EMBEDDING TRANSFER VIA SMOOTH CONTRASTIVE LOSS

Anonymous authors
Paper under double-blind review

ABSTRACT

This paper presents a novel method for *embedding transfer*, a task of transferring knowledge of a learned embedding model to another. Our method exploits pairwise similarities between samples in the source embedding space as the knowledge, and transfers it through a loss function used for learning target embedding models. To this end, we design a new loss called *smooth contrastive loss*, which pulls together or pushes apart a pair of samples in a target embedding space with strength determined by their semantic similarity in the source embedding space; an analysis of the loss reveals that this property enables more important pairs to contribute more to learning the target embedding space. Experiments on metric learning benchmarks demonstrate that our method improves performance, or reduces sizes and embedding dimensions of target models effectively. Moreover, we show that deep networks trained in a self-supervised manner can be further enhanced by our method with no additional supervision. In all the experiments, our method clearly outperforms existing embedding transfer techniques.

1 INTRODUCTION

Learning an embedding space where semantically similar samples are grouped together has played important roles in many tasks including data retrieval (Movshovitz-Attias et al., 2017; Song et al., 2016; Sohn, 2016; Kim et al., 2019; 2020), few-shot learning (Snell et al., 2017; Sung et al., 2018; Qiao et al., 2019), zero-shot learning (Bucher et al., 2016; Zhang & Saligrama, 2016), and self-supervised representation learning (Tian et al., 2019; Chen et al., 2020a; He et al., 2020). In these tasks, performance and efficiency of models rely heavily on the quality and dimension of their learned embedding spaces. To obtain high-quality and compact embedding spaces, previous methods have proposed new loss functions (Song et al., 2016; Sohn, 2016; Yu & Tao, 2019; Wang et al., 2019; Movshovitz-Attias et al., 2017; Kim et al., 2020), advanced sampling strategies (Wu et al., 2017; Harwood et al., 2017; Wang et al., 2020; Ko & Gu, 2020), regularization techniques (Jacob et al., 2019; Mohan et al., 2020), or ensemble models (Opitz et al., 2017; 2018; Kim et al., 2018).

For the same purpose, we study transferring knowledge of a learned embedding model (source) to another (target), which we call *embedding transfer*. The knowledge captured by the source embedding model can provide semantic information beyond class labels such as intra-class variations and degrees of semantic affinity between samples. Also, given a proper way to transfer such knowledge, embedding transfer enables us to improve performance of target embedding models or compress them effectively, as knowledge distillation does for classification models (Hinton et al., 2015; Romero et al., 2014; Zagoruyko & Komodakis, 2016; Yim et al., 2017; Furlanello et al., 2018). The two main factors of embedding transfer, the type of knowledge and the way to transfer, thus have to be carefully designed for the success of this task.

Previous methods on embedding transfer extract knowledge of a source embedding space in forms of probability distributions of samples (Passalis & Tefas, 2018), their geometric relations (Park et al., 2019; Yu et al., 2019), or the rank of their similarities (Chen et al., 2017). Then the knowledge is transferred by forcing target models to approximate those extracted patterns directly in their embedding spaces. Although these methods shed light on the problem of embedding transfer, there is room for further improvement since they fail to utilize detailed inter-sample relations in the source embedding space (Passalis & Tefas, 2018; Chen et al., 2017) or blindly accept the transferred knowledge without considering relative importance of the samples (Park et al., 2019; Yu et al., 2019).

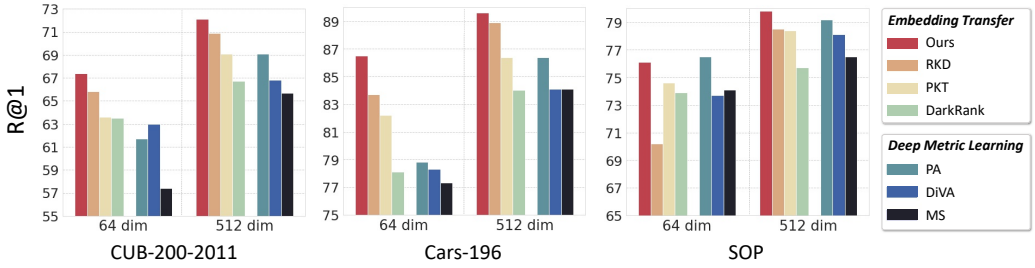

Figure 1: Accuracy in Recall@1 on the three standard benchmarks for metric learning. All embedding transfer methods adopt PA (Kim et al., 2020) with 512 dimension as the source model. Our method achieves state of the art when embedding dimension is 512, and is as competitive as recent metric learning models even with a substantially smaller embedding dimension. In all experiments, it is superior to other embedding transfer techniques. More results can be found in Table 1 and 2.

This paper presents a new embedding transfer method that overcomes the above limitations. Our method makes use of pairwise similarities between samples in a source embedding space as the knowledge to be transferred. Pairwise similarities are useful to characterize an embedding space in detail, thus have been widely used for learning embedding spaces (Hadsell et al., 2006; Schroff et al., 2015; Sohn, 2016; Wang et al., 2019) and identifying underlying manifolds of data (Cox & Cox, 2008; Tenenbaum et al., 2000). Also, they capture detailed inter-sample relations, which are missing in probability distributions (Passalis & Tefas, 2018) and the rank of similarities (Chen et al., 2017) used as knowledge in previous work.

The knowledge is in turn transferred through a loss function that is used for learning target embedding models. To this end, we propose a new loss called *smooth contrastive loss*. The proposed loss pushes apart or pulls together a pair of samples in a target embedding space, where their semantic similarity in the source embedding space determines the strength of pushing and pulling. Our analysis reveals that this property enables more important sample pairs to contribute more to learning the target embedding space, thus resolves the limitation of previous methods that treat samples equally during transfer (Park et al., 2019; Yu et al., 2019). For further improvement, we also present a data augmentation strategy, which allows to transfer semantic relations between multiple views of each sample as well as those between different samples.

The efficacy of the proposed method is demonstrated on two different tasks, deep metric learning and self-supervised representation learning. In metric learning experiments, our method substantially improves image retrieval performance when the target model has the same architecture with the source model, and greatly reduces the size and embedding dimension of the target model with a negligible performance drop when the target model is smaller than the source model, as illustrated in Fig. 1. We also show that deep networks trained in a self-supervised manner (Chen et al., 2020a; He et al., 2020) can be further enhanced by self embedding transfer with our method, analogous to the born-again network (Furlanello et al., 2018) yet with no supervision. In all the experiments, our method outperforms existing embedding transfer techniques (Passalis & Tefas, 2018; Park et al., 2019; Chen et al., 2017).

