Centralized Ranking Loss with Weakly Supervised Localization for Fine-Grained Object Retrieval

Xiawu Zheng[1,2], **Rongrong Ji**[1,2*], **Xiaoshuai Sun**[3], **Yongjian Wu**[4], **Feiyue Huang** [4], **Yanhua Yang** [5]
[1] Fujian Key Laboratory of Sensing and Computing for Smart City, Xiamen University
[2] School of Information Science and Engineering, Xiamen University
[3] Harbin Institute of Technology
[4] Tencent Technology (Shanghai) Co.,Ltd
[5] Xidian University

Abstract

Fine-grained object retrieval has attracted extensive research focus recently. Its state-of-the-art schemes are typically based upon convolutional neural network (CNN) features. Despite the extensive progress, two issues remain open. On one hand, the deep features are coarsely extracted at image level rather than precisely at object level, which are interrupted by background clutters. On the other hand, training CNN features with a standard triplet loss is time consuming and incapable to learn discriminative features. In this paper, we present a novel fine-grained object retrieval scheme that conquers these issues in a unified framework. Firstly, we introduce a novel centralized ranking loss (**CRL**), which achieves a very efficient (1,000 times training speedup comparing to the triplet loss) and discriminative feature learning by a "centralized" global pooling. Secondly, a weakly supervised attractive feature extraction is proposed, which segments object contours with top-down saliency. Consequently, the contours are integrated into the CNN response map to precisely extract features "within" the target object. Interestingly, we have discovered that the combination of CRL and weakly supervised learning can reinforce each other. We evaluate the performance of the proposed scheme on widely-used benchmarks including *CUB200-2011* and *CARS196*. We have reported significant gains over the state-of-the-art schemes, *e.g.*, 5.4% over SCDA [Wei *et al.*, 2017] on *CARS196*, and 3.7% on *CUB200-2011*.

1 Introduction

Given a query image, fine-grained object retrieval (FGOR) aims to retrieve images that contain object instances of the same variety. After firstly proposed in [Xie *et al.*, 2015], FGOR has rapidly become a research hotspot [Wei *et al.*, 2017; Huang *et al.*, 2016; Bell and Bala, 2015; Hyun *et al.*, 2016; Ustinova and Lempitsky, 2016; Wang *et al.*, 2014; Zhang *et al.*, 2016a], which poses various applications ranging from product search, car retrieval, to species identification. In such a setting, object instances are similar to each other and within a general class. Therefore, different instances can only be distinguished by subtle parts, which serves as the key challenge.

Earlier works in fine-grained image retrieval mainly resort to using hand-craft features. For instance, the work in [Xie *et al.*, 2015] adopted Bag-of-Visual-Words descriptor in combination with SVM classifier to identify instances with fine-grained semantic and visual appearance. More recently, deep learning have been applied in FGOR [Huang *et al.*, 2016; Bell and Bala, 2015; Hyun *et al.*, 2016; Wang *et al.*, 2014]. In particular, these methods follow a deep metric learning paradigm, which learns a deep embedding space that pulls similar images to be closer, and vice versa. [Wei *et al.*, 2017] proposed to select features by a coarse saliency map to promote the retrieval performance, which indicates the importance of object localization. However, extracting features from pre-trained CNNs with a coarse saliency map is not always discriminative, which requires precise object localization and contour segmentation.

Beyond FGOR, recent advances in fine-grained image classification also support this argument, which can help to distinguish the subtle differences among specific object components. Nevertheless, most methods in classification are required to provide full supervision (*i.e.*, bounding box [Jonathan *et al.*, 2015] or part annotations [Xie *et al.*, 2013]) to train accurate boundary segmentations. however, such a setting is, impractical for FGOR, which typically searches a large-scale space with a large amount of object categories. Under such a circumstance, it is infeasible to label sufficient boundaries or bounding boxes. Some recent works [He and Peng, 2017; Simon and Rodner, 2015; Xiao *et al.*, 2015; Zhang *et al.*, 2016b] attempted to classify fine-grained images in a weakly supervised condition, *i.e.*, the bounding boxes and part annotations are not needed at training. Instead, all testing examples in classification should be predefined, which are therefore co-segmented or co-localized. In contrast, for FGOR, the testing identities are usually disjoint from the training set, and are unknown for the object localization, making the approaches in [He and Peng, 2017; Simon and Rodner, 2015; Xiao *et al.*, 2015; Zhang *et al.*, 2016b] being impractical for FGOR.

