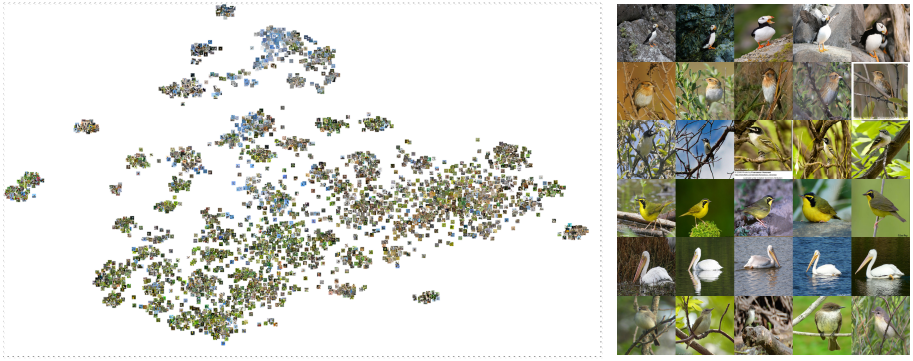
Method	CARS196						CUB-200-2011					
K=	1	2	4	8	16	32	1	2	4	8	16	32
Contrastive	21.7	32.3	46.1	58.9	72.2	83.4	26.4	37.7	49.8	62.3	76.4	85.3
Triplet	39.1	50.4	63.3	74.5	84.1	89.8	36.1	48.6	59.3	70.0	80.2	88.4
LiftedStruct	49.0	60.3	72.1	81.5	89.2	92.8	47.2	58.9	70.2	80.2	89.3	93.2
Facility Location	58.1	70.6	80.3	87.8	-	-	48.2	61.4	71.8	81.9	-	-
N-pairs	53.9	66.76	77.75	86.35	-	-	45.37	58.41	69.51	79.49	-	-
Binomial Deviance	-	-	-	-	-	-	52.8	64.4	74.7	83.9	90.4	94.3
Histogram Loss	-	-	-	-	-	-	50.3	61.9	72.6	82.4	88.8	93.7
PDDM+Quadruplet	57.4	68.6	80.1	89.4	92.3	94.9	58.3	69.2	79.0	88.4	93.1	95.7
SCDA	58.5	69.8	79.1	86.2	91.8	95.9	62.2	74.2	83.2	90.1	94.3	97.3
CRL-WSL	63.9	73.7	82.1	89.2	93.7	96.8	65.9	76.5	85.3	90.3	94.4	97.0
DGCRL	75.9	83.9	89.7	94.0	96.6	98.0	67.9	79.1	86.2	91.8	94.8	97.1
Our method(Resnet50)	**86.7**	**91.7**	**95.2**	**97.0**	**98.3**	**99.1**	**70.1**	**79.8**	**86.9**	**92.0**	**95.0**	**97.3**
Our method(Resnet101)	**87.7**	**92.6**	**95.3**	**97.3**	**98.3**	**99.1**	**71.2**	**80.2**	**87.3**	**92.4**	**95.3**	**97.3**

implement our models with the PyTorch framework [34], and all the experiments are based on a RTX-2080Ti GPU. We train our models with SGD optimizer and set 280×280 as the input size, 100 as the scale parameter α, 0.1 as the weight λ, 60 as the mini-batch-size, 0.001 as the initial learning rate, 0.9 as the momentum, 100 as the epochs and 5e-6 as the weight decay. In our experiment, we find that the same hyperparameters also can get good performance in different datasets.