2 Related Work

Learning embedding spaces. Deep metric learning is an approach to learning embedding spaces using class labels. Previous work in this field has developed loss functions for modeling inter-sample relations based on class labels and reflecting them on the learned embedding spaces. Contrastive loss (Chopra et al., 2005; Hadsell et al., 2006) pulls a pair of samples together if their class labels are the same and pushes them away otherwise. Triplet loss (Wang et al., 2014; Schroff et al., 2015) takes a triplet of anchor, positive, and negative as input, and makes the anchor-positive distance smaller than the anchor-negative distance. The idea of pushing and pulling a pair is extended to consider higher order relations in recently proposed losses (Sohn, 2016; Song et al., 2016; Wang et al., 2019). Meanwhile, self-supervised representation learning has been greatly advanced by leveraging pairwise relations between data as in deep metric learning. For example, MoCo (He

et al., 2020; Chen et al., 2020b) and SimCLR (Chen et al., 2020a) pull embedding vectors of the same image closer and push those of different images away. Since these approaches to learning embedding spaces demand binary relations, i.e., the equality of classes or identities, they cannot be used directly for transferring knowledge of an embedding space that is not binary.

Knowledge distillation. Knowledge distillation means a technique that transfers knowledge of a source model to a target model; embedding transfer can be regarded as its particular example focusing on embedding models. A seminal work by Hinton et al. (2015) achieves this goal by encouraging the target model to imitate class logits of the source model, and has been extended to transfer various types of knowledge of the source model (Romero et al., 2014; Zagoruyko & Komodakis, 2016; Yim et al., 2017; Ahn et al., 2019). Knowledge distillation has been employed for various purposes including model compression (Hinton et al., 2015; Romero et al., 2014; Zagoruyko & Komodakis, 2016; Yim et al., 2017), cross-modality learning (Tian et al., 2019), and network regularization (Yun et al., 2020) as well as performance improvement (Furlanello et al., 2018). In terms of target task, however, it has been applied mostly to classification; only a few methods introduced in the next paragraph study transferring knowledge of embedding spaces, i.e., embedding transfer.

Embedding transfer. Early approaches in this area extract and transfer the rank of similarities between samples (Chen et al., 2017) and probability distributions of their similarities (Passalis & Tefas, 2018) in the source embedding spaces. Unfortunately, these methods have trouble in capturing elaborate relations between samples. Meanwhile, recent methods utilize geometric relations between samples like distances and angles as the knowledge to take fine details of the source embedding space into account (Park et al., 2019; Yu et al., 2019). However, they let the target model blindly accept the knowledge without considering relative importance of samples, leading to less effective embedding transfer. Our method overcomes the aforementioned limitations: It makes use of rich pairwise similarities between samples as the knowledge, and the proposed loss enables to take relative importance of samples into account when transferring the knowledge.

3 Proposed Method

This section first reviews the original contrastive loss (Hadsell et al., 2006), the prototype of the proposed one. Then the derivation of the smooth contrastive loss and the multi-view data augmentation strategy are described in detail.

3.1 Revisiting Original Contrastive Loss

Contrastive loss (Hadsell et al., 2006) is one of the most representative losses for learning semantic embedding by leveraging pairwise relations of samples. Let $f_i := f(x_i)$ be the embedding vector of input data x_i produced by the embedding network f, and $D(f_i, f_j)$ denote the Euclidean distance between embedding vectors f_i and f_j. The contrastive loss is then formulated as

$$\mathcal{L}(X) = \underbrace{\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{n} y_{ij} D(f_i, f_j)^2}_{\textit{attracting}} + \underbrace{\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{n} (1 - y_{ij}) \big[\delta - D(f_i, f_j)\big]_+^2}_{\textit{repelling}}, \tag{1}$$

where X is a batch of embedding vectors, n is the number of samples in the batch, δ is a margin, and $[\cdot]_+$ denotes the hinge function. Also, y_{ij} indicates the class equivalence between the pair of samples (i, j): $y_{ij} = 1$ if the pair is of the same class (i.e., positive pair), and 0 otherwise (i.e., negative pair). Note that all embedding vectors are l_2 normalized to prevent the margin from becoming trivial. This loss consists of two constituents, an attracting term and a repelling term. In the embedding space, the attracting term forces positive pairs to be closer, and the repelling term encourages to push negative pairs apart beyond the margin.

The gradient of the contrastive loss with respect to $D(f_i, f_j)$ is given by

$$\frac{\partial \mathcal{L}(X)}{\partial D(f_i, f_j)} = \begin{cases} \dfrac{2}{n} D(f_i, f_j), & \text{if } y_{ij} = 1, \\ -\dfrac{2}{n}\big\{\delta - D(f_i, f_j)\big\}, & \text{else if } y_{ij} = 0 \text{ and } D(f_i, f_j) < \delta, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

As shown in Eq. (2), the gradient increases as the distance of a positive pair increases or the distance of a negative pair decreases. When the distance of a negative pair is larger than the margin δ, the gradient becomes 0.

3.2 SMOOTH CONTRASTIVE LOSS

The basic idea of the smooth contrastive loss is to pull or push a pair of samples in the target embedding space according to their semantic similarity. The loss is inspired by the original contrastive loss, which aims to pull samples together if their class labels are the same and push them apart otherwise. To reflect the knowledge, however, the smooth contrastive loss utilizes the pairwise similarities between samples in the source embedding space instead of their class labels.