Another key drawbacks lies in the poor training efficiency

*corresponding author

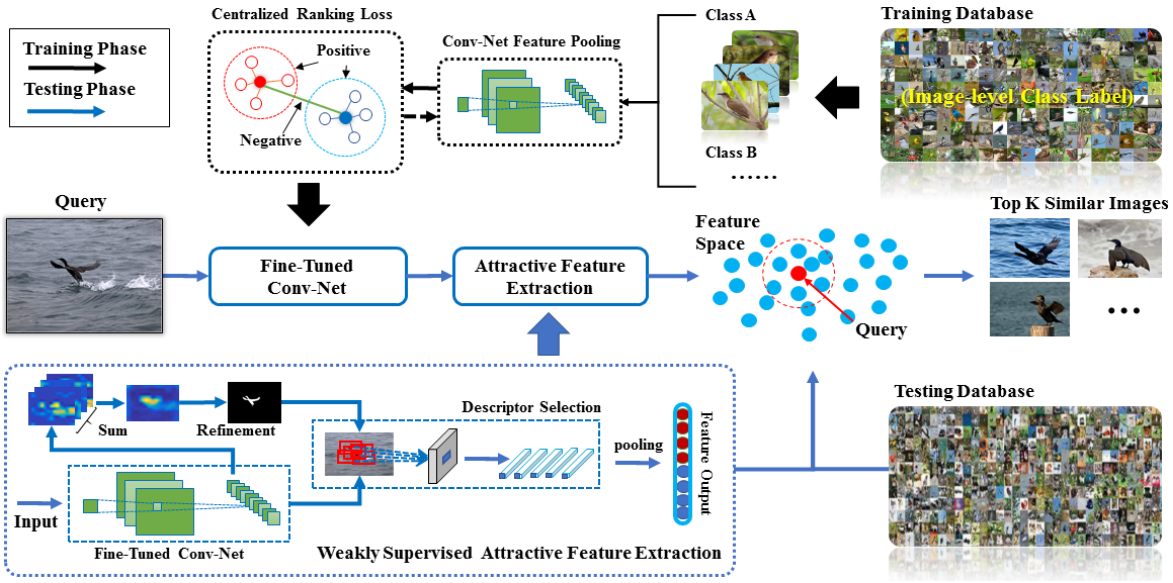

Figure 1: The proposed framework. We train our network by the Centralized Ranking Loss (CRL). In CRL, we compute the loss and gradients based on class-level global max pooling (GMP) and global average pooling (GAP) features. The loss function affects the activation distribution of the feature response map, leading to more accurate saliency maps and promoting more discriminative features. In the testing phase, We extract attractive features for the query image and database instances by a weakly supervised feature extraction method. For each image, our method first localizes important objects via saliency extraction and refinement. After that, attractive object features are selected according to the coverage rate between the feature receptive field and the salient object mask. The final feature is generated by GMP and GAP, based on which we retrieve the top-K related images using L2 distance.

and low feature discriminability in most deep metric learning based FGOR methods [Huang *et al.*, 2016; Bell and Bala, 2015; Hyun *et al.*, 2016; Wang *et al.*, 2014]. First, pairwise/triplet/high-order embedding methods are time consuming, where the complexity can be $\mathcal{O}(N^2)$ or $\mathcal{O}(N^3)$ given N images. Second, most samples used to fine-tune a pre-trained CNN model are easy samples, leading to the overfitting in model convergence, which is incapable of generating discriminative feature. Using hard example mining [Schroff *et al.*, 2015; Sohn, 2016] is an alternative way to train discriminative features. However, the efficiency drawback retains *i.e.*, it is difficult to implement and mine hard examples with a complexity of $\mathcal{O}(N^2)$ or $\mathcal{O}(N^3)$.

In this paper, we present a novel fine-grained object retrieval scheme that conquers the above two drawbacks in a unified framework. The proposed framework, termed **Centralized Ranking Loss** with **Weakly-Supervised Localization**, is illustrated in Fig.1. First, we present a novel weakly-supervised object localization, which provides object boundaries, from which more representative feature can be extracted against background clutters. Second, we introduce a novel centralized ranking loss, which can largely improve the training efficiency and feature discriminability. Moreover, we have found that both components can reinforce each other, since the latter component can essentially affect the discriminative objects/parts response in the feature maps of CNN (as shown in Fig.2). In particular, the contributions of our framework can be itemized as follows:

- We design a new loss function named Centralized Ranking Loss (**CRL**). As illustrated in Fig.1, the loss affects the distribution in the CNN response maps, which generates a more accurate saliency map (Fig.2) and promotes a more discriminative feature. Moreover, it also significantly reduces the search space (from $\mathcal{O}(N^3)$ to $\mathcal{O}(NL^2)$, where L denotes the number of enters, and N denotes the number of images, leading to a training acceleration by 1,000 times in our experiments, as quantitatively shown in Fig.3.
- We propose a novel scheme for weakly-supervised attractive feature extraction. The scheme employs fine-tuned CNNs to obtain a coarse object saliency map, which is subsequently refined by mixture models to generate a precise pixel-wise object mask without using any object bounding box or boundary supervision.

We have conducted experiments on image retrieval and weakly supervised localization on two widely-used fine-grained image retrieval datasets, *CUB-200-2011* [Wah *et al.*, 2011] and *CARS196* [Jonathan *et al.*, 2013]. It is shown that the proposed method significantly outperforms state-of-the-art methods [Wei *et al.*, 2017; Huang *et al.*, 2016]. For instance, 5.4% over SCDA in *CARS196*, 1,000 times faster in training over triplet loss.

