Deep Relational Metric Learning

Wenzhao Zheng*, Borui Zhang*, Jiwen Lu†, Jie Zhou
Department of Automation, Tsinghua University, China
Beijing National Research Center for Information Science and Technology, China
{zhengwz18, zhang-br21}@mails.tsinghua.edu.cn; {lujiwen, jzhou}@tsinghua.edu.cn

Abstract

This paper presents a deep relational metric learning (DRML) framework for image clustering and retrieval. Most existing deep metric learning methods learn an embedding space with a general objective of increasing interclass distances and decreasing intraclass distances. However, the conventional losses of metric learning usually suppress intraclass variations which might be helpful to identify samples of unseen classes. To address this problem, we propose to adaptively learn an ensemble of features that characterizes an image from different aspects to model both interclass and intraclass distributions. We further employ a relational module to capture the correlations among each feature in the ensemble and construct a graph to represent an image. We then perform relational inference on the graph to integrate the ensemble and obtain a relation-aware embedding to measure the similarities. Extensive experiments on the widely-used CUB-200-2011, Cars196, and Stanford Online Products datasets demonstrate that our framework improves existing deep metric learning methods and achieves very competitive results. [1]

1. Introduction

How to effectively measure the similarities among examples has been an important problem for many computer vision tasks. Metric learning aims to learn a distance metric under which samples from one class are close to each other and far away from samples from the other classes. Taking advantage of the deep learning technique [15, 26, 45, 50], deep metric learning (DML) methods utilize the combination of a convolutional neural network (CNN) and fully connected layers (FCs) to construct a mapping from the image space to an embedding space and employ the Euclidean distance in this space to measure the similarities between samples. Deep metric learning has outperformed conventional methods by a large margin and demonstrated promising results in a variety of tasks, such as image retrieval [35, 46, 47, 67], face recognition [16, 30, 43, 53], and person re-identification [4, 5, 44, 52, 68].

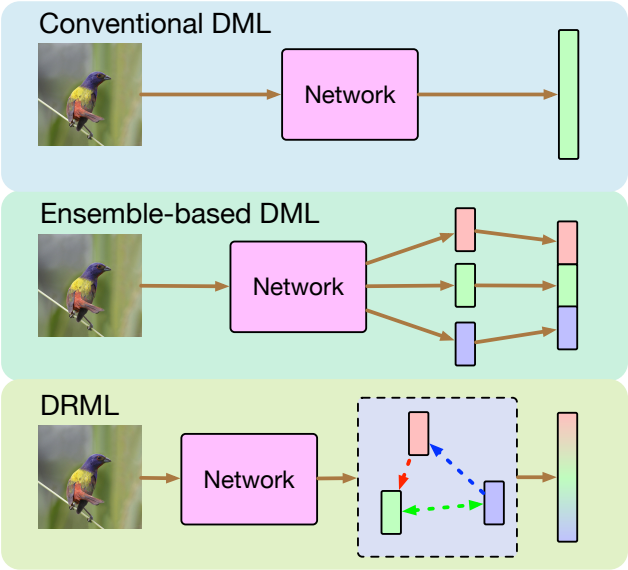


Figure 1. Comparisons of the proposed DRML framework with conventional deep metric learning methods and ensemble-based deep metric learning methods. Conventional DML employs a deep neural network to obtain an embedding for each image and measures the similarity of two samples by the Euclidean distance between their corresponding embeddings. Ensemble-based DML learns an ensemble of embeddings but simply concatenates them to obtain the final embedding, which ignores the structural relations among them. Differently, the proposed DRML framework represents one image using an ensemble of features as well as their interactions and then incorporates the relations to infer the final embedding. (Best viewed in color.)

Most existing metric learning methods utilize a discriminative objective to learn the embedding space, which encourages the metric to discard intraclass variations so that intraclass samples will form a compact cluster with a large margin from the other clusters. The large margin nature of metric learning has been proven useful to improve its robustness and generalization for *seen* class retrieval or classi-

*Equal contribution.
†Corresponding author.
[1]Code: https://github.com/zbr17/DRML.

fication [7, 53, 59]. However, the essence of metric learning is the ability to generalize to *unseen* classes in the test phase, as is usually evaluated in standard deep metric learning settings [35, 46, 47]. Directly learning a discriminative representation actually harms the performance for unseen class retrieval [41, 62], since the discarded intraclass variations may be useful to differentiate the unseen classes. Moreover, it seems in conflict with the general discriminative objective of metric learning to preserve certain characteristics that is unhelpful for distinguishing training data but might be helpful for unseen test data. A natural question is raised: *can we learn a discriminative metric that also generalizes well?*

In this paper, we provide a positive answer to this question. We propose a deep relational metric learning (DRML) framework which learns to comprehensively characterize an instance as well as distinguish different instances. We first learn a set of feature extractors to produce an ensemble of features, where each of them describes an image in one aspect. We adopt a bottleneck architecture to determine the dominant characteristics of each image and only use samples with the corresponding dominant characteristics to train each feature. The learned ensemble of features models both interclass differences and intraclass variations, and thus is not discriminative enough to be directly used to compute the distance between images. To effectively measure the similarity between two ensembles of features, we further propose a relational model to discover structural patterns in the feature ensemble and exploit them to obtain a relation-aware embedding. The proposed DRML framework can be trained effectively in an end-to-end manner and enjoys the advantage of efficient retrieval similar to existing deep metric learning methods. Differently, our framework induces a stronger relational bias than the combination of convolutional layers and fully connected layers and thus can generalize better to unseen classes. Figure 1 compares our framework with existing deep metric learning methods. We conduct extensive experiments on three widely-used datasets which demonstrate the effectiveness of the proposed framework.

2. Related Work

Deep Metric Learning: Deep metric learning methods employ deep neural networks to map an image to an embedding space so that we can effectively measure the similarities between two samples using the Euclidean distance. To achieve this, a variety of methods impose a discriminative constraint on the image embeddings [3, 10, 13, 43, 46, 47, 49, 55, 57, 64]. For example, the triplet loss [6, 43, 54] require the distance between each negative pair to be larger than that between each positive pair with a fixed margin. Song *et al.* [47] designed a lifted structured loss to consider all pairs in one batch and implicitly assign larger weights to harder samples. The large number of pairs (or tuples) makes sampling an important component in deep metric learning. A widely used technique is the hard mining strategy which proposes to select false positive tuples for effective training [12, 14, 17, 43, 65]. In addition, some methods explored other sampling schemes to boost the performance [8, 9, 12, 29, 31, 34, 39, 48, 60, 66, 67]. For example, Wu *et al.* [60] proposed to select pairs with possibilities based on their distances to achieve uniform sampling.

Recently proposed methods begin to consider learning an ensemble of embeddings and concatenate them as the final representation for distance measure [22, 32, 35, 42, 61, 65]. As pointed out by [2], the diversity of the learned embeddings in the ensemble is crucial to the final performance of the ensemble. Existing methods achieve diversity by initializing the learners diversely [35], using a diversity loss [22, 38], weighting each sample adaptively for different learners [35] or using different samples to train different learners [42, 61, 65]. However, they all use a simple concatenation to aggregate them, which fail to consider the structural relations between each entity and lead to inferior generalization performance.