This idea can be implemented by replacing the class equivalence indicator y_{ij} of the original contrastive loss with the semantic similarity between x_i and x_j in the source embedding space. The loss then becomes a linear combination of the attracting and repelling terms, in which their weights are proportional to the semantic similarities. Specifically, it is formulated as

$$\mathcal{L}(X) = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}^{s} D(f_i^t, f_j^t)^2 + \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{n}(1 - w_{ij}^{s})\Big[\delta - D(f_i^t, f_j^t)\Big]_+^2, \tag{3}$$

where w_{ij}^s denotes the weights derived from the semantic similarities in the source embedding space, and $f_i^t := f^t(x_i) \in \mathbb{R}^d$ indicates the embedding vector of input x_i produced by the target embedding model f^t. For computing the weight terms, we employ a Gaussian kernel based on the Euclidean distance as follows:

$$w_{ij}^{s} = K_G(f_i^s, f_j^s; \sigma) = \exp\left(-\frac{||f_i^s - f_j^s||_2^2}{\sigma}\right) \in [0, 1], \tag{4}$$

where σ is kernel bandwidth, $f_i^s := f^s(x_i)$ indicates the embedding vector of input x_i given by the source embedding model f^s, and $||\cdot||_2$ denotes l_2 norm of vector.

Eq. (3) shows that the strength of pulling or pushing embedding vectors is now controlled by the weights in the new loss function. In the target embedding space, a pair of samples that the source embedding model regards more similar attract each other more strongly while those considered more dissimilar are pushed more heavily out of the margin δ. This behavior of the loss can be explained through its gradient, which is given by

$$\frac{\partial \mathcal{L}(X)}{\partial D(f_i^t, f_j^t)} = \begin{cases} \frac{2}{n}\big\{D(f_i^t, f_j^t) - \delta(1 - w_{ij}^s)\big\}, & \text{if } D(f_i^t, f_j^t) < \delta, \\ \frac{2}{n} w_{ij}^s D(f_i^t, f_j^t), & \text{otherwise.} \end{cases} \tag{5}$$

Unlike the original one, the aspect of our loss gradient depends on the transferred knowledge w_{ij}^s, thus the force of pushing a pair (i, j) apart and that of pulling them together are determined by both of $D(f_i^t, f_j^t)$ and w_{ij}^s. In the ideal case, $D(f_i^t, f_j^t)$ will converge to $\delta(1 - w_{ij}^s)$, which is the semantic dissimilarity scaled by δ, and where the two forces are balanced.

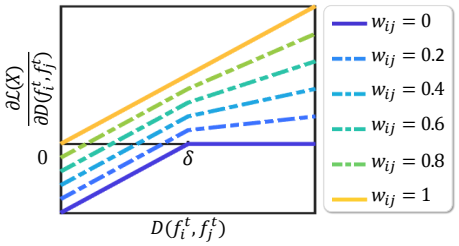

Figure 2: Gradient of the smooth contrastive loss versus pairwise distance.

This aspect of gradient also differentiates our method from the previous arts that imitate the knowledge through regression losses (Park et al., 2019; Yu et al., 2019). As illustrated in Fig. 2, the proposed loss rarely cares about a pair (i, j) when its distance is large in both of the source and target spaces, i.e., $w_{ij}^s \approx 0$ and $D(f_i^t, f_j^t) > \delta$, as its loss gradient is close to 0. This behavior can be interpreted as that our loss disregards less important pairs to focus on more important ones. Recall that what we expect from a learned embedding space is that *nearby* samples are semantically similar in the space; if the distance of a semantically dissimilar pair is sufficiently large, it does not impair such a quality of the embedding space and can be regarded as less

important consequently. On the other hand, previous methods using regression losses handle samples equivalently without considering their relative importance (Park et al., 2019; Yu et al., 2019), leading to embedding transfer less effective.

The loss in Eq. (3) takes advantage of the rich semantic information of the source embedding space in a flexible and effective manner, but it still has a problem to be resolved: It imposes a restriction on the manifold of the target space since it demands l_2 normalization of the embedding vectors to prevent the divergence of their magnitudes and to keep the margin non-trivial, as in the original contrastive loss. To resolve this issue, we replace the pairwise distances of the loss in Eq. (3) with their *relative* versions, then the final form of the smooth contrastive loss is given by

$$\mathcal{L}(X) = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}^{s}\left\{\frac{D(f_i^t, f_j^t)}{\mu_i}\right\}^2 + \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{n}(1-w_{ij}^{s})\left[\delta - \frac{D(f_i^t, f_j^t)}{\mu_i}\right]_+^2, \quad \text{where } \mu_i = \frac{1}{n}\sum_{k=1}^{n} D(f_i^t, f_k^t). \tag{6}$$

The relative distance between f_i^t and f_j^t is their pairwise distance divided by μ_i, the average distance of all pairs associated with f_i^t in the batch. Since scales of pairwise distances are roughly cancelled in their relative versions, the above loss can alleviate the aforementioned normalization issue. Thus, although the source embedding space is limited to the surface of unit hypersphere due to the l_2 normalization, the target embedding model can exploit the entire space of $\mathbb{R}^d$ with no restriction on its manifold; this advantage enables to utilize the given embedding dimension more effectively.

3.3 Multi-View Augmentation Strategy

Inspired by the recent self-supervised learning methods, we present a simple yet effective multi-view augmentation strategy for enhancing the effect of embedding transfer. We first apply random image augmentation to each sample in the batch to produce its multiple views. By taking as input the produced multi-view samples, our loss can take into account semantic relations between the multiple views of individual samples as well as those between different samples. This augmentation strategy is useful for embedding transfer since it diversifies the information of input samples and allows to consider fine-grained relations between multi-view samples produced from the same image. The empirical advantage of the multi-view augmentation is verified by experiments, where it improves stability and convergence of embedding transfer as well as performance of target embedding models. More details of the multi-view augmentation strategy is describe in Appendix A.2.

4 Experiments

The effectiveness of embedding transfer by our method is demonstrated in two different tasks, deep metric learning and self-supervised representation learning. In both of the tasks, we measure the performance of target models that are trained soley by embedding transfer techniques incorporated with source models pretrained for the tasks; no other supervision is introduced for the target models. In these experiments, the proposed method is compared with existing embedding transfer techniques, RKD (Park et al., 2019), PKT (Passalis & Tefas, 2018), and DarkRank (Chen et al., 2017).

4.1 Deep Metric Learning

We evaluate and compare target models trained by embedding transfer methods including ours on standard benchmarks for metric learning. The experiments are conducted in the following three settings by varying the type of target embedding model. *(i) Self-transfer for performance improvement*: Transfer to a model with the same architecture and embedding dimension. *(ii) Dimensionality reduction*: Transfer to the same architecture with a lower embedding dimension. *(iii) Model compression*: Transfer to a smaller network with a lower embedding dimension.