2 The Proposed Method

As shown in Fig.1, the proposed method contains both offline and online phases. In offline training, we fine-tune a pre-trained CNN by the proposed centralized ranking loss (Sec.2.1). To that effect, the loss function computes gradients based on class-level global max pooling (GMP)

and global average pooling (GAP) of the raw features. As quantitatively shown in Sec.3.2, the convolutional responses are able to capture discriminative objects/parts than used. In combination with saliency extraction and mixture models, the proposed localization scheme can precisely extract the contour of salient objects. After that, attractive features are extracted according to the coverage rate between the receptive field and the object mask, upon which the final feature is generated by GMP & GAP. In online retrieval, the feature of query is extracted from its attractive region (Sec.2.2), based on which we retrieve the most related images by measuring a simple L2 distance.

2.1 Centralized Ranking Loss

Motivation. Due to the tremendous search space and fully-connected features, previous methods of deep metric learning [Huang *et al.*, 2016; Bell and Bala, 2015; Hyun *et al.*, 2016; Wang *et al.*, 2014; Schroff *et al.*, 2015] are less effective in object localization and feature embedding. The major reason is that, the location information encoded in the convolutional layers fails to be transferred to the fully-connected layers. The proposed Centralized Ranking Loss (CRL) conquers this issue based on two intuitions: (1) For a batch, the feature of the center with same label is representative for the class, which can be regarded as an anchor to replace the traditional triplet loss; (2) Some channels in the feature map with high spatial responses can represent one object/part, the saliency map of which is required to be activated for a class. As a result, it is reasonable to back-propagate through the centre's global max and average pooling, which refines the convolutional map by emphasizing more on the representative dimensions.

In Fig.1, CRL updates parameters by class center, which strengthens the false negative dimensions and depresses the false positive dimensions, as reflected on the saliency map (shown in Fig.2). The gradient directly influences the response distribution of the corresponding feature. The above operations can be regarded as an implementation of a typical task-driven top-down attention, which is able to generate attractive regions.

Let f_i be The feature vector for image I_i. Let $\mathcal{A} = \{a_k\}, k = 1, 2, 3...K$ be The set of center features for K classes and $a_k = \frac{1}{|\mathcal{P}_k|}\sum_{\mathcal{P}_k} f_i$, where $|\mathcal{P}_k|$ denotes the number of samples in $\mathcal{P}_k$. Let $D_{i,j}$ be The distance between two features f_i and f_j.

Triplet Loss function. [Wang *et al.*, 2014] computes the penalty by triplet data $\{I_q, I_p, I_n\}$ where I_q and I_p have the same class labels, and I_q and I_n have different. The triplet loss function is defined as follows:

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

where $D_{q,p}$ and $D_{q,n}$ denote the distance and negative distances respectively. The performance of triplet loss depends highly on the sampling strategy [Schroff *et al.*, 2015].The idea is to construct triplets by associating with each positive pair in the minibatch a "semi-hard" negative example. To generate discriminative features, [Schroff *et al.*, 2015] had to use very large minibatches, making it hard to train on GPU.

Centralized Ranking Loss defines the ranking through the class center, aiming at to minimize the intra-class distance, as well as maximizing the inter-class distance in a very efficient manner. The corresponding loss is defined as:

$$\mathcal{L} = \sum_{a_k \in \mathcal{A}} \sum_{a_l \in \mathcal{A}} \sum_{f_i \in \mathcal{P}_k} max(0, m + ||f_i - a_k||^2 - ||f_i - a_l||^2), \tag{2}$$

where m is a positive scalar that controls the margin. Given a centralized triplet, the sub-gradients are defined as:

$$\frac{\partial \mathcal{L}}{\partial f_i} = \frac{f_i - a_k}{||f_i - a_k||^2} - \frac{f_i - a_l}{||f_i - a_l||^2}. \tag{3}$$

As we can see in Eq.(3), CRL forces the feature f_i to approach the target class center and leave away from centers of other classes. The class mean vectors are computed in each batch. We only update the parameters through the gradient of positive and negative examples, rather than using the class centers. The sub-gradient calculation using Eq.3 is extremely effective, which will be quantitatively shown in Fig.3. Moreover, Eq.3 can promote the precision of saliency detection, as well as the discrimination of feature representation. Please refer to the evidences quantitatively shown in Fig.2 and Tab.5, respectively.

As mentioned before, the resulting feature is composed by GMP and GAP. When a dimension is representative for a class, GMP will enhance the corresponding positions, while GAP will enhance its corresponding channels in the convolutional feature map, vice versa. Comparing with standard triplet loss, CRL can render CNN to precisely locate the object region, which avoids the training from overfitting. Some recent works [Oh Song *et al.*, 2017; Ming *et al.*, 2017] also employ class center in loss function. However, these methods require huge computational complexity and are hard to implement. Moreover, training with fully connected layers is unable to promote the quality of the saliency map for object localization.