Relational Inference: The recent success of deep learning in computer vision largely relies on the ability of deep convolutional neural networks to effectively represent images. However, it has been demonstrated that CNNs are not effective for non-Euclidean data and structural relation modeling [1]. As important complements to CNNs, graph neural networks (GNNs) [11, 19, 23, 24, 27, 28, 56, 63] have drawn a lot of attention and demonstrated great power for relational inference, which achieve promising results in many tasks like transfer learning [63], few-shot learning [11], video understanding [56], and scene understanding [19]. For example, Yang *et al.* [63] proposed to capture correlations among pixels and transfer a latent relational graph instead of a generic embedding vector. Wang *et al.* [56] utilized a graph to represent a video in order to model both spatial and temporal relations.

Motivated by recent advances in relational inference, we propose to represent an image using an ensemble of features and employ a relational model to discover structural patterns. We further perform relational inference to aggregate the ensemble and obtain a relation-aware embedding to measure similarities. Different from existing methods, we propose an end-to-end framework to simultaneously extract generic features and infer their relations to construct a discriminative metric with good generalization ability.

3. Proposed Approach

In this section, we first introduce our method of adaptive ensemble learning and then present the formulation of the relation-aware embedding. Lastly, we detail our framework of deep relational metric learning.

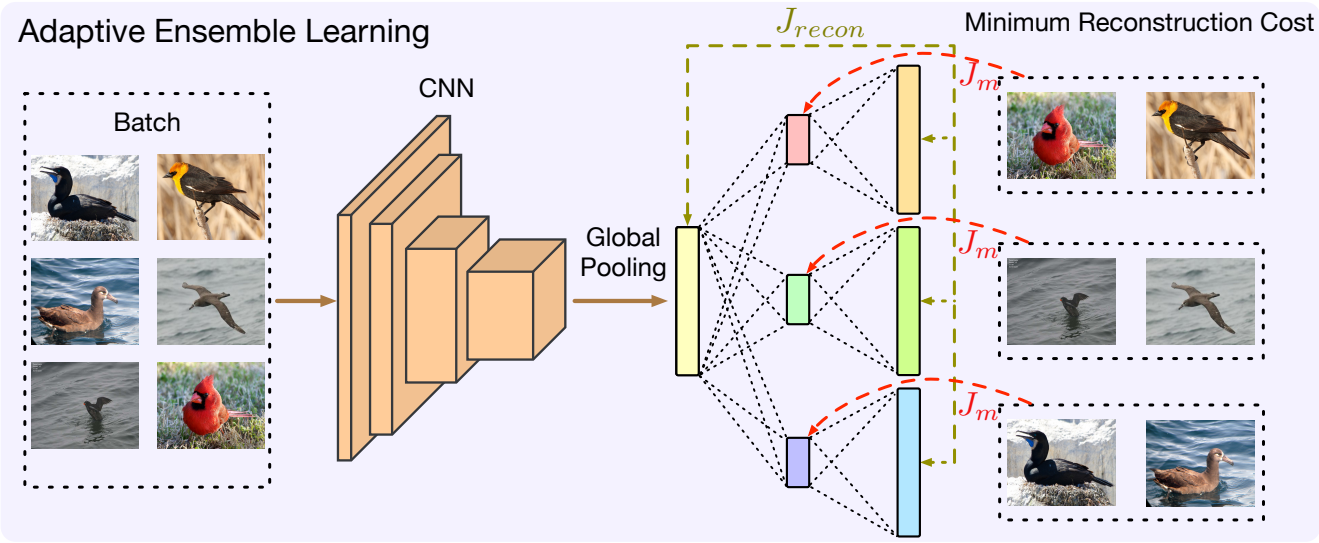


Figure 2. Illustration of the proposed adaptive ensemble learning method. We use the output after the global pooling layer of the CNN as a global feature and employ K parallel fully connected layers to obtain the ensemble of individual features. We then use K decoders to reconstruct the global feature from the corresponding individual features, where we impose a L2 reconstruction loss. We assign each image to the individual feature with the minimum reconstruction cost and only use them to train this branch with a discriminative loss J_m so that each individual feature can describe the image from one aspect. (Best viewed in color.)

3.1. Adaptive Ensemble Learning

We denote a set of images by $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$ with $L = \{l_1, l_2, \cdots, l_N\}$ as their corresponding labels, where each $l_i \in \{1, 2, \cdots, n\}$ means that $\mathbf{x}_i$ is from the l_ith class. Deep metric learning employs a deep neural network to learn a mapping $\mathbf{e}_\theta : \boldsymbol{\mathcal{X}} \xrightarrow{e_\theta} \mathbb{R}^D$ from the image space $\boldsymbol{\mathcal{X}}$ to an embedding space $\mathbb{R}^D$, and defines the learned metric as the Euclidean distance in the embedding space:

$$D(\mathbf{x}_i, \mathbf{x}_j; \theta) = ||\mathbf{e}_\theta(\mathbf{x}_i) - \mathbf{e}_\theta(\mathbf{x}_j)||_2, \tag{1}$$

where $||\cdot||_2$ denotes the L2 norm and θ is the parameters of the embedding network.

The general objective of deep metric learning is to pull closer samples from the same class and push further samples from different classes, which progressively discards intraclass variations and enlarges the distance margin between positive and negative pairs to enhance robustness. However, the discarded intraclass variations might contain information that is useful to differentiate unseen classes, and thus harm the generalization ability of the learned metric.

To address the above problem, we propose to learn an ensemble of individual features to comprehensively characterize an image from different aspects. The characteristics of an image include class-relevant ones like textures and class-irrelevant ones like backgrounds, resulting in the interclass and intraclass distributions, respectively. Instead of discarding the intraclass variations, we propose to capture both interclass and intraclass distributions by enforcing different individual features to capture different characteristics. We adaptively learn the feature ensemble and train each individual feature using only a subset of the samples.

Formally, we use a trunk CNN to extracts a global feature $\mathbf{y} = \mathbf{f}(\mathbf{x}) \in \mathbb{R}^{d'}$ (i.e., the feature after the final global pooling operation) of an image $\mathbf{x}$ and propose to learn K parallel fully connected layers $\mathbf{g}_1, \mathbf{g}_2, \cdots, \mathbf{g}_K$ to obtain an ensemble of individual features $\mathbf{G}(\mathbf{x}) = \{\mathbf{g}_1(\mathbf{f}(\mathbf{x})), \mathbf{g}_2(\mathbf{f}(\mathbf{x})), \cdots, \mathbf{g}_K(\mathbf{f}(\mathbf{x}))\}$, where each $\mathbf{g}_i(\mathbf{f}(\mathbf{x})) \in \mathbb{R}^d (d < d')$ is expected to characterize one aspect of the image. In this paper, we use the set of feature extractors $\mathbf{G}$ to represent both the trunk CNN and the subsequent parallel FCs for convenience.