4.1.1 Setup

Datasets and evaluation. Models trained through embedding transfer are evaluated in terms of image retrieval performance on the CUB200-2011 (Welinder et al., 2010), Cars-196 (Krause et al.,

Table 1: Image retrieval performance of embedding transfer methods in the three different settings: (a) Self-transfer, (b) dimensionality reduction, and (c) model compression. Embedding networks of the methods are denoted by abbreviations: BN–Inception with BatchNorm, R50–ResNet50, R18–ResNet18. Superscripts indicate embedding dimensions of the networks.

			CUB-200-2011			Cars-196			SOP		
Recall@K			1	2	4	1	2	4	1	10	100
(a)	*Source*: PA (Kim et al., 2020)	BN^{512}	69.1	78.9	86.1	86.4	91.9	95.0	79.2	90.7	96.2
	DarkRank (Chen et al., 2017)	BN^{512}	66.7	76.5	84.8	84.0	90.0	93.8	75.7	88.3	95.3
	PKT (Passalis & Tefas, 2018)	BN^{512}	69.1	78.8	86.4	86.4	91.6	94.9	78.4	90.2	96.0
	RKD (Park et al., 2019)	BN^{512}	70.9	80.8	87.5	88.9	93.5	96.4	78.5	90.2	96.0
	Ours w/o augmentation	BN^{512}	71.5	81.3	87.6	89.0	93.6	96.3	79.6	91.0	96.2
	Ours	BN^{512}	**72.1**	**81.3**	**87.6**	**89.6**	**94.0**	**96.5**	**79.8**	**91.1**	**96.3**
(b)	*Source*: PA (Kim et al., 2020)	BN^{512}	69.1	78.9	86.1	86.4	91.9	95.0	79.2	90.7	96.2
	DarkRank (Chen et al., 2017)	BN^{64}	63.5	74.3	83.1	78.1	85.9	91.1	73.9	87.5	**94.8**
	PKT (Passalis & Tefas, 2018)	BN^{64}	63.6	75.8	84.0	82.2	88.7	93.5	74.6	87.3	94.2
	RKD (Park et al., 2019)	BN^{64}	65.8	76.7	85.0	83.7	89.9	94.1	70.2	83.8	92.1
	Ours w/o augmentation	BN^{64}	67.1	77.8	85.8	85.2	91.0	95.0	75.7	86.2	94.7
	Ours	BN^{64}	**67.4**	**78.0**	**85.9**	**86.5**	**92.3**	**95.3**	**75.9**	**88.3**	94.6
(c)	*Source*: PA (Kim et al., 2020)	$R50^{512}$	69.9	79.6	88.6	87.7	92.7	95.5	80.5	91.8	98.8
	DarkRank (Chen et al., 2017)	$R18^{128}$	61.2	72.5	82.0	75.3	83.6	89.4	72.7	86.7	94.5
	PKT (Passalis & Tefas, 2018)	$R18^{128}$	65.0	75.6	84.8	81.6	88.8	93.4	76.9	89.2	95.5
	RKD (Park et al., 2019)	$R18^{128}$	65.8	76.3	84.8	84.2	90.4	94.3	75.7	88.4	95.1
	Ours w/o augmentation	$R18^{128}$	66.4	77.4	85.3	84.5	91.0	94.9	77.8	90.0	95.8
	Ours	$R18^{128}$	**66.6**	**78.1**	**85.9**	**86.0**	**91.6**	**95.3**	**78.4**	**90.4**	**96.1**

2013) and SOP datasets (Song et al., 2016). Each dataset is split into training and testing sets following the standard setting of Song et al. (2016). As a performance measure, we adopt Recall@K that counts how many queries have at least one correct sample among their K nearest neighbors in learned embedding spaces.

Source and target embedding networks. For the *self-transfer* and *dimensionality reduction* experiments, we employ Inception with BatchNorm (Ioffe & Szegedy, 2015) with 512 output dimension as the source embedding model. Target embedding models for the two settings basically have the same architecture with the source embedding model, but for *dimensionality reduction*, the output dimension is reduced to 64. On the other hand, in the *model compression* experiment, we adopt ResNet50 (He et al., 2016) with 512 output dimension as the source embedding model, and ResNet18 (He et al., 2016) with 128 output dimension as the target embedding model. In all the three settings, the source models are trained by the proxy-anchor loss (Kim et al., 2020) with l_2 normalization of embedding vectors, while the target models are pre-trained for the ImageNet classification task (Deng et al., 2009) and have no l_2 normalization applied.

Implementation details. We train all models using the AdamW optimizer (Loshchilov & Hutter, 2019) with the cosine learning decay (Loshchilov & Hutter, 2016) and initial learning rate of 10^{-4}. The models are trained for 90 epochs in the CUB-200-2011 and Cars-196 datasets and 150 epochs on the SOP dataset. Training images are randomly cropped to 224×224 with random horizontal flip, and testing images are center-cropped after being resized to 256×256. The multi-view augmentation strategy applies random augmentation operation twice per image so that the input batch contains two different views for each image. We set both δ and σ in our loss to 1 for all the experiments.

4.1.2 RESULTS

The proposed method is evaluated and compared with existing embedding transfer techniques and state-of-the-art metric learning models on the three benchmark datasets. We also report the performance of our method without the multi-view data augmentation strategy to demonstrate its effectiveness. The results are summarized in Table 1 and 2.

In the *self-transfer* setting (Table 1(a)), the proposed method notably improves retrieval performance and clearly surpasses the state of the art on all the datasets without bells and whistles (Table 2); the effect of embedding transfer by our method is qualitatively demonstrated in Fig. 3. On the other

Table 2: Image retrieval performance of the proposed method and the state-of-the-art metric learning models. Embedding networks of the methods are fixed by Inception with BatchNorm (BN) for fair comparisons, and superscripts indicate embedding dimensions of the networks.

