Deep Compositional Metric Learning

Wenzhao Zheng, Chengkun Wang, Jiwen Lu*, Jie Zhou
Department of Automation, Tsinghua University, China
Beijing National Research Center for Information Science and Technology, China
{zhengwz18, wck20}@mails.tsinghua.edu.cn; {lujiwen, jzhou}@tsinghua.edu.cn

Abstract

In this paper, we propose a deep compositional metric learning (DCML) framework for effective and generalizable similarity measurement between images. Conventional deep metric learning methods minimize a discriminative loss to enlarge interclass distances while suppressing intraclass variations, which might lead to inferior generalization performance since samples even from the same class may present diverse characteristics. This motivates the adoption of the ensemble technique to learn a number of sub-embeddings using different and diverse subtasks. However, most subtasks impose weaker or contradictory constraints, which essentially sacrifices the discrimination ability of each sub-embedding to improve the generalization ability of their combination. To achieve a better generalization ability without compromising, we propose to separate the sub-embeddings from direct supervisions from the subtasks and apply the losses on different composites of the sub-embeddings. We employ a set of learnable compositors to combine the sub-embeddings and use a self-reinforced loss to train the compositors, which serve as relays to distribute the diverse training signals to avoid destroying the discrimination ability. Experimental results on the CUB-200-2011, Cars196, and Stanford Online Products datasets demonstrate the superior performance of our framework. [1]

1. Introduction

Learning a discriminative and generalizable metric to compute the distances between images is a long-standing problem in computer vision, which serves as the foundation to a variety of tasks such as face clustering [18, 60, 66], person re-identification [6, 7, 72] and image retrival [47, 49, 70]. The objective of metric learning is to compress samples from the same class and maintain a margin between different classes in the learned metric space [9, 19, 61].

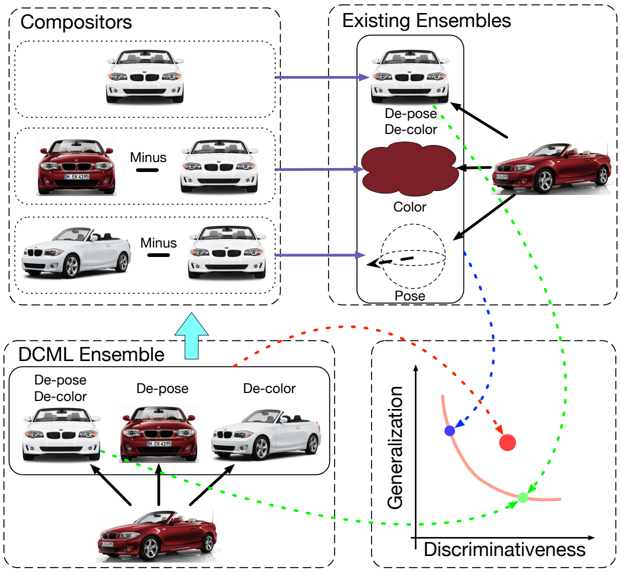

Figure 1. Motivation of the proposed DCML framework. "De-" stands for "remove" where embeddings usually discard intra-class variations such as pose and color for discriminativeness, while existing ensemble-based deep metric learning methods learn a set of sub-embeddings by directly performing different subtasks on them to capture diverse data characteristics including intraclass variations such as colors and poses. They generally compromise on the discriminativeness of the metric to improve the generalization ability. Differently, we only require a set of adaptive composites of the sub-embeddings to perform well on subtasks and simultaneously impose a discriminative constraint on the concatenation of the sub-embeddings. Our framework can improve the generalization of the learned metric without sacrificing the discriminativeness.

Recently, deep metric learning (DML) methods achieve outstanding performance by exploiting the powerful representation ability of deep convolutional neural networks (CNNs) [22, 29, 46, 52] to transform an image to the corresponding embedding, where the distance metric is defined as the Euclidean distance between embeddings.