To achieve this, we employ a bottleneck structure to determine the dominant characteristics of each image and only train each individual feature using a subset of training samples with the corresponding dominant characteristics. Specifically, we use a set of K fully connected layers $\mathbf{P} = \{\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_K\}$ as decoders to map each individual feature back to the d'-dimension space $\mathbb{R}^{d'}$ to reconstruct the global feature $\mathbf{y}$. We adopt the L2 loss the train the reconstruction layers $\mathbf{P}$:

$$\min_{\theta_\mathbf{P}} J_{recon}(\mathbf{y}, \mathbf{P}(\mathbf{G}(\mathbf{x}))) = \min_{\theta_\mathbf{P}} \frac{1}{K} \sum_{k=1}^{K} ||\mathbf{p}_k(\mathbf{g}_k(\mathbf{y})) - \mathbf{y}||_2, \tag{2}$$

where $||\cdot||_2$ denotes the L2 norm. Note that we only use J_{recon} to learn the parameters of the decoders which has no effect on the individual features.

We assume that different images can possess different sets of dominant characteristics that can best describe this image. For example, color is an important characteristic to distinguish different bird species, but it usually cannot be used to identify a specific car model since one car model

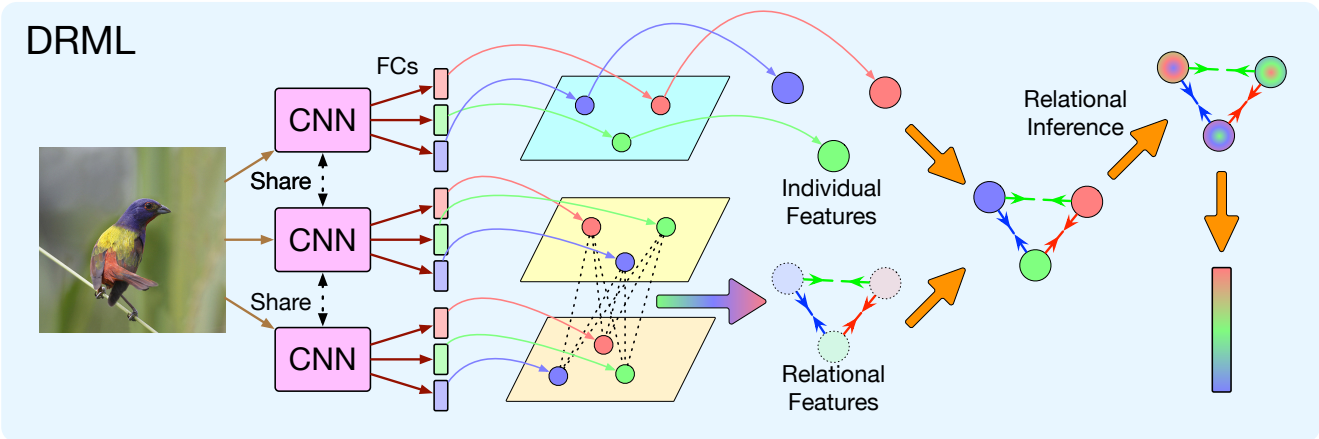


Figure 3. The flowchart of the proposed DRML framework. We employ a shared CNN network and $3K$ different fully connected layers to extract K individual features and $2K$ meta-relational features. We use the meta-relational features to produce the relational features between each pair in the ensemble of individual features. We then construct a graph to represent the input image by using the individual features and relational features as the node features and edge features, respectively. We further perform relational inference on the graph to update each individual feature and concatenate them to obtain a relation-aware embedding to measure similarities. (Best viewed in color.)

can present different colors. We further argue that if one individual feature mainly represents the dominant characteristics of the image, it should contain more information and we should be able to better reconstruct the global feature. Therefore, we first classify each image to the individual feature with the minimum reconstruction cost:

$$k^*(\mathbf{x}) = \arg\min_k ||\mathbf{p}_k(\mathbf{g}_k(\mathbf{y})) - \mathbf{y}||_2. \tag{3}$$

We then use the corresponding classified subset of images to train each individual feature:

$$\min_{\theta_{\mathbf{G}_k}} J_{ind}^k(\{\mathbf{x}\}) = \min_{\theta_{\mathbf{G}_k}} \sum_{\{\mathbf{x}|k^*(\mathbf{x})=k\}} J_m(\{\mathbf{x}\}), \tag{4}$$

where $\theta_{\mathbf{G}_k}$ represents the parameters of the trunk CNN $\mathbf{f}$ and the kth fully connected layer $\mathbf{g}_k$, and J_m can be any discriminative loss of metric learning like the triplet loss [43] or Proxy-NCA loss [34], and $\{\mathbf{x}\}$ denotes a tuple depending on the specific form of J_m.

The K decoders (i.e., $\mathbf{P}$) are trained with J_{recon} using all the samples, while each encoder (i.e., $\mathbf{g}$) is trained with J_{ind} using the subset of samples with the minimal reconstruction cost. The set of decoders serves as a classifier to assign each sample to the feature extractor that best preserves its characteristics. Still, we allow the gradients of J_{ind} through the backbone CNN so that it is trained using all the samples.

Intuitively, we learn the feature ensemble to comprehensively represent an image as well as reinforce the characteristics that can well recognize samples. We only use a subset of training samples to learn each individual feature, which essentially imposes a less discriminative constraint. Still, J_{ind} restricts the feature ensembles of a positive pair to have at least one feature that can well determine the resemblance between them and strengthens only the dominant characteristics to differentiate the negative pair to avoid pushing away features on which they actually resemble each other.

3.2. Relation-Aware Embedding Learning

While the learned feature ensemble can comprehensively represent an image, how to exploit them to measure the similarities between images is not trivial. Simply concatenating the features in the ensemble and employing the Euclidean distance to measure the similarities is not effective since we do not explicitly constrain the distance between the concatenation of individual features. The difference between a positive pair can be large and the difference between a negative pair can be small. However, there exist certain relational patterns other than simple distances in the distributions of the individual features that can distinguish them, since we describe each image from different aspects and preserve the dominant characteristics that best characterize it.

We hence propose a relational model $\mathbf{h}(\mathbf{G}(\mathbf{x}))$ that can well capture the relations among the individual features and emploit them to aggregate the features to obtain the final embedding. Ideally, we want to model each of the relations between two features using separate functions, i.e. $\mathbf{R} = \{\mathbf{R}_{ij}, i, j \in 1, 2, \cdots, K\}$, but the total number of functions to be learned is with the complexity of $O(K^2)$, which is prone to overfitting. Instead, we propose to learn two set of meta-relational features $\mathbf{A} = \{\mathbf{a}_1, \mathbf{a}_2, \cdots, \mathbf{a}_K\}$ and $\mathbf{B} = \{\mathbf{b}_1, \mathbf{b}_2, \cdots, \mathbf{b}_K\}$ to represent the relations among features. The relation feature between the ith and jth individual feature of an image $\mathbf{x}$ is formulated by the difference between the corresponding meta-relational features:

$$\mathbf{R}_{ij}(\mathbf{x}) = \mathbf{a}_i(\mathbf{f}(\mathbf{x})) - \mathbf{b}_j(\mathbf{f}(\mathbf{x})), \tag{5}$$

where we only need $2K$ functions to represent all the rela-

tions. We can regard $\mathbf{A}$ and $\mathbf{B}$ as representations of the individual feature ensemble in a relational space, where the difference between two features indicates the relations. Note that the relations are not symmetric, i.e., $\mathbf{R}_{ij} \neq \mathbf{R}_{ji}$. This enables the proposed relational model to encode more complex and structural relations.