Time Cost Implementation of the triplet loss involves $\mathcal{O}(N^3)$ computations. In contrast, training with the proposed CRL needs only $\mathcal{O}(NL^2)$, where L denotes the class number. In practice, the class number L should less than $\frac{N}{2}$ to generate triplets in a batch. In most cases, with large batch size and small class number, our method is extremely effective than previous methods, as quantitatively shown in Fig. 3.

2.2 Weakly Supervised Feature Extraction

We first coarsely localize the object via SCDA [Wei *et al.*, 2017] based salient object extraction, followed by a refinement module with Gaussian mixture models. Then, raw features are aggregated to form the final output of region-aware deep features.

Our localization scheme is inspired by [He and Peng, 2017] which adopts saliency extraction and co-segmentation for weakly-supervised object localization. However, the task in [He and Peng, 2017] is designed based upon a closed-set protocol, which differs from our task. Such a closed-set protocol pre-defines all testing identities in the training set. In contrast, FGOR is more like an open-set protocol, where the testing identities are disjoint from the training set, making the co-segmentation impossible. To tackle this issue, the

proposed weakly supervised localization targets at getting the object mask without using bounding boxes or object labels. The method consists of two stages, *i.e.*, saliency extraction and contour refinement. The first stage is to corsely localize the object by using the saliency information obtained from CNN. The second stage is to segment an accurate object mask, which further refines the coarse saliency map. The final feature is extracted by a feature aggregation operation.

Saliency Extraction. Following the principle of SCDA [Wei *et al.*, 2017], given an image I (width:n, height:m) and a CNN model, the saliency map $M \in \mathbb{R}^{m \times n}$ is computed as follows: First, an $h \times w \times c$ 3D tensor X is computed from the last convolutional layer by forward-propagation, which has the best discriminative ability and retrains certain spatial cues. This 3D tensor X is then mapped to a 2D map A by aggregating the feature map X over the third dimension c. Mathematically, the function can be defined as $\psi : \mathbb{R}^{h \times w \times c} \rightarrow \mathbb{R}^{h \times w}$ such that $\psi(X) = A$ where $A = \sum_{k=1}^{c} X_k$. After that, a mean value $\bar{a}$ of all the positions in A is calculated as the threshold to localize object. In particular, the position (i, j) whose activation response is higher than $\bar{a}$ indicates the main object. Then, a mask M of the same size A can be obtained:

$$M_{i,j} = \begin{cases} 1 & if A_{i,j} > \bar{a} \\ 0 & otherwise, \end{cases} \tag{4}$$

where (i, j) is a particular position in these $h \times w$ positions. Finally, we obtain the saliency map by resizing M from $h \times w$ to $m \times n$ by using a bilinear interpolation.

Mask Refinement. The subsequent refinement is inspired by [Carsten *et al.*, 2004] to get a more accurate object mask. According to the estimated coarse mask M, we firstly label a pixel as foreground if the mask value is 1, or background otherwise. Then two Gaussian Mixture Models (GMMs) are learned to model the foreground and background appearances, respectively, with each GMM containing $K = 5$ components. Given an image I, let θ_f be the foreground model and θ_b be the background model, and y_p denotes the pixel p of the image with a corresponding RGB value v_p. The objective function of refinement can be formulated as:

$$\max_{Y,\theta} \sum_{p} E(y_p, \theta) + \sum_{p,q} E(y_p, y_q) \tag{5}$$

$$E(y_p, \theta) = (1 - y_p) \log \big(p(v_p; \theta_b)\big) + y_p \log \big(p(v_p; \theta_f)\big), \tag{6}$$

and Y is the set of saliency assignments across the image. $E(y_p, y_q)$ is a pairwise term between pixels p and q, which enforces consistency between neighboring pixels. Based upon the coarse saliency map, Eq.5 learns and creates a precise pixel distribution of the object. Then, the unknown pixels are labelled by this distribution. The optimization process can be done by following [Carsten *et al.*, 2004]. With such an accurate segmentation, we extract discriminative features only from the segmented mask.

Feature Aggregation. The distinguishing feature should cover the object region. Given the above object segmentation, we re-extract more discriminative features as:

$$f(f_{(i,j)}, \alpha) = \begin{cases} f_{i,j} & if \quad \frac{|\Delta_{(i,j)} \bigcap M|}{|M|} > \alpha \\ 0 & otherwise, \end{cases} \tag{7}$$

Algorithm 1: Attractive Feature Extraction

Input: Training data: $\mathcal{D}_t$; Testing data: $\mathcal{D}_n$; CNN model: $\mathcal{F}$
. **Output:** Testing data features $\mathcal{T}$
. **for** *t=1,...,T* **epoch do**
  Caculate the loss according to Eq.2;
  Get gradient through Eq.3;
  Update CNN model $\mathcal{F}$ by $t^t h$ epoch data ;
end
for *i* **in** $\mathcal{D}_n$ **do**
  $X = \mathcal{F}(i)$;
  $A = \sum_{k=1}^{c} X_k$;
  generate Saliency map M_s by Eq.4;
  get pixel-wise object mask M_a by Eq.5;
  Select convolution feature using Eq.7;
  $\mathcal{T}_i$ = aggregate features by GAP & GMP;
end

where M denotes the refined object mask and $\Delta_{i,j}$ denotes the receptive field at the spatial location (i, j). We simply select the spatial feature where the intersection area between the receptive field and the object mask is large than a given threshold α, whose quantitative evaluation is given in Tab.6. Then, the feature is aggregated by a global max pooling and a global average pooling. The overall framework is summarized in Alg.1.