Losses in deep metric learning are generally highly discriminative, which encourage small intraclass variations and large interclass margins in order to make the learned metric robust to differentiate samples from differ-

*Corresponding author
[1] Code: https://github.com/wzzheng/DCML

ent classes [10, 11, 55, 61]. However, samples even from the same class can show different formations and present different characteristics, so suppressing the intraclass variations may in turn undermine the ability of the learned metric to generalize to *unseen* classes, as verified by a number of recent works [43, 65]. To alleviate this issue, some methods turn to the ensemble technique and employ a number of learners to map each image to several sub-embeddings [36, 44, 64]. They train the learners with different subtasks which might be contradictory to the discriminative objective of metric learning, e.g., clustering samples from different classes to learn more general concepts or distinguishing samples from the same class to preserve more intraclass details. This essentially sacrifices the discrimination ability of each sub-embedding for better generalization ability of their combination, which raises a natural question: *how can we improve the generalization ability without compromising on the discrimination ability?*

In this work, we provide a deep compositional metric learning (DCML) framework as a positive solution, as illustrated in Figure 1. Instead of directly imposing the contradictory constraints of the subtasks on the sub-embeddings, we relax the constraints and propose to apply the losses on different composites of them. We adaptively learn a set of compositors to effectively re-weight all the sub-embeddings to obtain the corresponding composites. The compositors are randomly initialized and trained with a self-reinforced objective to enlarge the diversity as well as try to make the composites perform well on the downstream subtasks, which act as relays to re-balanced the training signals to better instruct the sub-embeddings towards better generalization. We simultaneously impose the discriminative constraints of conventional metric learning objective on the concatenation of sub-embeddings to maintain the discrimination ability. The sub-embeddings remain discriminative while preserving certain generalizable characteristics to enable the composites to complete various tasks. The overall framework of the proposed DCML can be trained efficiently in an end-to-end manner, and we directly use the concatenation of sub-embeddings to measure the similarity during testing, which requires no additional resources compared to conventional methods. We perform extensive experiments on the widely-used CUB-200-2011, Cars196, and Stanford Online Products datasets which demonstrate that our framework achieves very competitive performance.

2. Related Work

Deep Metric Learning: Deep metric learning aims to construct an effective embedding space to reflect the semantic distances among images. Various methods focus on the design of a discriminative loss on the embeddings to enlarge the interclass Euclidean distance and reduce the intraclass Euclidean distance [5, 12, 17, 45, 47, 49, 57, 58, 67, 67]. For example, the commonly used triplet loss [8, 45, 56] imposes a distance ranking between the positive pair and the negative pair within a triplet and requires a margin between them to improve robustness. To mine richer relations among samples, Sohn *et al.* [47] generalized the triplet loss and considered an (N+1)-tuple all together to simultaneously push away multiple samples. Cakir *et al.* [5] designed a FastAP loss to directly optimize the average precision over a list of samples to punish falsely ranked examples.

With the large number of possible tuples in the training data, sampling for more effective samples has been proven to be particularly helpful for deep metric learning methods [13, 14, 16, 31, 32, 35, 50, 63, 69–71]. For example, hard negative mining improves the performance and convergence speed of the triplet loss by selecting discriminative negatives that are considered challenging for the current metric [16, 21, 24, 45, 68]. Recently, Xuan *et al.* [65] observed that the use of easy positive samples can preserve the intraclass variations and thus improve the generalization ability of the triplet loss. However, the use of easy positives constantly under-challenges the metric resulting in a less discriminative embedding space.

Ensemble Learning: Ensemble learning combines the outcomes of several weak learners for the final prediction, which has been proven to be effective in a variety of machine learning tasks such as supervised learning [36, 39, 39], reinforcement learning [4, 30, 62], and unsupervised learning [15, 23, 53]. It is based on the observation that the combination of a set of weak learners can often achieve better generalization than the best single learner [20]. Recent methods incorporate ensemble learning into deep metric learning to boost the generalization performance, which instantiate different learners by partitioning the last layer [1, 33, 36, 64], using features from different layers [68], or employing different attention modules [27]. They design different subtasks to train each learner to encode different characteristics from the images. For example, Opitz *et al.* [36] re-weighted each sample adaptively based on previous learners so that hard samples receives a larger weight. Sanakayeu [44] employed a divide-and-conquer strategy to first divide the embedding space to several clusters and use each cluster to train a single learner.