		CUB-200-2011			Cars-196			SOP		
Recall@K		1	2	4	1	2	4	1	10	100
MS (Wang et al., 2019)	BN^{64}	57.4	69.8	80.0	77.3	85.3	90.5	74.1	87.8	94.7
DiVA (Milbich et al., 2020)	BN^{64}	63.0	74.5	83.3	78.3	86.6	91.2	73.7	87.5	94.8
PA (Kim et al., 2020)	BN^{64}	61.7	73.0	81.8	78.8	87.0	92.2	**76.5**	**89.0**	**95.1**
Ours	BN^{64}	**67.4**	**78.0**	**85.9**	**86.5**	**92.3**	**95.3**	75.9	88.3	94.6
MS (Wang et al., 2019)	BN^{512}	65.7	77.0	86.3	84.1	90.4	94.0	78.2	90.5	96.0
DiVA (Milbich et al., 2020)	BN^{512}	66.8	77.7	-	84.1	90.7	-	78.1	90.6	-
PA (Kim et al., 2020)	BN^{512}	69.1	78.9	86.1	86.4	91.9	95.0	79.2	90.7	96.2
Ours	BN^{512}	**72.1**	**81.3**	**87.6**	**89.6**	**94.0**	**96.5**	**79.8**	**91.1**	**96.3**

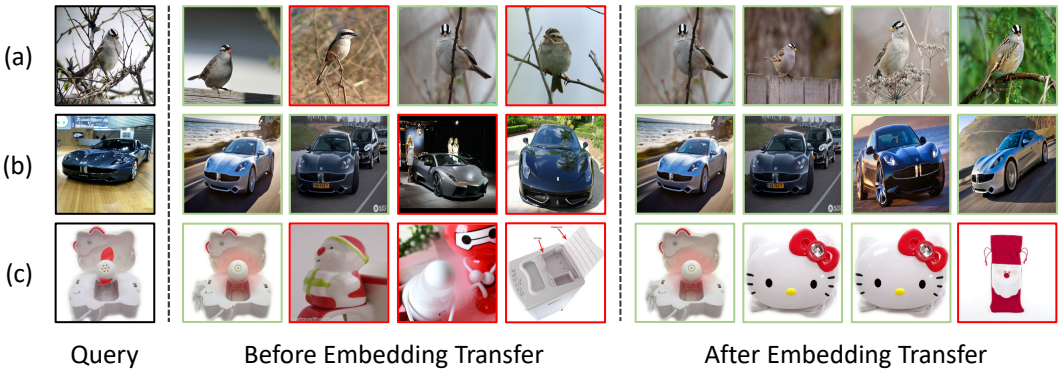

Figure 3: Top 4 image retrievals of the state of the art (Kim et al., 2020) before and after the proposed method is applied. (a) CUB-2020-2011. (b) Cars-196. (c) SOP. Images with green boundary are success cases and those with red boundary are false positives. More qualitative results can be found in Appendix A.3.

hand, the performance of existing embedding transfer methods is inferior to that of the source model on the SOP dataset. The proposed method demonstrates more interesting results in the *dimensionality reduction* setting (Table 1(b)): It outperforms recent metric learning methods, MS and DiVA, whose embedding dimension is 8 times higher (Table 2). This result enables significant speedup of image retrieval systems at the cost of a tiny performance drop. Finally, in the *model compression* setting (Table 1(c)), our method achieves impressive performance even with a substantially smaller network and a lower embedding dimension; the performance drop by the compression is marginal and its accuracy is as competitive as MS with a heavier network and a larger embedding dimension. Note that, the proposed method is superior to other embedding transfer techniques in every experiment, and the multi-view augmentation strategy improves performance in most cases.

4.2 Self-supervised Representation Learning

Knowledge distillation has been known to improve performance of classification networks through self-transfer (Furlanello et al., 2018), but is not available for self-supervised representation learning due to the absence of class label. We argue that embedding transfer can play this role for self-supervised networks since it distills and transfers knowledge without relying on class labels.

This section examines potential of embedding transfer methods in this context, by learning representations using the embedding transfer methods and the knowledge extracted from existing self-supervised networks. Our method is compared with RKD (Park et al., 2019) and PKT (Passalis & Tefas, 2018), but DarkRank (Chen et al., 2017) is excluded since its complexity, proportional to the number of sample permutations, is excessively large in the self-supervised learning setting. To verify universality of the methods, we incorporate them with two different self-supervised representation learning frameworks, SimCLR (Chen et al., 2020a) and MoCo (Chen et al., 2020b).

Table 3: Performance of linear classifiers trained on representations obtained by embedding transfer techniques incorporated with self-supervised learning frameworks.

	CIFAR-10		STL-10
Source model	SimCLR	MoCo v2	SimCLR
Before embedding transfer	93.4	86.8	89.2
PKT (Passalis & Tefas, 2018)	65.3	47.6	71.6
RKD (Park et al., 2019)	<u>93.6</u>	**87.7**	<u>79.8</u>
Ours	**93.9**	<u>87.7</u>	**89.6**

4.2.1 SETUP

Datasets and evaluation. Self-supervised models and those enhanced by embedding transfer are evaluated on the CIFAR-10 (Krizhevsky & Hinton, 2009) and STL-10 (Coates et al., 2011) datasets. In the STL-10 dataset, both of the labeled training set and unlabeled set are used for training, and the rest are kept for testing. Performance of the models is measured by the linear evaluation protocol (Zhang et al., 2016; Bachman et al., 2019; Zhai et al., 2019), in which a linear classifier on top of a frozen self-supervised network is trained and evaluated.

Source models and their training. We reimplement SimCLR (Chen et al., 2020a) and MoCo v2 (Chen et al., 2020b) frameworks to train source embedding models. Following the original frameworks, ResNet50 is employed as the based network of source models and a Multi-Layer Perceptron (MLP) head is appended to its last pooling layer. On the CIFAR-10 dataset, source models are trained for 1K epochs while following the details (e.g., augmentation, learning rate, and temperature) described in Chen et al. (2020a). On the STL-10 dataset, we adopt the same configuration except that Gaussian blur is additionally employed for data augmentation.

Target models and their training. For training of target models, the MLP on top of the source models are removed and their embedding vectors are l_2 normalized. Target models have the same architecture with their source counterpart where the MLP head is removed, but with no l_2 normalization. Details of training target models are the same with those for the corresponding source models. All target models are trained using the LARS optimizer (You et al., 2017) with initial learning rate of 4.0 and weight decay of 10^{-6}. We warm up the learning rate linearly during the first 10 epochs and apply the cosine decay (Loshchilov & Hutter, 2016) to it after that. Regarding hyperparameters, both of δ and σ in our loss are set to 1 in this experiment also.