4.3. Baseline and Performance

We compare our method with the following state-of-the-art methods: Contrastive [31], Triplet [12], LiftedStruct [7] and PDDM+Quadruplet [14] are the methods that train the feature extractor by using pairwise loss, triplet loss or quadruplet loss. Facility Location [36] proposes a novel metric learning method for the global structure. N-pairs loss [37] improves the triplet loss by using softmax cross-entropy loss. Binomial Deviance [38] is proposed to estimate the cost between similar examples and Histogram Loss [39] is proposed to make the distributions of the positive and negative pairs less overlapping. SCDA [13] select distinguishing position and generate representative feature without fine-tune. CRL-WSL [9] employs the centralized ranking loss with weakly supervised localization method to train the feature extractor. DGCRL [10] eliminates the gap

(a) CARS196



(b) CUB-200-2011

Figure 3: Barnes-Hut t-SNE visualization [35] of our embedding feature (left) and some results of FGIR (right) on CARS196 and CUB-200-2011.

between inner-product and the Euclidean distance in training stage and testing stage by adding Normalize-Scale layer, and enhance the intra-class separability and inter class compactness by its Decorrelated Global-aware Centralized Ranking Loss.

As shown in Tab.1, our methods outperform the state-of-the-arts in terms of Recall@**K**. Specifically, our schemes (Resnet50/Resnet101) achieve 86.7/87.7 Recall@**1** and 70.1/71.2 Recall@**1** on CARS196 and CUB-200-2011, respectively. Besides, Fig.3 shows the t-SNE [35] and the task of fine-grained retrieval of our Resnet50 scheme. We find that our method can further group the similar objects and split diverse classes.

Table 2: The retrieval performances of different γ. The best results are in **bold**.

Resnet50												
Method	CARS196						CUB-200-2011					
Piecewise Cross Entropy	1	2	4	8	16	32	1	2	4	8	16	32
$\gamma = 1$ (Cross Entropy)	82.2	88.0	92.5	95.3	97.6	98.7	68.2	78.4	85.9	91.2	94.7	96.8
$\gamma = 0.9$	84.9	91.0	94.5	96.3	98.1	99.0	68.9	79.0	86.1	91.4	94.8	97.1
$\gamma = 0.8$	85.2	91.2	95.1	96.9	98.1	**99.2**	69.7	79.2	85.9	91.4	94.9	97.1
$\gamma = 0.7$	**86.7**	**91.7**	**95.2**	**97.0**	**98.3**	99.1	**70.1**	**79.8**	**86.9**	**92.0**	**95.0**	**97.3**
$\gamma = 0.6$	85.3	91.4	95.0	97.0	98.2	**99.2**	69.9	79.4	86.4	91.6	94.8	97.2
Resnet101												
Method	CARS196						CUB-200-2011					
Piecewise Cross Entropy	1	2	4	8	16	32	1	2	4	8	16	32
$\gamma = 1$ (Cross Entropy)	83.9	90.0	93.9	96.4	98.0	98.9	69.3	78.9	86.3	91.8	95.1	97.1
$\gamma = 0.9$	86.4	91.8	94.9	97.2	98.4	99.1	69.8	79.5	87.1	92.0	95.0	97.1
$\gamma = 0.8$	87.2	92.6	**95.6**	97.3	**98.5**	**99.2**	70.9	79.9	87.1	92.0	95.3	97.4
$\gamma = 0.7$	**87.7**	**92.6**	95.3	**97.3**	98.3	99.1	**71.2**	**80.2**	87.3	**92.4**	95.3	**97.3**
$\gamma = 0.6$	86.8	92.0	95.1	97.1	98.3	99.1	71.0	80.2	**87.4**	92.3	**95.4**	97.2

5. Discussion

5.1. Ablation Study: the parameter γ

In our scheme, there are three important hyperparameters: the scale α, the weight λ and the threshold γ. The previous work [10] find that the performance of FGIR is stable when the hyperparameter α is higher than 16 and lesser than 128, so we set $\alpha = 100$. λ is hyperparameter for the vertical constraint, and we set $\lambda = 0.1$, as same as [10]. The threshold γ is the most important hyperparameter of the proposed loss, and the classification output p_{y_i} will be shaking around it in training stage. And we use different CNN structures (Resnet50/Resnet101) to further confirm the effect of the proposed loss. We investigate how γ affects the FGIR performance by setting different value. Table 2 shows the retrieval performances with different value of γ on CARS196 and CUB-200-2011.