3 Experiments

Datasets: Both *CUB-200-2011* and *CARS196* datasets are used in evaluation. The *CUB-200-2011* [Wah *et al.*, 2011] contains 200 bird classes with 11,788 images. We employ the first 100 classes (5,864 images) for training and use the remaining 100 classes (5.924 images) for testing. The *CARS196* [Jonathan *et al.*, 2013] contains 196 car classes with 16,185 images. We employ the first 98 classes (8,054 images) for training and the remaining 100 classes (8,131 images) for testing. Both datasets have class labels and bounding box annotations, the latter of which are only used to evaluate object localization.[1]. To further evaluate the effectiveness of our method, we also conduct on some datasets whose categories are disjoint with ImageNet, e.g., Moth [Rodner *et al.*, 2015].

Evaluation Protocols: We evaluate the retrieval by the standard*Recall@K*. *Recall@K* is the average recall scores over all query images in the test set, which strictly follows the setting in [Hyun *et al.*, 2016]. Specifically, for each query, the top K similar images are returned. The recall score will be 1 if there is at least one positive image in the top K returning, and 0 otherwise. For object localization, the performance of object localization is defined by at least 50%, 60% and 70% of Intersection-over-Union (*IoU*) overlap with the ground-truth bounding box.

Baselines: We compare the proposed scheme with several state-of-the-art fine-grained image retrieval algorithms, including: (1) *Contrastive* [Bell and Bala, 2015] and *Triplet* [Wang *et al.*, 2014] methods that aim at training the feature

[1]We follow the standard train/test split in [Huang *et al.*, 2016; Bell and Bala, 2015; Hyun *et al.*, 2016; Schroff *et al.*, 2015]

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**
Our Method	**63.9**	**73.7**	**82.1**	**89.2**	**93.7**	**96.8**	**65.9**	**76.5**	**85.3**	**90.3**	**94.4**	97.0

Table 1: *Recall@K* on *CARS196* and *CUB-200-2011*. *Recall@K* is the average recall scores over all query images in the testing set. Specifically, for each query image, top K nearest images will be returned, the recall score will be 1 if at least one positive image in the return K images and 0 otherwise.

Method	recall@K					
K =	1	2	4	8	16	32
SCDA	75.9	84.3	91.7	96.1	98.0	98.9
Lift Loss	78.5	87.2	94.2	97.8	98.6	99.3
Triplet loss	77.2	86.3	93.1	97.1	98.5	99.1
Pairwise loss	76.3	86.5	93.5	97.3	98.3	99.4
Our Method	**81.8**	**89.9**	**94.9**	**97.2**	**98.6**	**99.5**

Table 2: Recall@K with different methods on *Moth*.

IoU	0.5	0.6	0.7
[Zhou *et al.*, 2016]	2.29%	1.03%	0.31%
[Karen *et al.*, 2013]	64.20%	41.08%	19.31%
[He and Peng, 2017]	65.52%	46.16%	28.36%
SCDA	72.31%	45.91%	19.55%
SCDA + CRF	26.95%	15.92%	8.96%
our method	**84.69%**	**70.58%**	**51.39%**
FCN	**86.35%**	**79.19%**	**69.44%**

Table 3: The precision of object localization on *CUB-200-2011*. The precision is defined by the proportion of Intersection-over-Union (IoU) overlap between the generated box and the ground truth at least 0.5, 0.6 and 0.7 respectively. We can get more accurate bounding boxes than other weakly supervised methods, and achieve comparable performance with supervised method like FCN.

with pairwise loss and triplet loss, respectively. The feature is extracted by using the convolution neural network and updated through back-propagation. (2) *LiftedStruct* [Hyun *et al.*, 2016] uses a novel object function to automatically find the hard examples in each training batch. The hard example usually has a large loss comparing to that of the normal pairs. (3) *Facility Location* [Oh Song *et al.*, 2017] relies on a new metric learning based on structured prediction, and is aware of the global structure of the embedding space. (4) *Histogram Loss* [Ustinova and Lempitsky, 2016] aims at penalizing the overlap between distributions of positive pairs' distances and negative pairs' distance. (5) *Binomial Deviance* [Ustinova and Lempitsky, 2016] evaluates the cost between similarities, which is proven to be robust to outliers. (6) *PDDM+Quadruplet* [Huang *et al.*, 2016] choses the hard positive examples and negative examples to update the parameters in CNN, which adopts the PDDM block to evaluate the similarities. (7) *N-pairs* [Sohn, 2016] proposed N-pairs loss which enforces softmax cross-entropy loss among the pairwise similarity values in the batch. (8) *SCDA* [Wei *et al.*, 2017] selects discriminative and representative examples in the last convolution layer of VGG-16 without further fine-tuning, which is a combination of max pooling and average pooling features.