Having obtained the relations, we construct a graph with the individual features as node features and the relational features as edge features. The graph representation of an image encodes more information than the conventional embedding representation, as it not only expresses the characteristics of the image but also the relations among them.

We can then employ any Graph Neural Networks to perform the relational inference and obtain a final global graph embedding for each image. In this work, we design a simple but effective module to aggregate the individual features based on the learned relations. We first produce the messages sent to each individual feature by a weighted sum of all features:

$$\mathbf{M}_i = \sum_{j=1}^{K} r_{ji} \mathbf{g}_j(\mathbf{f}(\mathbf{x})). \tag{6}$$

r_{ji} is the weight of the jth individual feature computed by the normalized relational score:

$$r_{ji} = \frac{s(\mathbf{R}_{ji}(\mathbf{x}))}{\sum_{j=1}^{K} s(\mathbf{R}_{ji}(\mathbf{x}))}, \tag{7}$$

where $s(\cdot)$ is a relational score function instantiated by a fully connected layer to compute a score reflecting the "closeness" between two individual features.

We then integrate the incoming message into the individual feature and update it as follows:

$$\mathbf{g}_i^u(\mathbf{x}) = \mathbf{U}(concat([\mathbf{g}_i(\mathbf{f}(\mathbf{x})); \mathbf{M}_i])), \tag{8}$$

where $concat(\cdot)$ denotes the concatenation operation and $\mathbf{U}$ is an updater instantiated by a fully connected layer.

We concatenate all the updated individual features to obtain the final relation-aware embedding of the image $\mathbf{z} = concat([\mathbf{g}_1^u(\mathbf{x}); \mathbf{g}_2^u(\mathbf{x}); \cdots; \mathbf{g}_K^u(\mathbf{x})])$ and measure the similarities between two images by the Euclidean distance of the relation-aware embeddings:

$$D(\mathbf{x}_i, \mathbf{x}_j) = ||\mathbf{h}(\mathbf{G}(\mathbf{x}_i)) - \mathbf{h}(\mathbf{G}(\mathbf{x}_j))||_2 = ||\mathbf{z}_i - \mathbf{z}_j||_2. \tag{9}$$

We then apply an embedding loss J_{emb} to the embeddings similar to existing deep metric learning methods.

Though eventually our model still uses a single embedding to represent the similarity between images, the underlying model to obtain the embedding is essentially different from conventional deep metric learning methods. The proposed DRML framework first extracts an ensemble of features to comprehensively characterize an image and then employs a relational model to exploit their relations to produce the final embedding. It exploits more structural information and induces a stronger relational bias than the simple combination of CNNs and FCs.

3.3. Deep Relational Metric Learning

We present our proposed DRML framework, which is composed of the trunk CNN to extract the comprehensive feature, a set of FCs $\{\mathbf{g}_1, \mathbf{g}_2, \cdots, \mathbf{g}_K\}$ to produce an ensemble of individual features, and a relational model $\mathbf{h}$ to obtain the relation-aware embedding to compute the distance, as shown in Figure 3.

The feature extractors $\mathbf{G}$ and the relational model $\mathbf{h}$ can be trained simultaneously, and we block the effect of J_{emb} on $\mathbf{G}$. This enables the set of feature extractors $\mathbf{G}$ to focus on learning a more comprehensive representation of the images, while the relational model $\mathbf{h}$ aims to mine the relations among the individual features and discover patterns to identify the images. The two modules work together towards the same direction only with slightly different purposes.

We use the ensemble loss J_{ensem} to train the set of feature extractors $\mathbf{G}$, the reconstruction loss J_{recon} to learn the decoders $\mathbf{P}$, and the embedding loss J_{emb} to train the relational model $\mathbf{h}$:

$$\min_{\theta_{\mathbf{G}}, \theta_{\mathbf{P}}, \theta_{\mathbf{h}}} J = \min_{\theta_{\mathbf{G}}} J_{ensem} + \lambda_1 \min_{\theta_{\mathbf{P}}} J_{recon} + \lambda_2 \min_{\theta_{\mathbf{h}}} J_{emb}, \tag{10}$$

where $J_{ensem} = \sum_{k=1}^{K} J_{ind}^k$ represents the ensemble objective, $\theta_{\mathbf{G}}$, $\theta_{\mathbf{P}}$ and $\theta_{\mathbf{h}}$ are the parameters of $\mathbf{G}$, $\mathbf{P}$, and $\mathbf{h}$, respectively, and λ_1 and λ_2 are two pre-defined parameters to balance the contributions of the three losses. Though $\mathbf{G}$ has an effect on the final embedding $\mathbf{z}$ and thus J_{emb}, we do not backpropagate the gradients of J_{emb} through $\mathbf{G}$.

Our framework can be generally applied to existing deep metric learning methods with various loss functions and sampling strategies. In this paper, we adopt the same discriminative loss as the individual feature loss J_m and the embedding loss J_{emb}, but note that our DRML framework does not stipulate this identity. We can use different sampling strategies to sample tuples based on the employed loss and use them to compute the embedding loss J_{emb}.

The proposed DRML framework can be trained in an end-to-end manner. We use an ensemble of individual features to describe an image and employ a relational model with stronger relational inductive biases than existing deep metric learning methods to improve the generalization ability. Still, our framework preserves the advantage of efficient retrieval, where we can pre-compute the embeddings of all the images in the gallery and simply compare the embedding of a query image with them.

4. Experiments

In this section, we evaluated our framework on three widely used datasets on image clustering and retrieval tasks.

4.1. Datasets

To evaluate the generalization performance of our framework on unseen classes, we followed existing methods [46, 47, 67] to conduct experiments under a zero-shot setting, where the training subset has no intersection with the test subset. The three datasets were split as follows.

The **CUB-200-2011** dataset [51] includes 200 bird species of 11,788 images. We used the first 100 species of 5,864 images as the training subset and the remaining 100 species of 5,924 images as the test subset. The **Cars196** dataset [25] contains 196 car makes and models of 16,185 images. We used the first 98 models of 8,054 images as the training subset and the remaining 100 models of 8,131 images as the test subset. The **Stanford Online Products** dataset [47] includes 22,634 products of 120,053 images collected from eBay.com. We used the first 11,318 products of 59,551 images as the training subset and the remaining 11,316 products of 60,502 images as the test subset.