We can simply consider the performances of cross entropy loss is a re-implement of the previous work [10], and it achieves better performance due to the superior hyperparameters. From Table 2, we can find that the Piecewise Cross Entropy loss is helpfully to obtain a better feature extractors in FGIR, whether the value of γ and the CNN structure is. In summary, we find that the models obtain the best retrieval performances if $\gamma = 0.7$.

5.2. Fine-grained recognition of the proposed loss function

As shown in Eq.7, the proposed PCE loss is a variant of Softmax loss. Thus, we aim to confirm the influence of PCE loss in fine-grained object recognition. We employ PCE loss with $\eta = 0.8$ to train three different CNN models (Resnet50[32], InceptionV3[40] and Densenet161[41])[3] on three fine-grained datasets (CARS196[5], CUB-200-2011[6] and Dog120[42]). We train these models with SGD optimizer, 32 as the mini-batch-size, 0.01 as the initial learning rate and is divided by 2 in every 50 epochs, 0.9 as the momentum, 600 as the epochs and 5e-4 as the weight decay. As shown in Table 3, we find that the results are unstable: sometime the PCE loss is helpful to improve the performance of fine-grained object recognition but sometime it is worse than the performance of cross entropy loss.

Table 3: The recognition performances of different loss function.

Model	Loss	CARS196	CUB-200-2011	Dog120
Resnet50	cross entropy	90.7%	82.2%	82.2%
	PCE($\eta = 0.8$)	**91.8%**	**83.1%**	82.2%
InceptionV3	cross entropy	90.5%	82.0%	85.0%
	PCE($\eta = 0.8$)	**91.2%**	**82.5%**	**85.4%**
Densenet161	cross entropy	91.7%	**83.9%**	**82.8%**
	PCE($\eta = 0.8$)	**92.7%**	83.8%	82.4%

6. Conclusion

In this paper, we propose a variant of cross entropy loss, named Piecewise Cross Entropy loss, for enhancing model generalization. The Piecewise Cross Entropy loss can not only improve the performances in FGIR and FGVC without extra computation in testing stage, but also simply implement. Comparing to

[3]The results are worse than the best performances of these the CNN models, due to the hyperparameters may be different from their previous works.

the previous works in FGIR, we obtain the best performance on CARS196 and CUB-200-2011.

Acknowledgments

This work was supported by National Natural Science Foundation of China: 61503084, U1501251, Natural Science Foundation of Guangdong Province, China: 2016A030310348 and the Science and Technology Program of Guangzhou, China: 201804010098.

References

[1] Y. Gong, L. Wang, R. Guo, S. Lazebnik, Multi-scale orderless pooling of deep convolutional activation features, in: European conference on computer vision, Springer, 2014, pp. 392–407.

[2] G. Tolias, R. Sicre, H. Jégou, Particular object retrieval with integral max-pooling of cnn activations, arXiv preprint arXiv:1511.05879.

[3] H. Noh, A. Araujo, J. Sim, T. Weyand, B. Han, Large-scale image retrieval with attentive deep local features, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 3456–3465.

[4] N. Garcia, G. Vogiatzis, Learning non-metric visual similarity for image retrieval, Image and Vision Computing 82 (2019) 18–25.

[5] J. Krause, M. Stark, J. Deng, L. Fei-Fei, 3d object representations for fine-grained categorization, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2013, pp. 554–561.

[6] C. Wah, S. Branson, P. Welinder, P. Perona, S. Belongie, The caltech-ucsd birds-200-2011 dataset.

[7] H. Oh Song, Y. Xiang, S. Jegelka, S. Savarese, Deep metric learning via lifted structured feature embedding, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4004–4012.

[8] P. Golik, P. Doetsch, H. Ney, Cross-entropy vs. squared error training: a theoretical and experimental comparison., in: Interspeech, Vol. 13, 2013, pp. 1756–1760.

[9] X. Zheng, R. Ji, X. Sun, Y. Wu, F. Huang, Y. Yang, Centralized ranking loss with weakly supervised localization for fine-grained object retrieval., in: IJCAI, 2018, pp. 1226–1233.

[10] X. Zheng, R. Ji, X. Sun, B. Zhang, Y. Wu, F. Huang, Towards optimal fine grained retrieval via decorrelated centralized loss with normalize-scale layer.