4.2.2 RESULTS

Performance of embedding transfer methods in the self-supervised learning task is summarized in Table 3. The proposed method improves the quality of the learned representations on both datasets and for both self-supervised learning frameworks. Moreover, our method clearly outperforms the existing embedding transfer techniques when incorporated with SimCLR while achieving the second best by a narrow margin when applied to MoCo v2. In contrast, other embedding transfer methods are often inferior to the source model, and especially PKT shows unstable performance in every experiment. This is because of their limitations: As the batch size increases, the probability distributions considered by PKT becomes nearly uniform, and the computational burden of RKD grows significantly due to its angle calculation. Our method enhances the performance of the existing self-supervised models without such difficulties.

5 CONCLUSION

We have presented a novel loss and a data augmentation strategy to distill and transfer knowledge of a learned embedding model effectively. Our loss utilizes rich pairwise relations between samples in the source embedding space as the knowledge, and effectively transfers the knowledge by focusing more on sample pairs important for learning target embedding models. As a result, our method has achieved impressive performance over the state of the art on the deep metric learning benchmarks and demonstrated that it can reduce the size and embedding dimension of an embedding model significantly with a negligible performance drop. Moreover, we have shown that our method can enhance the quality of self-supervised representation by self embedding transfer.

REFERENCES

Sungsoo Ahn, Shell Xu Hu, Andreas Damianou, Neil D. Lawrence, and Zhenwen Dai. Variational information distillation for knowledge transfer. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *Proc. Neural Information Processing Systems (NeurIPS)*, 2019.

Maxime Bucher, Stéphane Herbin, and Frédéric Jurie. Improving semantic embedding consistency by metric learning for zero-shot classification. In *Proc. European Conference on Computer Vision (ECCV)*, 2016.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proc. International Conference on Machine Learning (ICML)*, 2020a.

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020b.

Yuntao Chen, Naiyan Wang, and Zhaoxiang Zhang. Darkrank: Accelerating deep metric learning via cross sample similarities transfer. In *Proc. AAAI Conference on Artificial Intelligence (AAAI)*, 2017.

S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proc. International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2011.

Michael A. A. Cox and Trevor F. Cox. *Multidimensional Scaling*, pp. 315–347. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: a large-scale hierarchical image database. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

Tommaso Furlanello, Zachary Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born again neural networks. In *Proc. International Conference on Machine Learning (ICML)*, 2018.

R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality reduction by learning an invariant mapping. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

Ben Harwood, Vijay Kumar B G, Gustavo Carneiro, Ian Reid, and Tom Drummond. Smart mining for deep metric learning. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2017.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proc. International Conference on Machine Learning (ICML)*, 2015.

Pierre Jacob, David Picard, Aymeric Histace, and Edouard Klein. Metric learning with horde: High-order regularizer for deep embeddings. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2019.

Sungyeon Kim, Minkyo Seo, Ivan Laptev, Minsu Cho, and Suha Kwak. Deep metric learning beyond binary supervision. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Sungyeon Kim, Dongwon Kim, Minsu Cho, and Suha Kwak. Proxy anchor loss for deep metric learning. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Wonsik Kim, Bhavya Goyal, Kunal Chawla, Jungmin Lee, and Keunjoo Kwon. Attention-based ensemble for deep metric learning. In *Proc. European Conference on Computer Vision (ECCV)*, 2018.

Byungsoo Ko and Geonmo Gu. Embedding expansion: Augmentation in embedding space for deep metric learning. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

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

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. 2009.

Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Proc. International Conference on Learning Representations (ICLR)*, 2019. URL `https://openreview.net/forum?id=Bkg6RiCqY7`.

Timo Milbich, Karsten Roth, Homanga Bharadhwaj, Samarth Sinha, Yoshua Bengio, Björn Ommer, and Joseph Paul Cohen. Diva: Diverse visual feature aggregation fordeep metric learning. In *Proc. European Conference on Computer Vision (ECCV)*, 2020.

Deen Dayal Mohan, Nishant Sankaran, Dennis Fedorishin, Srirangaraj Setlur, and Venu Govindaraju. Moving in the right direction: A regularization for deep metric learning. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Yair Movshovitz-Attias, Alexander Toshev, Thomas K Leung, Sergey Ioffe, and Saurabh Singh. No fuss distance metric learning using proxies. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2017.

M. Opitz, G. Waltner, H. Possegger, and H. Bischof. Bier — boosting independent embeddings robustly. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2017.

Michael Opitz, Georg Waltner, Horst Possegger, and Horst Bischof. Deep metric learning with bier: Boosting independent embeddings robustly. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2018.

Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Nikolaos Passalis and Anastasios Tefas. Learning deep representations with probabilistic knowledge transfer. In *Proc. European Conference on Computer Vision (ECCV)*, 2018.

Limeng Qiao, Yemin Shi, Jia Li, Yaowei Wang, Tiejun Huang, and Yonghong Tian. Transductive episodic-wise adaptive metric for few-shot learning. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2019.

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. In *Proc. International Conference on Learning Representations (ICLR)*, 2014.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. FaceNet: A unified embedding for face recognition and clustering. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Proc. Neural Information Processing Systems (NeurIPS)*, 2017.

Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *Proc. Neural Information Processing Systems (NeurIPS)*, 2016.

Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. In *Proc. International Conference on Learning Representations (ICLR)*, 2019.

Jiang Wang, Yang Song, T. Leung, C. Rosenberg, Jingbin Wang, J. Philbin, Bo Chen, and Ying Wu. Learning fine-grained image similarity with deep ranking. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

Xun Wang, Xintong Han, Weilin Huang, Dengke Dong, and Matthew R Scott. Multi-similarity loss with general pair weighting for deep metric learning. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Xun Wang, Haozhi Zhang, Weilin Huang, and Matthew R Scott. Cross-batch memory for embedding learning. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6388–6397, 2020.

P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology, 2010.

Chao-Yuan Wu, R. Manmatha, Alexander J. Smola, and Philipp Krahenbuhl. Sampling matters in deep embedding learning. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2017.

Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Yang You, Igor Gitman, and Boris Ginsburg. Large batch training of convolutional networks. *arXiv preprint arXiv:1708.03888*, 2017.

Baosheng Yu and Dacheng Tao. Deep metric learning with tuplet margin loss. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2019.