4.2. Implementation Details

We used the PyTorch package in all the following experiments. We used the ImageNet [40] pretrained CNN as the backbone model in our framework for a fair comparison with most existing deep metric learning methods. We set the number of individual features K to 4. We added twelve randomly initialized fully connected layers after the global pooling layer, where four of them output the individual features in the ensemble and eight of them output the meta-relational features. We set the dimensions of individual features and meta-relation features to 128. We set the output dimension of the updater to 128 which is the same with that of the individual features. We first normalized all the images to 256×256. For training, we performed data augmentation by standard random cropping to 227×227 and random horizontal mirror. We fixed the batch size to 80 and set the learning rate to 10^{-5} for the CNN and 10^{-4} for the other fully connected layers. We set λ_1 and λ_2 to 0.1 and 10 to balance the effects of the losses.

4.3. Evaluation Metrics

We conducted experiments on both image clustering and retrieval tasks following existing works [46, 47, 67]. For the image clustering task, we adopted the normalized mutual information (NMI) and F_1 score as the evaluation metrics. NMI computes the mutual information between clusters and the ground truth classes normalized by the arithmetic mean of the entropies of them, i.e., $\text{NMI}(\Omega, \mathbb{C}) = \frac{2I(\Omega;\mathbb{C})}{H(\Omega)+H(\mathbb{C})}$, where $\Omega = \{\omega_1, \cdots, \omega_K\}$ denotes the set of clusters and $\mathbb{C} = \{c_1, \cdots, c_K\}$ denotes the set of ground truth classes. ω_i includes samples predicted to belong to the ith cluster, and c_j includes samples from the jth class. The F_1 score computes the harmonic mean of the precision and recall , i.e., $\text{F}_1 = \frac{2PR}{P+R}$. For the image retrieval task, we adopted the Recall@Ks as the evaluation metrics. Recall@K measures the percentage of legitimate samples in the test subset. A sample is deemed legitimate if there exists at least one positive sample in its K nearest neighbors. We direct interesting readers to [47] for detailed explanation.

Table 1. Effect of different numbers of individual features.

No. of features	R@1	R@2	R@4	R@8	NMI
2	68.0	77.4	85.7	90.9	68.6
4	**68.7**	**78.6**	**86.3**	**91.6**	**69.3**
8	66.1	76.0	84.8	90.9	68.0
16	65.7	76.2	84.2	90.1	66.8

Table 2. Ablation study of different model settings.

Methods	R@1	R@2	R@4	R@8	NMI
DiVA [32]	66.4	77.2	85.8	91.5	**69.6**
DRML-DiVA	63.8	74.8	84.1	90.2	67.5
AEL-PA	60.5	71.8	81.1	88.1	62.6
DRML-PA	**68.7**	**78.6**	**86.3**	**91.6**	69.3

Table 3. Ablation study using different loss functions.

Method	R@1	R@2	R@4	R@8	NMI
DRML w/o J_{ensem}	45.6	59.5	72.2	82.5	56.3
DRML w/o J_{emb}	60.4	71.4	81.1	88.4	61.3
DRML w/o J_{recon}	66.6	77.3	85.3	90.7	69.3
DRML-PA	**68.8**	**79.3**	**87.1**	**91.6**	**71.6**

4.4. Results and Analysis

Number of individual features: The proposed DRML framework employs K individual features to describe an image from different aspects. We conducted experiments on the CUB-200-2011 dataset to analyze the effect of the number of individual features, as shown in Table 1. We apply the proposed DRML to the ProxyAnchor loss [21] (i.e., DRML-PA) and fix the embedding size to 512. We use 2, 4, 8, and 16 as the number of individual features, rendering the size of each individual feature to 256, 128, 64, and 32, respectively. We see that our method achieves the best result at $K = 4$, and using a larger K slightly harms the performance. This is because, with a larger K, each individual feature possesses a lower information capacity which cannot fully describe the main characteristics of an image.

Ablation study of each module: We conducted an ablation study to analyze the effectiveness of each module in the proposed DRML framework. We first remove the relational module and only employ the proposed adaptive ensemble learning method to learn a set of individual features using the ProxyAnchor loss and simply concatenate them as the final embedding (i.e., AEL-PA). We also adopt the state-of-the-art ensemble learning method DiVA [32] to learn the individual features and employ the proposed relational mod-

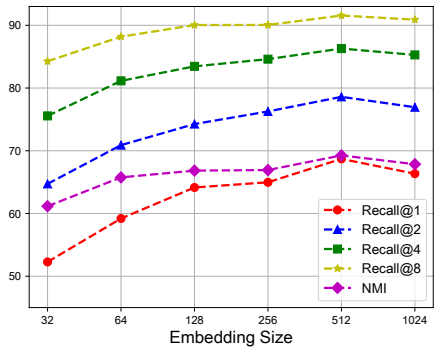
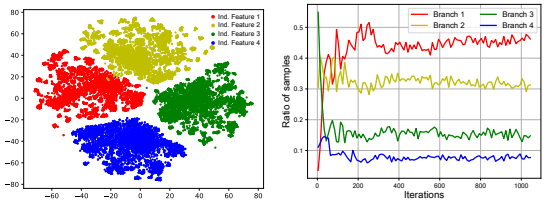


Figure 4. Analysis of embedding size.



(a) Visualization. (b) Ratio of samples.

Figure 5. Diversity analysis of the proposed ensemble learning.

ule to model the relations to infer the final embedding (i.e., DRML-DiVA). Table 2 shows the experimental results on the CUB-200-2011 dataset using BN-Inception [18] as the trunk CNN. We observe that adding the relational module hinders the performance of DiVA. We suspect this is because the ensemble learned by DiVA cannot describe an image from different aspects and our relational module cannot capture their relations. On the contrary, though the concatenation of the proposed adaptive ensemble is not discriminative enough to distinguish different samples, further employing a relational module improves the performance dramatically. This demonstrates the effectiveness of exploiting relations to infer the final embedding.

Ablation study of each loss function: We also conducted an ablation study of each loss function on the CUB-200-2011 dataset as shown in Table 3. We see that combining the three proposed loss functions achieves the best result, which demonstrates the effectiveness of each loss.

Effectiveness of the learned relations: We learn the relations in an end-to-end manner to produce valid embeddings. To directly verify the effectiveness of the relations, we conducted an experiment to use random relations and fix them while training the other modules with the ProxyAnchor loss on the CUB-200-2011 dataset. We discover that the R@1 performance decreases from 68.7 to 62.6, which shows that using the learned relations is helpful.

Effect of embedding dimension: To study the effect of using different embedding dimensions, we conducted a series of experiments on the CUB-200-2011 datasets using an embedding dimension of 32, 64, 128, 256, 512, and 1024. Figure 4 shows the performance of the proposed DRML-PA using different embedding dimensions with a fixed $K = 4$.

Table 4. Experimental results (%) of DRML in comparison with baseline methods on the CUB-200-2011 dataset.