Ensemble-based deep metric learning methods achieve superior generalization ability by forcing the sub-embeddings to preserve different characteristics in order to complete various subtasks. However, explicitly constraining the embedding to encode more intraclass features inevitably reduces the discrimination ability, leading to a metric susceptible to noise. Differently, our framework simultaneously learns a set of compositors in a self-reinforced manner and imposes the auxiliary constraints on the composites of sub-embeddings which can improve the generalization of the metric without compromising on the discriminativeness.

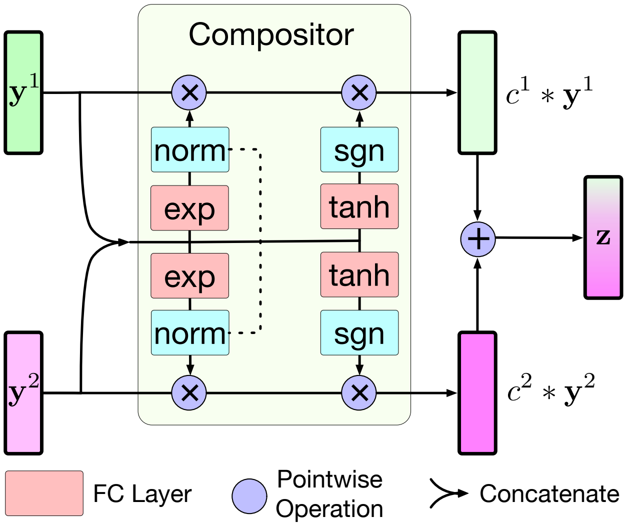

Figure 2. Design of the proposed compositor. We concatenate all the sub-embeddings as input to the compositor to aggregate global information. The compositor adaptively produces a weight for each sub-embedding and then computes the weighted average.

3. Proposed Approach

In this section, we first formulate the problem of deep metric learning and provide a unified view of existing ensemble-based deep metric learning methods. Then, we present the learnable re-weighting method of sub-embeddings and the self-reinforced training scheme of compositors to enlarge diversity. Lastly, we elaborate on the proposed deep compositional metric learning framework.

3.1. Revisit of Ensemble-based DML

Consider an image set $\mathbf{X}$ composed of N training samples $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N$ with their corresponding ground truth labels $\mathbf{L} = \{l_1, l_2, \cdots, l_N\}$, where $l_i \in \{1, 2, \cdots, n\}$ indicates that $\mathbf{x}_i$ belongs to the l_ith class. Deep metric learning exploits the strong representation ability of convolutional neural networks to transform each image $\mathbf{x}_i$ to a corresponding embedding $\mathbf{f}(\mathbf{x}_i) = \mathbf{y}_i$, where the learned metric is defined as the Euclidean distance between embeddings:

$$D(\mathbf{x}_i, \mathbf{x}_j) = ||\mathbf{f}(\mathbf{x}_i) - \mathbf{f}(\mathbf{x}_j)||_2 = ||\mathbf{y}_i - \mathbf{y}_j||_2, \qquad (1)$$

where $|| \cdot ||_2$ denotes the L2 norm.

The objective of deep metric learning generally punishes large intraclass distances and small interclass distances:

$$J(\mathbf{y}; \theta) = \sum_{l_i = l_j} p(D(\mathbf{x}_i, \mathbf{x}_j)) - \sum_{l_i \neq l_j} q(D(\mathbf{x}_i, \mathbf{x}_j)), \qquad (2)$$

where θ denotes all the parameters in the metric network, and p, q are two positive monotonically increasing function which determine the specific loss formulation.