[11] L. Xie, J. Wang, B. Zhang, Q. Tian, Fine-grained image search, IEEE Transactions on Multimedia 17 (5) (2015) 636–647.

[12] X. Zhang, F. Zhou, Y. Lin, S. Zhang, Embedding label structures for fine-grained feature representation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1114–1123.

[13] X.-S. Wei, J.-H. Luo, J. Wu, Z.-H. Zhou, Selective convolutional descriptor aggregation for fine-grained image retrieval, IEEE Transactions on Image Processing 26 (6) (2017) 2868–2881.

[14] C. Huang, C. C. Loy, X. Tang, Local similarity-aware deep feature embedding, in: Advances in neural information processing systems, 2016, pp. 1262–1270.

[15] T.-Y. Lin, A. RoyChowdhury, S. Maji, Bilinear cnn models for fine-grained visual recognition, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 1449–1457.

[16] Y. Wang, V. I. Morariu, L. S. Davis, Learning a discriminative filter bank within a cnn for fine-grained recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4148–4157.

[17] P. Li, J. Xie, Q. Wang, Z. Gao, Towards faster training of global covariance pooling networks by iterative matrix square root normalization, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 947–955.

[18] M. Jaderberg, K. Simonyan, A. Zisserman, et al., Spatial transformer networks, in: Advances in neural information processing systems, 2015, pp. 2017–2025.

[19] J. Fu, H. Zheng, T. Mei, Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4438–4446.

[20] H. Zheng, J. Fu, T. Mei, J. Luo, Learning multi-attention convolutional neural network for fine-grained image recognition, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 5209–5217.

[21] H. Zheng, J. Fu, Z.-J. Zha, J. Luo, Looking for the devil in the details: Learning trilinear attention sampling network for fine-grained image recognition, arXiv preprint arXiv:1903.06150.

[22] S. Reed, H. Lee, D. Anguelov, C. Szegedy, D. Erhan, A. Rabinovich, Training deep neural networks on noisy labels with bootstrapping, arXiv preprint arXiv:1412.6596.

[23] J. Krause, B. Sapp, A. Howard, H. Zhou, A. Toshev, T. Duerig, J. Philbin, L. Fei-Fei, The unreasonable effectiveness of noisy data for fine-grained recognition, in: European Conference on Computer Vision, Springer, 2016, pp. 301–320.

[24] A. Dubey, O. Gupta, P. Guo, R. Raskar, R. Farrell, N. Naik, Pairwise confusion for fine-grained visual classification, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 70–86.

[25] A. Dubey, O. Gupta, R. Raskar, N. Naik, Maximum-entropy fine grained classification, in: Advances in Neural Information Processing Systems, 2018, pp. 637–647.

[26] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, Y. Wu, Learning fine-grained image similarity with deep ranking, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1386–1393.

[27] K. Q. Weinberger, J. Blitzer, L. K. Saul, Distance metric learning for large margin nearest neighbor classification, in: Advances in neural information processing systems, 2006, pp. 1473–1480.

[28] F. Schroff, D. Kalenichenko, J. Philbin, Facenet: A unified embedding for face recognition and clustering, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 815–823.

[29] E. Schmidt, Über die auflösung linearer gleichungen mit unendlich vielen unbekannten, Rendiconti del Circolo Matematico di Palermo (1884-1940) 25 (1) (1908) 53–77.

[30] T. Hu, H. Qi, See better before looking closer: Weakly supervised data augmentation network for fine-grained visual classification, arXiv preprint arXiv:1901.09891.

[31] S. Bell, K. Bala, Learning visual similarity for product design with convolutional neural networks, ACM Transactions on Graphics (TOG) 34 (4) (2015) 98.

[32] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[33] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: 2009 IEEE conference on computer vision and pattern recognition, Ieee, 2009, pp. 248–255.

[34] [link].
URL https://pytorch.org/

[35] L. Van Der Maaten, Accelerating t-sne using tree-based algorithms, The Journal of Machine Learning Research 15 (1) (2014) 3221–3245.