Methods	Setting	R@1	R@2	R@4	R@8	NMI
N-Pair [46]	512G	50.1	63.3	74.3	83.2	60.4
Angular [55]	512G	53.6	65.0	75.3	83.7	61.0
HDML [67]	512G	53.7	65.7	76.7	85.7	62.6
HTL [12]	512BN	57.1	68.8	78.7	86.5	-
RLL-H [58]	512BN	57.4	69.7	79.2	86.9	63.6
HTG [66]	512R	59.5	71.8	81.3	88.2	-
Margin [60]	128R	63.6	74.4	83.1	90.0	69.0
SoftTriple [37]	512BN	65.4	76.4	84.5	90.4	69.3
Multi-Sim [57]	512BN	65.7	77.0	86.3	91.2	-
MIC [38]	128R	66.1	76.8	85.6	-	69.7
DR [33]	512BN	66.1	77.0	85.1	91.1	-
CircleLoss [49]	512R	66.7	77.4	86.2	91.2	-
RankMI [20]	128R	66.7	77.2	85.1	91.0	71.3
PADS [39]	128BN	67.3	78.0	85.9	-	69.9
Ensemble-based methods:						
HDC [65]	384G	53.6	65.7	77.0	85.6	-
A-BIER [36]	512G	57.5	68.7	78.3	86.2	-
ABE-8 [22]	512G	60.6	71.5	79.8	87.4	-
Ranked [58]	1536BN	61.3	72.7	82.7	89.4	66.1
DREML [61]	9216R	63.9	75.0	83.1	89.7	67.8
D & C [42]	128R	65.9	76.6	84.4	90.6	69.6
Triplet-SH* [43]	512R	58.8	70.6	80.6	88.3	64.6
DRML-TSH	512R	**61.7**	**73.5**	**82.4**	**88.7**	**65.6**
Margin-DW* [60]	512R	62.3	73.5	83.2	89.9	67.2
DRML-MDW	512R	**65.7**	**76.9**	**85.6**	**91.1**	**69.0**
ProxyAnchor* [21]	512BN	67.3	77.7	85.7	91.4	68.7
DRML-PA	512BN	**68.7**	**78.6**	**86.3**	**91.6**	**69.3**

We see that the performance improves as the embedding size increases and achieves the best result at 512. However, using a larger embedding size (1024) instead hinders the performance, suggesting a redundancy in the embedding space. Note that our method still maintains a fairly good performance even with a small embedding size like 32, where each individual feature only has a size of 8. We think this is because we further exploit the relations among individual features to infer the final embedding, which can partially compensate for the lack of information capacity.

Diversity of the ensemble: Since the diversity of the ensemble is important to the performance, we visualized the individual features for all the samples on the CUB-200-2011 dataset using t-SNE in Figure 5(a). We see that the learned individual features are very diverse in the feature space. This is because we encourage each feature extractor to focus on one set of characteristics by only using the corresponding subset of samples to train it. To further study the ensemble training process, we further tracked the ratio of samples assigned to each branch as shown in Figure 5(b). We observe that all the four feature extractors are properly trained only with different samples, which results in the diversity of the individual features.

Quantitative results: Since the proposed DRML framework can be generally applied to existing deep metric learning methods, we instantiated our framework with various network architectures, loss functions, and sampling strate-

Table 5. Experimental results (%) of DRML in comparison with baseline methods on the Cars196 dataset.

Methods	Setting	R@1	R@2	R@4	R@8	NMI
N-Pair [46]	512G	71.1	79.7	86.5	91.6	64.0
Angular [55]	512G	71.3	80.7	87.0	91.8	62.4
RLL-H [58]	512BN	74.0	83.6	90.1	94.1	65.4
HTG [66]	512R	76.5	84.7	90.4	94.0	-
HDML [67]	512G	79.1	87.1	92.1	95.5	69.7
Margin [60]	128R	79.6	86.5	91.9	95.1	69.1
HTL [12]	512BN	81.4	88.0	92.7	95.7	-
MIC [38]	128R	82.6	89.1	93.2	-	68.4
RankMI [20]	128R	83.3	89.8	93.8	96.5	69.4
CircleLoss [49]	512R	83.4	89.8	94.1	96.5	-
PADS [39]	128BN	83.5	89.7	93.8	-	68.8
Multi-Sim [57]	512BN	84.1	90.4	94.0	96.5	-
SoftTriple [37]	512BN	84.5	90.7	94.5	96.9	70.1
DR [33]	512BN	85.0	90.5	94.1	96.4	-
Ensemble-based methods:						
HDC [65]	384G	73.7	83.2	89.5	93.8	-
A-BIER [36]	512G	82.0	89.0	93.2	96.1	-
Ranked [58]	1536BN	82.1	89.3	93.7	96.7	71.8
D & C [42]	128R	84.6	90.7	94.1	96.5	70.3
ABE-8 [22]	512G	85.2	90.5	94.0	96.1	-
DREML [61]	9216R	86.0	91.7	95.0	97.2	76.4
Triplet-SH* [43]	512R	62.7	74.9	83.8	90.6	59.9
DRML-TSH	512R	**64.4**	**76.4**	**85.2**	**91.5**	**61.0**
Margin-DW* [60]	512R	72.3	82.3	89.3	94.2	64.6
DRML-MDW	512R	**73.3**	**83.0**	**89.8**	**94.4**	**65.3**
ProxyAnchor* [21]	512BN	84.4	90.7	94.3	96.8	69.7
DRML-PA	512BN	**86.9**	**92.1**	**95.2**	**97.4**	**72.1**

gies to evaluate the effectiveness, which includes triplet loss with random sampling [6, 54]. We applied our framework to the triplet loss with semi-hard sampling (i.e., TSH) [43], the margin loss with distance-weighted sampling (MDW) [60], and the ProxyAnchor loss with random sampling (PA) [21]. We use n-G/BN/R to indicate the model setting where n is the embedding size and G, BN, R to denote GoogleNet [50], BN-Inception [18], and ResNet-50 [15], respectively.

Tables 4, 5, and 6 show the experimental results on the CUB-200-2011, Cars196, and Stanford Online Products datasets, respectively, where * denotes our reproduced results under the same setting. We use red numbers to indicate the best results and bold numbers to highlight the improvement of our framework over the associated method without DRML. We observe that the proposed framework improves the performance of existing methods on all three datasets. While the use of an advanced sampling scheme can benefit the original loss, the proposed DRML can further boost the performance. In particular, our framework performs the best combined with the ProxyAnchor loss and achieves very competitive results. Note that the ProxyAnchor loss in the original paper [21] uses a more sophisticated pooling operation (average + max pooling) and a more advanced optimizer (AdamW) than the conventional metric learning setting, so we only report our reproduced results. Compared with the original methods, our framework represents the characteristics of an image more comprehensively

Table 6. Experimental results (%) of DRML in comparison with baseline methods on the Stanford Online Products dataset.