We see the metric objective tends to suppress intraclass variations and encourage large margins between classes to make the learned metric more discriminative and robust, which might simultaneously discard the key features that enable the metric to generalize well to unseen classes during testing. This motivates the ensemble-based deep metric learning methods to employ K leaners to obtain a set of sub-embeddings $\{\mathbf{y}_i^1 = \mathbf{g}_1(\mathbf{x}_i), \mathbf{y}_i^2 = \mathbf{g}_2(\mathbf{x}_i), \cdots, \mathbf{y}_i^K = \mathbf{g}_K(\mathbf{x}_i)\}$ for an image $\mathbf{x}_i$ to encode diverse features.

We summarize existing ensemble-based methods as performing various subtasks on different sub-embeddings and provide a unified view by differentiating them by the design of learners, the sampling of training data, and the assignment of auxiliary labels. Formally, the overall objective of ensemble-based methods can be formulated as follows:

$$J(\{\mathbf{y}^k\}; \theta) = \sum_{k=1}^{K} \lambda_k J_{T_k}(\mathbf{g}_k, \widetilde{\mathbf{X}}_k, \widetilde{\mathbf{L}}_k) + \lambda_{div} \sum_{i \neq j = 1}^{K} J_{div}(\mathbf{g}_i, \mathbf{g}_j), \qquad (3)$$

where $\theta = \cup_{k=1}^{K} \phi_k$ with ϕ_k denoting the parameters of the kth learner, J_{T_k} denotes the objective of the kth subtask, J_{div} is a loss to encourage diversity among different sub-embeddings [27, 36, 40], $\{\lambda_k\}$ and λ_{div} are hyperparameters to balance the effects of different losses, $\widetilde{\mathbf{X}}_k$ is a re-sampling of the training data, and $\widetilde{\mathbf{L}}_k$ is a re-assignment of the ground truth labels. Note that different learners can share part of the parameters. After training, they concatenate all the sub-embeddings $[\mathbf{y}_i^{1^T}, \mathbf{y}_i^{2^T}, \cdots, \mathbf{y}_i^{K^T}]^T$ as the final embedding $\mathbf{f}(\mathbf{y})$ for testing.

Most existing methods focus on improving the ensemble design from one of these aspects. For example, ABE [26] employs $\{\mathbf{g}_k = \mathbf{E}(\mathbf{A}_k(\mathbf{S}))\}$, where $\mathbf{E}$ is a shared fully connected layer, $\{\mathbf{A}_k\}$ is a set of different attention modules, and $\mathbf{S}$ is a shared CNN. D & C [44] uses $\widetilde{\mathbf{X}}_k = \{\mathbf{x} | \mathbf{x} \in \mathbf{C}_k\}$ where $\mathbf{C}_k$ denotes the kth cluster obtained by performing K-means on the concatenated embeddings. DREML [64] adopts $\widetilde{\mathbf{L}}_k = \{l_i | l_i \sim l_{r(i)}\}$ where r is a random mapping to $\{1, 2, \cdots, n\}$ to cluster samples from different classes.

3.2. Learning to Compose

We find that all existing ensemble-based methods perform subtasks directly on the sub-embeddings, which generally impose weaker or contradictory constraints compared to the original objective, leading to a less discriminative metric. Therefore, they essentially sacrifice the discriminativeness and robustness for better generalization ability.

It seems that improving the generalization ability of the embeddings by forcing them to perform well on subtasks would inevitably reduce the discrimination ability, since the metric is directly defined over the concatenation of sub-embeddings. However, we argue that though the subtasks are needed for preserving diverse properties for better generalization, the sub-embeddings do not need to explicitly perform well on them. We deem a set of sub-embeddings with good generalization if we can extract enough diverse