Implementation Details. In our experiments, we apply the widely-used VGG-16 [Karen and Andrew, 2014] and initialize the weights from the network pretrained on ImageNet ILSVRC-2012 [Deng *et al.*, 2009] [2]. Due to the distinctive parts in cars, the refinement is less effective in *CARS196*. Therefore, we directly estimate object locations by SCDA [Wei *et al.*, 2017] in *CARS196*. We use the same hyperparameters in all experiments without specific tuning, with a mini-batch size of 60, a margin parameter m of 1, and an initial learning rate starting from 0.0001 and being divided by 10 in every 100-200 epochs. We extract features from the last convolutional layer of VGG-16 with the max and average pooling, and normalize the feature through $L2$ normalization. Correspondingly, the feature dimension for retrieval is 1,024.

3.1 Fine-grained Image Retrieval

Tab.1 quantifies our method on both *CARS196* and *CUB200-2011*. Without any training and object position information, both our method and SCDA [Wei *et al.*, 2017] perform better than other baselines on *CARS196* and *CUB-200-2011*, which demonstrates the importance of localizing objects. Note that PDDM+Quadruplet [Bell and Bala, 2015] proposed to crop object images with object annotations, by which cluttered backgrounds are removed. However, it performs not as good as ours, which indicates that an ideal model should not encode information from a single object only. Instead, all high response of the object should be maintained in the output feature. On *CARS196* and *CUB-200-2011*, we improve the recall score of the state-of-the-art SCDA scheme from 58.5% to 63.9%, as well as from 62.2% to

[2]Note that our scheme is compatible with other convolutional networks, the choice of which is orthogonal to our contribution.

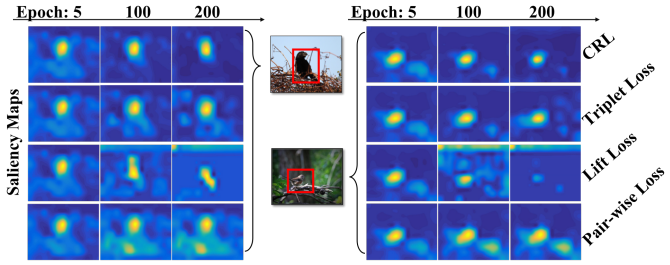

Figure 2: Saliency maps generated during the training process with different loss functions. With the proposed CRL, the saliency map is obscuring with background in epoch 5, and the discriminative part becomes more clear after 200 epochs, which imply the effectiveness of our proposed embedding method (Target objects are marked by red boxes).

Method	recall@K					
K =	1	2	4	8	16	32
Pairwise Loss	61.7	73.3	82.6	89.3	93.8	96.8
Triplet Loss	61.6	72.2	81.0	87.1	91.9	95.1
Center Loss	63.7	75.2	83.9	90.3	94.7	96.9
Lift Loss	62.4	74.5	83.9	90.2	94.0	96.8
our method	**65.9**	**76.5**	**85.3**	**90.3**	**94.4**	**97.0**

Table 4: R@K on *CUB-200-2011* with different loss. The proposed CRL is the best among all the tested loss functions.

65.9%, respectively, which validates the robustness of our method. *CARS196* and *CUB-200-2011*'s categories are joint with the ImageNet dataset. So, we have quantitative evaluated on the Moth[Rodner *et al.*, 2015] dataset. As shown in Tab.2, the proposed method still achieves the state-of-the-art performance, with 6.04 gains over SCDA, and 3.34 gains over lift loss.

3.2 Visualizing and Understanding CRL

CRL exploits ranking information to fine-tune the network through feature pooling. As presented in Sec.2.1, when updating parameters by the center feature, the associated convolution channels will be highlighted, vice versa, which makes the saliency map being attracted to the discriminative object part. We visualize the saliency map by our scheme and SCDA [Wei *et al.*, 2017] during the training iterations in Fig.2. As shown in Fig.2, the saliency map is initially obscured with the background in the 5th epoch. After only a few epochs, the discriminative part becomes more clear comparing to other loss function, which implies the effectiveness of the proposed method.

3.3 Object Localization

We explored the performance of object localization in different label conditions, and compared our method with both weakly supervised and fully supervised methods [He and Peng, 2017; Krähenbühl and Koltun, 2011; Karen *et al.*, 2013; Zhou *et al.*, 2016; Long *et al.*, 2015]. We evaluate the performance of object localization and how it affects the FGOR on *CUB-200-2011*. Tab.3 shows that we can get the best accuracy comparing to other weakly supervised methods on *CUB-200-2011* in terms of IoU. Moreover, our method is comparable to the fully-supervised FCN [Long *et al.*, 2015], which uses pixel-level annotations to train the network. In our experiments, we testify different hyper-parameters of CRF to pick up the best one, and to be integrated with SCDA+CRF. We have found that when dealing with a quite coarse saliency map such as SCDA, the dense-CRF will be confused. We explain, that more background information is included after refining SCDA with CRF, so in Tab.3, SCDA+CRF is worse than CRF.