Lu Yu, Vacit Oguz Yazici, Xialei Liu, Joost van de Weijer, Yongmei Cheng, and Arnau Ramisa. Learning metrics from teachers: Compact networks for image embedding. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Sukmin Yun, Jongjin Park, Kimin Lee, and Jinwoo Shin. Regularizing class-wise predictions via self-knowledge distillation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *Proc. International Conference on Learning Representations (ICLR)*, 2016.

Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4l: Self-supervised semi-supervised learning. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2019.

Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *Proc. European Conference on Computer Vision (ECCV)*, 2016.

Ziming Zhang and Venkatesh Saligrama. Zero-shot learning via joint latent similarity embedding. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

A APPENDIX

This appendix presents a deeper analysis on the proposed method, its implementation details, and experimental results omitted from the main paper due to the space limit. First, Section A.1 qualitatively analyzes the roles and effects of the knowledge our method distills and transfers, and Section A.2 illustrates details of the multi-view data augmentation strategy. Finally, in Section A.3, we presents more qualitative examples for image retrieval before and after applying the proposed method on the three metric learning benchmarks.

A.1 ANALYSIS ON ROLES AND EFFECTS OF TRANSFERRED KNOWLEDGE

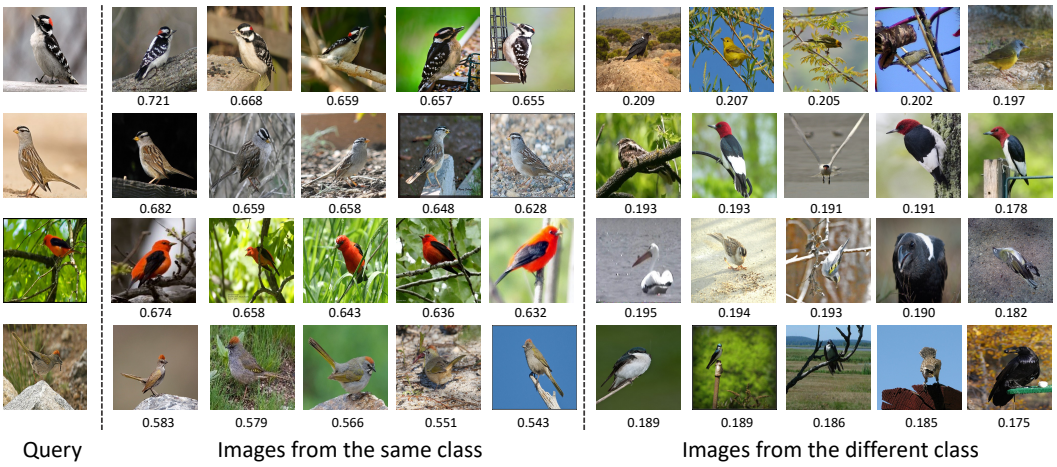

Figure 4: Image pairs sorted by the normalized weights of Eq. (4) on the CUB-200-2011 dataset. The leftmost images are paired with the other images in their rows.

The proposed method extracts the knowledge in the form of normalized weights, denoted by w_{ij}^s, that are used to control the behavior of the smooth contrastive loss. Such a knowledge delivers information beyond what the conventional approaches to metric learning and self-supervised learning can provide, i.e., the equality of class labels in the case of metric learning or that of image identities in the case of self-supervised learning. In this section, we present qualitative analysis on the knowledge to understand its roles and effects in the proposed method.

Fig. 4 presents five samples with high weights and five samples with low weights for each query; weights values are also reported below each image. As shown in the figure, images from the same class with similar poses or similar backgrounds have higher weights. These weights can help to deliver elaborate information such as intra-class variation, beyond the class labels used in conventional metric learning. Also, images with low weights from different classes are substantially different from query in terms of appearance. Unlike conventional metric learning that pushes samples of different classes equally, the knowledge given by the normalized weights allows to push samples away stronger if they have more different visual features. In summary, the knowledge extracted from source embedding allow to use more diverse and sophisticated information for learning than simple binary labels used in metric learning.

A.2 DETAILS OF MULTI-VIEW DATA AUGMENTATION

In recent approaches to self-supervised representation learning, the use of multi-view samples produced from the same image plays an important role for performance improvement. We use the multi-view augmentation in embedding transfer to transfer knowledge by considering relations between multiple views of individual samples, such as relations between different parts of an object. The overall procedure of our multi-view augmentation is as follows. We first apply the standard random augmentation technique multiple times to images of input batch. Then, all augmented multi-view images are passed through the source and target embedding networks. Note that the source and target

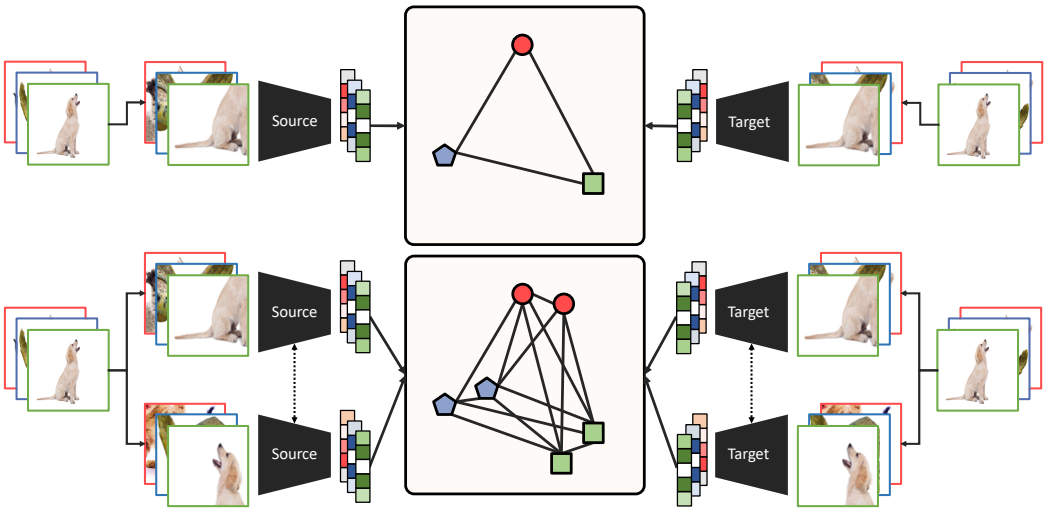

Figure 5: An illustration for standard augmentation strategy and multi-view augmentation strategy. Different colors and shapes represent distinct samples.

model take the same augmented image as input. The output embedding vectors are concatenated and used as the inputs of the embedding transfer loss. Fig. 5 illustrates this procedure where the number of views is two. The top and bottom of the figure describe the standard augmentation technique and our strategy, respectively. When using standard augmentation, only relations between different samples are considered. Applying a multi-view augmentation strategy for embedding transfer allows knowledge transfer to consider more diverse and detailed relations between samples produced from the same image.