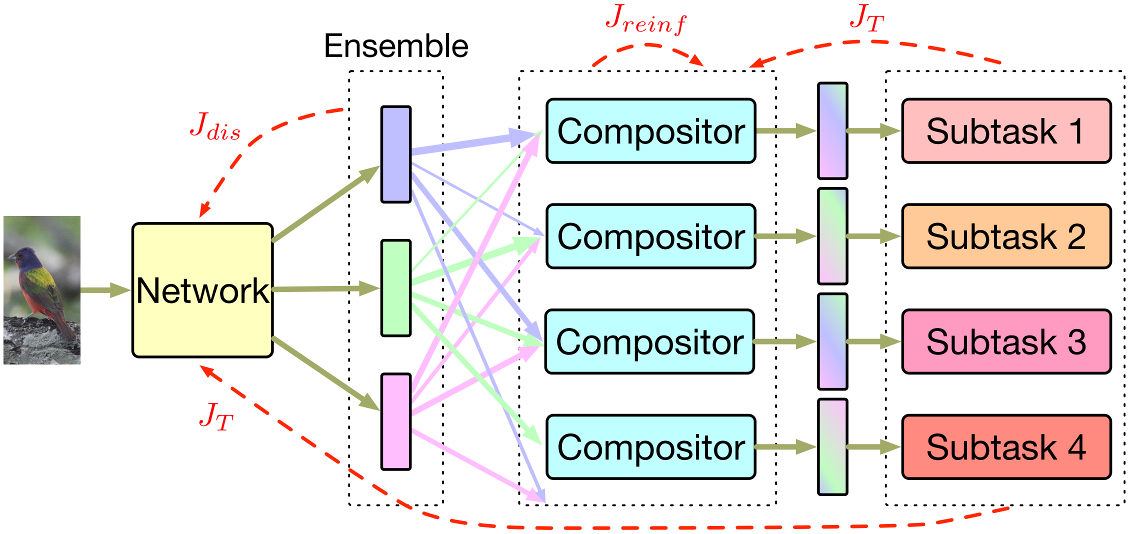

Figure 3. An illustration of the proposed DCML framework. We employ a convolutional neural network with a number of learners to produce an ensemble of sub-embeddings for each image. Instead of directly performing subtasks on the sub-embeddings, we use a set of compositors to diversely and adaptively re-weight each sub-embeddings to obtain the composites. We then impose the subtask losses on the composites as well as a discriminative metric loss on the concatenation of the sub-embeddings to train the metric network. The compositors are trained using a self-reinforced loss to enlarge diversity and the subtask losses to improve the performance of the composites. The learners and the compositors work together to perform well on various subtasks to learn sub-embeddings with better generalization ability without compromising on the discrimination ability.

information from them. Motivated by this, we propose to impose the constraints from subtasks on the combinations of sub-embeddings and simultaneously apply a discriminative loss on the sub-embeddings. In other words, the sub-embeddings can still be discriminative as long as their combinations can encode various information.

To achieve this, we propose to employ M compositors $\{\mathbf{c}_m\}$ to construct different combinations of sub-embeddings, where each compositor considers all the sub-embeddings to produce a weight for each one. We use the concatenation $\mathbf{y}$ of all sub-embeddings as input to the compositor and employ a fully connected layer with the softmax activation function to compute the weighting score $\mathbf{t}_m$:

$$t_m^k(\mathbf{y}) = \frac{\exp({\mathbf{w}_m^k}^T \mathbf{y} + b_m^k)}{\sum_{k=1}^{K} \exp({\mathbf{w}_m^k}^T \mathbf{y} + b_m^k)}, \tag{4}$$

where t_m^k is the kth component of the weighting score $\mathbf{t}_m$.

We simultaneously employ another fully connected layer with $\tanh$ as the activation function and adopt a sign function to produce the sign score $\mathbf{s}_m$:

$$s_m^k(\mathbf{y}) = \mathrm{sgn}(\tanh({\hat{\mathbf{w}}_m^k}^T \mathbf{y} + \hat{b}_m^k)), \tag{5}$$

where $\mathrm{sgn}(x)$ is the sign function which outputs 1 if $x > 0$ and -1 otherwise, and s_m^k is the kth component of the sign score $\mathbf{s}_m$. For backpropagation, we customize the gradient of the sign function to directly pass the training signals to the tanh function similar to the straight-through estimator (STE) [2]. This is reasonable since tanh is a soft approximation of the sign function.