[36] H. Oh Song, S. Jegelka, V. Rathod, K. Murphy, Deep metric learning via facility location, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5382–5390.

[37] K. Sohn, Improved deep metric learning with multi-class n-pair loss objective, in: Advances in Neural Information Processing Systems, 2016, pp. 1857–1865.

[38] D. Yi, Z. Lei, S. Li, Deep metric learning for practical person re-identification (2014), ArXiv e-prints.

[39] E. Ustinova, V. Lempitsky, Learning deep embeddings with histogram loss, in: Advances in Neural Information Processing Systems, 2016, pp. 4170–4178.

[40] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2818–2826.

[41] G. Huang, Z. Liu, L. Van Der Maaten, K. Q. Weinberger, Densely connected convolutional networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4700–4708.

[42] A. Khosla, N. Jayadevaprakash, B. Yao, F.-F. Li, Novel dataset for fine-grained image categorization: Stanford dogs, in: Proc. CVPR Workshop on Fine-Grained Visual Categorization (FGVC), Vol. 2, 2011.

Highlights

- Fine-grained image retrieval is an important problem in computer vision.
- In this paper, PCE loss is proposed for fine-grained image retrieval.
- Due to the proposed loss, our model obtains SOTA performances on two benchmarks.

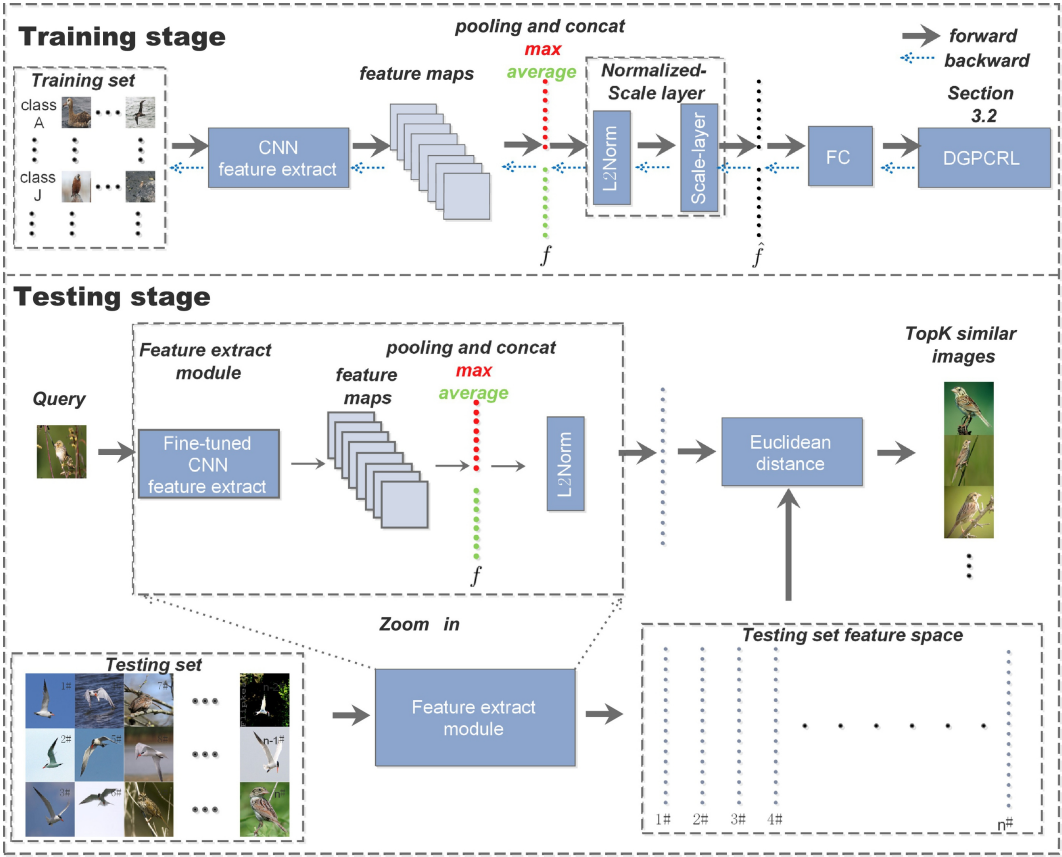
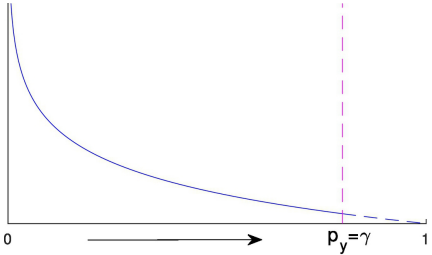
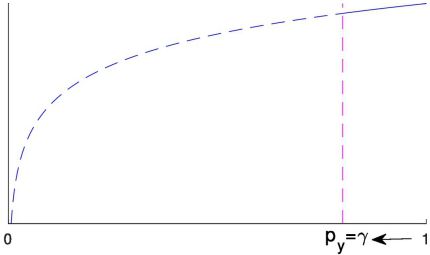