Method	recall@K					
K =	1	2	4	8	16	32
SCDA	64.3	75.2	83.8	90.4	94.3	96.8
FCN	65.7	76.7	85.1	90.5	94.5	96.7
our method	65.9	76.5	**85.3**	90.3	94.4	**97.0**
ground truth	**66.3**	**77.2**	85.1	**90.7**	**94.7**	96.9

Table 5: Recall@K with different object localization methods on *CUB-200-2011*. In the row of ground truth, the feature is extracted by object annotations.

Threshold α	recall@K					
K =	1	2	4	8	16	32
0.10	65.9	76.0	84.8	**90.5**	**94.6**	97.0
0.20	65.6	76.7	85.2	90.4	94.6	96.9
0.40	62.0	73.6	82.3	88.8	93.6	96.4
0.80	35.7	46.1	56.7	65.3	72.4	77.1
$\alpha = 0.16$	**65.9**	**76.5**	**85.3**	90.3	94.4	**97.0**

Table 6: Recall@K with different α on *CUB-200-2011*. The α is the feature selection threshold.

In Tab.5, we combine our proposed method with different object localization methods [He and Peng, 2017; Karen *et al.*, 2013; Zhou *et al.*, 2016; Long *et al.*, 2015] on *CUB-200-2011* in terms of Recall@K. In this experiment, our method substantially outperforms other weakly supervised methods, which even approximates the fully-supervised method using ground-truth object locations. Note that the difference between the supervised method FCN and our method is quite subtle, which further proves our effectiveness. Tab.6 further shows the tuning of the hyper-parameter α, we have found that $\alpha = 0.16$ is the optimal one. We also observe that with a large α, the results would be decreased.

3.4 On Different Loss Functions

To evaluate the effectiveness of the proposed centralized ranking loss, we further replace our loss functions with different loss functions and quantify the retrieval degeneration by *Recall@K* on *CUB-200-2011*. As shown in Tab.4, our method is the best among different loss functions under the same setting of the rest components. Please note that, the center loss [Wen *et al.*, 2016] is similar to our method, which directly characterizes the intra-class variations. Our proposed loss considers the variations in intra-class and inter-class simultaneously. As a result, our loss can achieve better

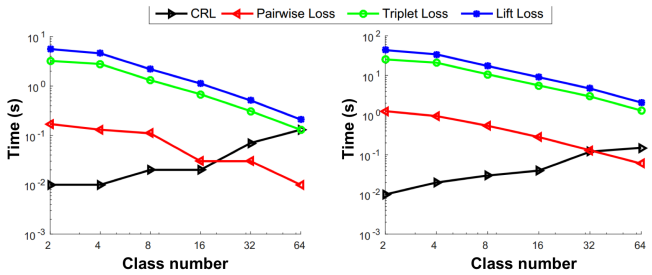

Figure 3: The running time with different class number L and batch size 128 (left), 256 (right). The horizontal axis is the class number L within each batch, and the vertical axis is the running time. So our CRL scheme is extremely effective in training, *i.e.*, 1,000 times speedup comparing to triplet loss when the class number is 2.

performance comparing to center loss. In Fig.3, we further test the layer-wise training time with respect to different class numbers L and batch-size n. The high-order loss functions are time consuming, particular with large batch size and small class number. Instead, our scheme is extremely effective in training, *i.e.*, 1,000 times speedup comparing to triplet loss when the class number is 2. Interestingly, differing from other loss functions, the training complexity of our loss function decrease with the class number, which further remedies the complexity when limited amount of object classes is needed.

4 Conclusions

This paper contributers to the filed of FGOR in two-fold: First, we propose a centralized ranking loss, which achieves a very efficient training (1,000 times speedup for training comparing to triplet loss). Second, we propose an effective weakly supervised framework which precisely locate objects without bounding box or contour supervision. We achieve the best results on *CUB-200-2011* and *CARS196* dataset comparing to a variety of state-of-the-arts. In the future, this work will be pushed forward. First, since the key differences in FGOR only reveal on small parts, we will exploit new methods for discovering part-level salient regions such as head, torso, or claws. Second, we will combine features from different layers to obtains more discriminative representation.

Acknowledgements

This work is supported by the National Key R&D Program (No.2017YFC0113000, and No.2016YFB1001503), Nature Science Foundation of China (No.U1705262, No.61772443, and No.61572410), Post Doctoral Innovative Talent Support Program under Grant BX201600094, China Post-Doctoral Science Foundation under Grant 2017M612134, Scientific Research Project of National Language Committee of China (Grant No. YB135-49), and Nature Science Foundation of Fujian Province, China (No. 2017J01125 and No. 2018J01106).

References

[Bell and Bala, 2015] Sean Bell and Kavita Bala. Learning visual similarity for product design with convolutional neural networks. *ACM Transactions on Graphics (TOG)*, 34(4):98, 2015.