We then obtain the weights for each sub-embedding by multiplying the weighting scores and the sign scores, i.e., $c_m^k(\mathbf{y}) = t_m^k(\mathbf{y}) \cdot s_m^k(\mathbf{y})$, where c_m^k is the kth component of the compositor $\mathbf{c}_m$. The weighting score determines the contribution of each sub-embedding and the sign score determines the direction of the training signal. We illustrate the design of the proposed compositor in Figure 2.

We use the compositor to re-weight each individual sub-embedding to obtain a composite $\mathbf{z}_m$:

$$\mathbf{z}_m = h_m(\mathbf{x}) = \sum_{k=1}^{K} c_m^k(\mathbf{y}) g_k(\mathbf{x}), \tag{6}$$

which is then used to perform one subtask. Note that the absolute weights for all the sub-embeddings add up to 1, so the compositors serve as relays to allocate the training signals from the subtasks for better generalization. The number of compositors M is not necessarily equal to K, which introduces more flexibility to the use of different subtasks.

The key to improve the generalization ability for ensemble-based methods lies in the diversity among the ensembles, and existing methods usually employ a diversity loss to push sub-embeddings of the same sample away from each other [27, 36, 40]. However, we think increasing the distances among sub-embeddings alter the training signal of the sub-tasks and might further reduce the discrimination ability. So instead of directly manipulating the sub-embeddings, we propose a self-reinforced training scheme to guide the compositors towards larger diversity. We randomly initialize each compositor and then progressively reinforce its current selection of sub-embeddings:

$$J_{reinf}(\mathbf{c}_m;\psi_m) = -\log(\max_k(\{c_m^k\})), \tag{7}$$

where $\psi_m = \{\mathbf{w}_m^k, b_m^k, \hat{\mathbf{w}}_m^k, \hat{b}_m^k\}$ denotes the learnable parameters of $\mathbf{c}_m$.

J_{reinf} aims to increase the largest weight produced by each compositor, which progressively reinforce the current focus of sub-embeddings. As the compositors are randomly initialized, they assign the largest weights to different sub-embeddings at the start. J_{reinf} then constantly guides them towards different directions to promote diversity. The self-reinforced training scheme enables the set of compositors to constantly enlarge diversity and produce different combinations of sub-embeddings without directly affecting the relations among sub-embeddings.

3.3. Deep Compositional Metric Learning

We present the formulation of the our DCML framework, which is comprised of a set of learners $\{\mathbf{g}_k\}$ to obtain K sub-embeddings and a set of compositors $\{\mathbf{c}_m\}$ to produce M composites for subtasks, as illustrated in Figure 3.

We perform various subtasks on the composites and use the corresponding subtask losses $\{J_{T_m}\}$ to train the learners to encourage them to encoder more diverse features for better generalization. We additionally impose J_{T_m} on the compositor so that it can better combine the sub-embeddings to extract meaningful information. To maintain the discriminativeness of the metric, we further apply a discriminative metric objective J_{dis} on the concatenation of the sub-embeddings. The overall objective of the proposed DCML framework can be formulated as follows:

$$\begin{aligned}\min_{\theta,\psi} J(\theta,\psi) \;=\; & \min_{\theta} J_{dis}(\mathbf{y};\theta) + \lambda_r \min_{\psi} \sum_{m=1}^{M} J_{reinf}(\mathbf{c}_m;\psi_m) \\ & + \min_{\theta,\psi} \sum_{m=1}^{M} \lambda_m J_{T_m}(\mathbf{z}_m;\theta,\psi_m),\end{aligned} \tag{8}$$

where λ_r and $\{\lambda_m\}$ are pre-defined parameters to balance the contributions of different losses and $\psi = \cup_{m=1}^{M}\psi_m$ includes the parameters of all compositors.