A.3 Additional Qualitative Results for Deep Metric Learning

More qualitative results of image retrieval on the CUB-200-2011, Cars-196, and SOP datasets are presented in Fig. 6, 7, and 8, respectively. We prove the positive effect of the proposed method by showing qualitative results before and after applying the proposed method in the *self-transfer* setting; the source embedding model is Inception-BatchNorm with 512 embedding dimension and trained with the proxy-anchor loss (Kim et al., 2020). The overall results indicate that the proposed method significantly improves the source embedding model. From the examples of the 1st, 2nd, and 4th rows of Fig. 6, both models retrieve birds visually similar to the query, but only the models after embedding transfer successfully retrieved birds of the same species. Meanwhile, the examples of the 2nd and 4th rows of Fig. 7 show that the model trained with our method provides accurate results regardless of the color changes of the cars. Also, in the examples of the 1st, 2nd, and 4th rows of Fig. 8, the source model makes mistakes easily since the false positives are similar to the query in terms of appearance, yet it becomes more accurate after applying embedding transfer with the proposed method.

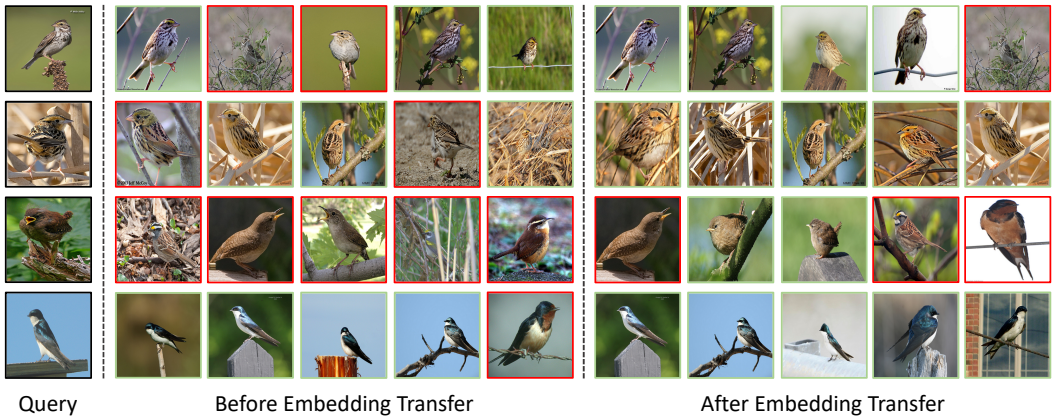

Figure 6: Top 5 image retrievals of the state of the art (Kim et al., 2020) before and after the proposed method is applied on the CUB-200-2011 dataset. Images with green boundary are success cases and those with red boundary are false positives.

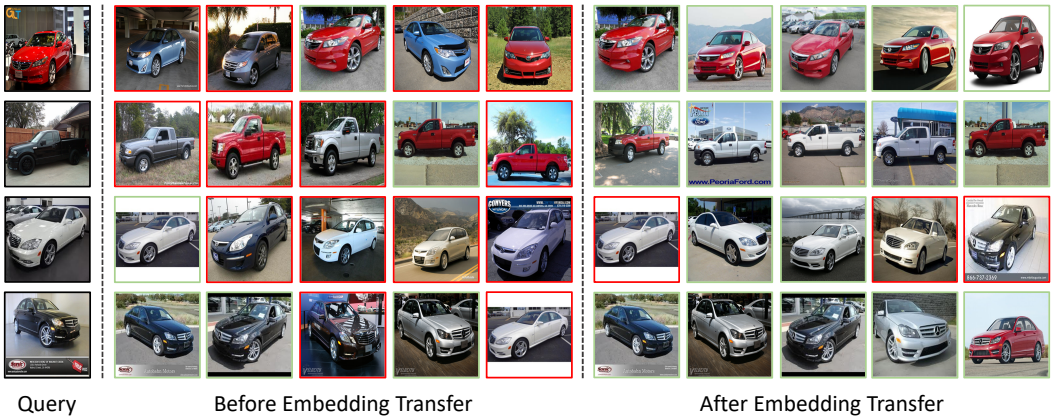

Figure 7: Top 5 image retrievals of the state of the art (Kim et al., 2020) before and after the proposed method is applied on the Cars-196 dataset. Images with green boundary are success cases and those with red boundary are false positives.

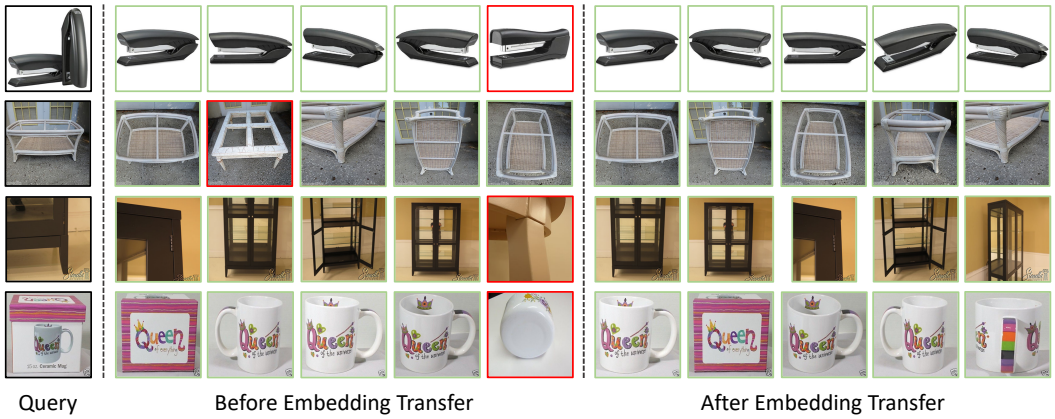

Figure 8: Top 5 image retrievals of the state of the art (Kim et al., 2020) before and after the proposed method is applied on the SOP dataset. Images with green boundary are success cases and those with red boundary are false positives.