[Carsten *et al.*, 2004] Rother Carsten, Kolmogorov Vladimir, and Blake Andrew. Grabcut: Interactive foreground extraction using iterated graph cuts. In *ACM transactions on graphics (TOG)*, volume 23, pages 309–314. ACM, 2004.

[Deng *et al.*, 2009] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009.

[He and Peng, 2017] Xiangteng He and Yuxin Peng. Weakly supervised learning of part selection model with spatial constraints for fine-grained image classification. In *Proceedings of the Eleventh National Conference on Artificial Intelligence*, pages 4075–4081, 2017.

[Huang *et al.*, 2016] Chen Huang, Chen Change Loy, and Xiaoou Tang. Local similarity-aware deep feature embedding. In *Advances in Neural Information Processing Systems*, pages 1262–1270, 2016.

[Hyun *et al.*, 2016] Oh Song Hyun, Xiang Yu, Jegelka Stefanie, and Savarese Silvio. Deep metric learning via lifted structured feature embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4004–4012, 2016.

[Jonathan *et al.*, 2013] Krause Jonathan, Stark Michael, Deng Jia, and Fei-Fei Li. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 554–561, 2013.

[Jonathan *et al.*, 2015] Krause Jonathan, Hailin Jin, Jianchao Yang, and Fei-Fei Li. Fine-grained recognition without part annotations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5546–5555, 2015.

[Karen and Andrew, 2014] Simonyan Karen and Zisserman Andrew. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[Karen *et al.*, 2013] Simonyan Karen, Vedaldi Andrea, and Zisserman Andrew. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[Krähenbühl and Koltun, 2011] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in neural information processing systems*, pages 109–117, 2011.

[Long *et al.*, 2015] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[Ming *et al.*, 2017] Zuheng Ming, Joseph Chazalon, Muhammad Muzzamil Luqman, Muriel Visani, and Jean-Christophe Burie. Simple triplet loss based on intra/inter-class metric learning for face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1656–1664, 2017.

[Oh Song *et al.*, 2017] Hyun Oh Song, Stefanie Jegelka, Vivek Rathod, and Kevin Murphy. Deep metric learning via facility location. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[Rodner *et al.*, 2015] Erik Rodner, Marcel Simon, Gunnar Brehm, Stephanie Pietsch, J Wolfgang Wägele, and Joachim Denzler. Fine-grained recognition datasets for biodiversity analysis. *arXiv preprint arXiv:1507.00913*, 2015.

[Schroff *et al.*, 2015] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 815–823, 2015.

[Simon and Rodner, 2015] Marcel Simon and Erik Rodner. Neural activation constellations: Unsupervised part model discovery with convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1143–1151, 2015.

[Sohn, 2016] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *Advances in Neural Information Processing Systems*, pages 1857–1865, 2016.

[Ustinova and Lempitsky, 2016] Evgeniya Ustinova and Victor Lempitsky. Learning deep embeddings with histogram loss. In *Advances in Neural Information Processing Systems*, pages 4170–4178, 2016.

[Wah *et al.*, 2011] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[Wang *et al.*, 2014] Jiang Wang, Yang Song, Thomas Leung, Chuck Rosenberg, Jingbin Wang, James Philbin, Bo Chen, and Ying Wu. Learning fine-grained image similarity with deep ranking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1386–1393, 2014.

[Wei *et al.*, 2017] Xiu-Shen Wei, Jian-Hao Luo, Jianxin Wu, and Zhi-Hua Zhou. Selective convolutional descriptor aggregation for fine-grained image retrieval. *IEEE Transactions on Image Processing*, 26(6):2868–2881, 2017.

[Wen *et al.*, 2016] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *European Conference on Computer Vision*, pages 499–515. Springer, 2016.

[Xiao *et al.*, 2015] Tianjun Xiao, Yichong Xu, Kuiyuan Yang, Jiaxing Zhang, Yuxin Peng, and Zheng Zhang. The application of two-level attention models in deep convolutional neural network for fine-grained image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 842–850, 2015.

[Xie *et al.*, 2013] Lingxi Xie, Qi Tian, Richang Hong, Shuicheng Yan, and Bo Zhang. Hierarchical part matching for fine-grained visual categorization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1641–1648, 2013.

[Xie *et al.*, 2015] Lingxi Xie, Jingdong Wang, Bo Zhang, and Qi Tian. Fine-grained image search. *IEEE Transactions on Multimedia*, 17(5):636–647, 2015.

[Zhang *et al.*, 2016a] Xiaofan Zhang, Feng Zhou, Yuanqing Lin, and Shaoting Zhang. Embedding label structures for fine-grained feature representation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1114–1123, 2016.

[Zhang *et al.*, 2016b] Xiaopeng Zhang, Hongkai Xiong, Wengang Zhou, Weiyao Lin, and Qi Tian. Picking deep filter responses for fine-grained image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1134–1142, 2016.

[Zhou *et al.*, 2016] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2921–2929, 2016.