Methods	Setting	R@1	R@10	R@100	NMI
N-Pair [46]	512G	67.7	83.8	93.0	88.1
Angular [55]	512G	67.9	83.2	92.2	87.8
HDML [67]	512G	68.7	83.2	92.4	89.3
Margin [60]	128R	72.7	86.2	93.8	90.7
RankMI [20]	128R	74.3	87.9	94.9	90.5
HTL [12]	512BN	74.8	88.3	94.8	-
RLL-H [58]	512BN	76.1	89.1	95.4	89.7
FastAP [3]	512R	76.4	89.1	95.4	-
PADS [39]	128BN	76.5	89.0	95.4	89.9
MIC [38]	128R	77.2	89.4	95.6	90.0
Multi-Sim [57]	512BN	78.2	90.5	96.0	-
SoftTriple [37]	512BN	78.3	90.3	95.9	92.0
CircleLoss [49]	512R	78.3	90.5	96.1	-
Ensemble-based methods:					
HDC [65]	384G	70.1	84.9	93.2	-
A-BIER [36]	512G	74.2	86.9	94.0	-
D & C [42]	128R	75.9	88.4	94.9	90.2
ABE-8 [22]	512G	76.3	88.4	94.8	-
Ranked [58]	1536BN	79.8	91.3	96.3	90.4
Triplet-SH* [43]	512R	76.1	87.7	94.3	89.3
DRML-TSH	512R	**75.7**	**88.6**	**95.4**	**89.8**
ProxyAnchor* [21]	512BN	70.1	84.3	92.6	87.2
DRML-PA	512BN	**71.5**	**85.2**	**93.0**	**88.1**
Margin-DW* [60]	512R	76.8	88.9	95.1	89.7
DRML-MDW	512R	**79.9**	**90.7**	**96.1**	**90.1**

by learning a feature ensemble with a large diversity and further employs a relational model to capture the structural relations. Still, our DRML preserves the advantage of metric learning to learn a discriminative embedding space to effectively measure the similarities among samples.

5. Conclusion

In this paper, we have presented a deep relational metric learning (DRML) framework for image clustering and retrieval. We represent an image by an ensemble of features to capture both interclass and intraclass distributions and employ a relational model to characterize the relations among features. We further perform relational inference to produce a relation-aware embedding to measure similarities. We have conducted experiments on three datasets which have demonstrated that our method can boost the performance of existing methods on deep metric learning in both image clustering and retrieval tasks. It is an interesting future direction to extend our framework to process other formats of input such as video, 3D point clouds, and event-based data.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China under Grant U1813218, Grant U1713214, and Grant 61822603, in part by a grant from the Beijing Academy of Artificial Intelligence (BAAI), and in part by a grant from the Institute for Guo Qiang, Tsinghua University.

References

[1] Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv*, abs/1806.01261, 2018. 2

[2] Leo Breiman. Random forests. *ML*, 45(1):5–32, 2001. 2

[3] Fatih Cakir, Kun He, Xide Xia, Brian Kulis, and Stan Sclaroff. Deep metric learning to rank. In *CVPR*, pages 1861–1870, 2019. 2, 8

[4] Guangyi Chen, Tianren Zhang, Jiwen Lu, and Jie Zhou. Deep meta metric learning. In *ICCV*, 2019. 1

[5] Weihua Chen, Xiaotang Chen, Jianguo Zhang, and Kaiqi Huang. Beyond triplet loss: a deep quadruplet network for person re-identification. In *CVPR*, pages 1320–329, 2017. 1

[6] De Cheng, Yihong Gong, Sanping Zhou, Jinjun Wang, and Nanning Zheng. Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In *CVPR*, pages 1335–1344, 2016. 2, 8

[7] Huyen Do, Alexandros Kalousis, Jun Wang, and Adam Woznica. A metric learning perspective of svm: on the relation of lmnn and svm. In *AISTATS*, pages 308–317, 2012. 2

[8] Yueqi Duan, Lei Chen, Jiwen Lu, and Jie Zhou. Deep embedding learning with discriminative sampling policy. In *CVPR*, pages 4964–4973, 2019. 2

[9] Yueqi Duan, Wenzhao Zheng, Xudong Lin, Jiwen Lu, and Jie Zhou. Deep adversarial metric learning. In *CVPR*, pages 2780–2789, 2018. 2

[10] Ismail Elezi, Sebastiano Vascon, Alessandro Torcinovich, Marcello Pelillo, and Laura Leal-Taixe. The group loss for deep metric learning. In *ECCV*, 2020. 2

[11] Victor Garcia and Joan Bruna. Few-shot learning with graph neural networks. In *ICLR*, 2018. 2

[12] Weifeng Ge, Weilin Huang, Dengke Dong, and Matthew R Scott. Deep metric learning with hierarchical triplet loss. In *ECCV*, pages 269–285, 2018. 2, 7, 8

[13] Soumyadeep Ghosh, Richa Singh, and Mayank Vatsa. On learning density aware embeddings. In *CVPR*, pages 4884–4892, 2019. 2

[14] Ben Harwood, Vijay Kumar B G, Gustavo Carneiro, Ian Reid, and Tom Drummond. Smart mining for deep metric learning. In *ICCV*, pages 2840–2848, 2017. 2

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1, 8

[16] Junlin Hu, Jiwen Lu, and Yap-Peng Tan. Discriminative deep metric learning for face verification in the wild. In *CVPR*, pages 1875–1882, 2014. 1

[17] Chen Huang, Chen Change Loy, and Xiaoou Tang. Local similarity-aware deep feature embedding. In *NIPS*, pages 1262–1270, 2016. 2

[18] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, pages 448–456, 2015. 7, 8

[19] Seong Jae Hwang, Sathya N Ravi, Zirui Tao, Hyunwoo J Kim, Maxwell D Collins, and Vikas Singh. Tensorize, factorize and regularize: Robust visual relationship learning. In *CVPR*, pages 1014–1023, 2018. 2

[20] Mete Kemertas, Leila Pishdad, Konstantinos G Derpanis, and Afsaneh Fazly. Rankmi: A mutual information maximizing ranking loss. In *CVPR*, pages 14362–14371, 2020. 7, 8

[21] Sungyeon Kim, Dongwon Kim, Minsu Cho, and Suha Kwak. Proxy anchor loss for deep metric learning. In *CVPR*, pages 3238–3247, 2020. 6, 7, 8

[22] Wonsik Kim, Bhavya Goyal, Kunal Chawla, Jungmin Lee, and Keunjoo Kwon. Attention-based ensemble for deep metric learning. In *ECCV*, pages 760–777, 2018. 2, 7, 8

[23] Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. Neural relational inference for interacting systems. In *ICML*, pages 2693–2702, 2018. 2

[24] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2016. 2

[25] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *ICCVW*, pages 554–561, 2013. 6

[26] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012. 1

[27] Guohao Li, Matthias Muller, Ali Thabet, and Bernard Ghanem. Deepgcns: Can gcns go as deep as cnns? In *CVPR*, pages 9267–9276, 2019. 2

[28] Yujia Li, Chenjie Gu, Thomas Dullien, Oriol Vinyals, and Pushmeet Kohli. Graph matching networks for learning the similarity of graph structured objects. In *ICML*, pages 3835–3845, 2019. 2