Training stage
pooling and concat
max
average
forward
backward
Training set
class A
class J
feature maps
CNN feature extract
Normalized-Scale layer
L2Norm
Scale-layer
FC
Section 3.2
DGPCRL
f
f̂
Testing stage
Feature extract module
pooling and concat
max
average
feature maps
Query
Fine-tuned CNN feature extract
L2Norm
Euclidean distance
TopK similar images
Zoom in
Testing set
Feature extract module
Testing set feature space
1#
2#
3#
4#
n#

Figure 1

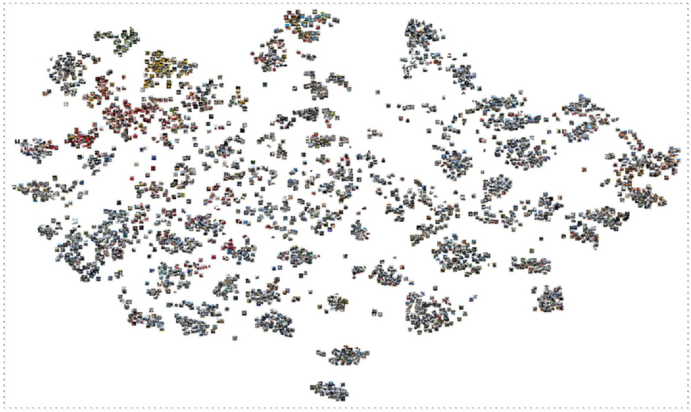


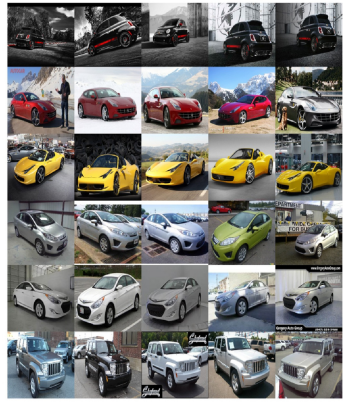
(a) $loss = -log(p_y)$



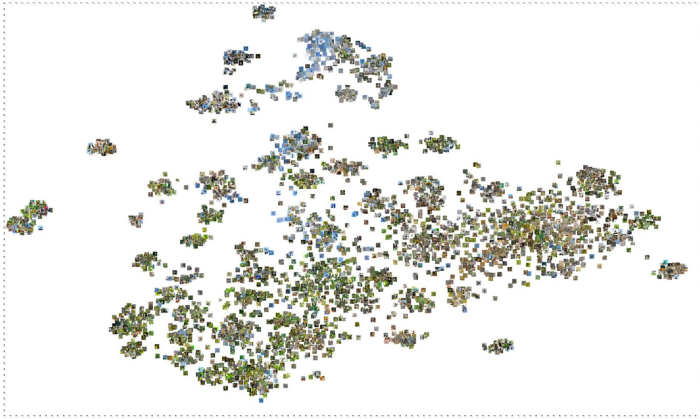
(b) $loss = log(p_y)$

Figure 2





(a) CARS196



(b) CUB-200-2011

Figure 3