[29] Xudong Lin, Yueqi Duan, Qiyuan Dong, Jiwen Lu, and Jie Zhou. Deep variational metric learning. In *ECCV*, pages 689–704, 2018. 2

[30] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *CVPR*, pages 6738–6746, 2017. 1

[31] Jing Lu, Chaofan Xu, Wei Zhang, Ling-Yu Duan, and Tao Mei. Sampling wisely: Deep image embedding by top-k precision optimization. In *ICCV*, pages 7961–7970, 2019. 2

[32] 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 *ECCV*, 2020. 2, 6

[33] Deen Dayal Mohan, Nishant Sankaran, Dennis Fedorishin, Srirangaraj Setlur, and Venu Govindaraju. Moving in the right direction: A regularization for deep metric learning. In *CVPR*, pages 14591–14599, 2020. 7, 8

[34] Yair Movshovitz-Attias, Alexander Toshev, Thomas K. Leung, Sergey Ioffe, and Saurabh Singh. No fuss distance metric learning using proxies. In *ICCV*, pages 360–368, 2017. 2, 4

[35] Michael Opitz, Georg Waltner, Horst Possegger, and Horst Bischof. Bier - boosting independent embeddings robustly. In *ICCV*, pages 5189–5198, 2017. 1, 2

[36] Michael Opitz, Georg Waltner, Horst Possegger, and Horst Bischof. Deep metric learning with bier: Boosting independent embeddings robustly. *TPAMI*, 2018. 7, 8

[37] Qi Qian, Lei Shang, Baigui Sun, and Juhua Hu. Softtriple loss: Deep metric learning without triplet sampling. In *ICCV*, 2019. 7, 8

[38] Karsten Roth, Biagio Brattoli, and Bjorn Ommer. Mic: Min-

ing interclass characteristics for improved metric learning. In *ICCV*, pages 8000–8009, 2019. 2, 7, 8

[39] Karsten Roth, Timo Milbich, and Bjorn Ommer. Pads: Policy-adapted sampling for visual similarity learning. In *CVPR*, pages 6568–6577, 2020. 2, 7, 8

[40] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015. 6

[41] Alexandre Sablayrolles, Matthijs Douze, Nicolas Usunier, and Hervé Jégou. How should we evaluate supervised hashing? In *ICASSP*, pages 1732–1736, 2017. 2

[42] Artsiom Sanakoyeu, Vadim Tschernezki, Uta Buchler, and Bjorn Ommer. Divide and conquer the embedding space for metric learning. In *CVPR*, pages 471–480, 2019. 2, 7, 8

[43] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, pages 815–823, 2015. 1, 2, 4, 7, 8

[44] Hailin Shi, Yang Yang, Xiangyu Zhu, Shengcai Liao, Zhen Lei, Weishi Zheng, and Stan Z Li. Embedding deep metric for person re-identification: A study against large variations. In *ECCV*, pages 732–748, 2016. 1

[45] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv*, abs/1409.1556, 2014. 1

[46] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *NIPS*, pages 1857–1865, 2016. 1, 2, 6, 7, 8

[47] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *CVPR*, pages 4004–4012, 2016. 1, 2, 6

[48] Yumin Suh, Bohyung Han, Wonsik Kim, and Kyoung Mu Lee. Stochastic class-based hard example mining for deep metric learning. In *CVPR*, pages 7251–7259, 2019. 2

[49] Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. Circle loss: A unified perspective of pair similarity optimization. In *CVPR*, pages 6398–6407, 2020. 2, 7, 8

[50] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, pages 1–9, 2015. 1, 8

[51] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge J Belongie. The Caltech-UCSD Birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. 6

[52] Faqiang Wang, Wangmeng Zuo, Liang Lin, David Zhang, and Lei Zhang. Joint learning of single-image and cross-image representations for person re-identification. In *CVPR*, pages 1288–1296, 2016. 1

[53] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. In *CVPR*, pages 5265–5274, 2018. 1, 2

[54] 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 *CVPR*, pages 1386–1393, 2014. 2, 8

[55] Jian Wang, Feng Zhou, Shilei Wen, Xiao Liu, and Yuanqing Lin. Deep metric learning with angular loss. In *ICCV*, pages 2593–2601, 2017. 2, 7, 8

[56] Xiaolong Wang and Abhinav Gupta. Videos as space-time region graphs. In *ECCV*, pages 399–417, 2018. 2

[57] Xun Wang, Xintong Han, Weilin Huang, Dengke Dong, and Matthew R Scott. Multi-similarity loss with general pair weighting for deep metric learning. In *CVPR*, pages 5022–5030, 2019. 2, 7, 8

[58] Xinshao Wang, Yang Hua, Elyor Kodirov, Guosheng Hu, Romain Garnier, and Neil M Robertson. Ranked list loss for deep metric learning. In *CVPR*, pages 5207–5216, 2019. 7, 8

[59] Kilian Q Weinberger and Lawrence K Saul. Distance metric learning for large margin nearest neighbor classification. *JMLR*, 10(2):207–244, 2009. 2

[60] Chao-Yuan Wu, R Manmatha, Alexander J Smola, and Philipp Krähenbühl. Sampling matters in deep embedding learning. In *ICCV*, pages 2859–2867, 2017. 2, 7, 8

[61] Hong Xuan, Richard Souvenir, and Robert Pless. Deep randomized ensembles for metric learning. In *ECCV*, pages 723–734, 2018. 2, 7, 8

[62] Hong Xuan, Abby Stylianou, and Robert Pless. Improved embeddings with easy positive triplet mining. In *WACV*, pages 2474–2482, 2020. 2

[63] Zhilin Yang, Jake Zhao, Bhuwan Dhingra, Kaiming He, William W Cohen, Ruslan R Salakhutdinov, and Yann LeCun. Glomo: unsupervised learning of transferable relational graphs. In *NIPS*, pages 8950–8961, 2018. 2

[64] Baosheng Yu and Dacheng Tao. Deep metric learning with tuplet margin loss. In *ICCV*, pages 6490–6499, 2019. 2

[65] Yuhui Yuan, Kuiyuan Yang, and Chao Zhang. Hard-aware deeply cascaded embedding. In *ICCV*, pages 814–823, 2017. 2, 7, 8

[66] Yiru Zhao, Zhongming Jin, Guo-jun Qi, Hongtao Lu, and Xian-sheng Hua. An adversarial approach to hard triplet generation. In *ECCV*, pages 501–517, 2018. 2, 7, 8

[67] Wenzhao Zheng, Zhaodong Chen, Jiwen Lu, and Jie Zhou. Hardness-aware deep metric learning. In *CVPR*, pages 72–81, 2019. 1, 2, 6, 7, 8

[68] Jiahuan Zhou, Pei Yu, Wei Tang, and Ying Wu. Efficient online local metric adaptation via negative samples for person re-identification. In *ICCV*, pages 2420–2428, 2017. 1