We simultaneously train the learners $\{\mathbf{g}_k\}$ and the compositors $\{\mathbf{c}_m\}$. Though the compositors take as input the concatenation of all the sub-embeddings, we only back-propagate J_{T_m} through $g_k(\mathbf{x})$ (the embeddings) but not $c_m^k(\mathbf{y})$ (the compositors) to prevent the learners to manipulate the weights themselves. This forces the learners $\{\mathbf{g}_k\}$ to focus on learning sub-embeddings that capture diverse image characteristics, while the compositors aim to diversely combine the sub-embeddings to perform well on downstream subtasks. The learners and the compositors work together to improve the generalization ability of the metric as well as preserving the discrimination ability.

Our DCML framework is compatible with a variety of loss formulations and sampling strategies. For example, we can instantiate J_{dis} with the margin loss and use the DWS strategy [63] to select uniform examples for training:

$$J_{dis}(\mathbf{y};\theta) = \sum_{l_i=l_j}[D_{ij}-\alpha]_+ - \sum_{l_i\neq l_j} I(p(D_{ij}))[\beta - D_{ij}]_+, \tag{9}$$

where $[\cdot]_+ = \max(\cdot,0)$, $D_{ij} = D(\mathbf{x}_i,\mathbf{x}_j)$, α and β are two pre-defined margins, $I(p)$ is a random variable which has a probability of p to be 1 and outputs 0 otherwise, $p(d) = \min(\gamma, d^{2-n}[1-\frac{1}{4}d^2]^{\frac{3-n}{2}})$, and γ is a positive constant.

We can apply our framework to various existing ensemble-based methods by performing the corresponding subtasks [33, 40, 44, 64] on the composites. For example, by using the discriminative metric objective on the composites, our method can be viewed as a soft version of the D & C [44] method which employs non-overlaping samples to train the learners. Differently, our method can automatically learn diverse data distributions to train different learners by using the compositors to distribute the gradients from a training sample, so that each leaner can see all the training data only with different weights.

Note that we can perform the same task and employ the same loss on several different composites without rendering the same trivial composites. This is because the self-reinforced training scheme of the compositors encourages diversity among the composites, which helps further disentangle the features learned from the same task.

Though the constraints from subtasks might still be contradictory to the discriminative objective of metric learning, we employ a set of compositors to distribute the training signals and only require the combinations of sub-embeddings to perform well on the subtasks, which produces a weaker effect on the discrimination ability of the learned embeddings. Compared to existing ensemble-based methods, our DCML framework can improve the generalization ability of the metric without compromising on the discriminativeness.

4. Experiments

In this section, we conducted various experiments to evaluate the image retrieval and clustering performance of the proposed DCML framework on three widely-used benchmark datasets: CUB-200-2011 [54], Cars196 [28], and Stanford Online Products [49]. For fair comparisons with existing methods, we partitioned each dataset into the training and test subset with disjoint classes to evaluate the performance of our framework under a zero-shot setting. The **CUB-200-2011** dataset [54] contains 200 bird species of 11,788 images. We used the first 100 species of 5,864 images for training and the rest 100 species of 5,924 images for testing. The **Cars196** dataset [28] contains 196 car models of 16,185 images. We used the first 98 models of 8,054 images for training and the rest 98 models of 8,131 images for testing. The **Stanford Online Products** dataset [49] contains 22,634 products of 120,053 images. We used the first 11,318 products of 59,551 images for training and the rest 11,316 products of 60,502 images for testing.

4.1. Evaluation Metrics

We evaluated our framework on both image retrieval and clustering tasks following previous works [47, 58, 70]. For clustering, we computed the normalized mutual information (NMI), which is defined as the mutual information normalized by the average of the entropies of clusters and actual classes, i.e., $\text{NMI}(\Omega, \mathbb{C}) = \frac{2I(\Omega;\mathbb{C})}{H(\Omega)+H(\mathbb{C})}$. $\Omega = \{\omega_1, \cdots, \omega_K\}$ is the set of clusters and $\mathbb{C} = \{c_1, \cdots, c_K\}$ indicates the set of ground truth classes, where ω_i denotes the samples with the cluster label i, and c_j denotes the samples with the ground truth label j. For retrieval, we computed the Recall@Ks defined as the percentages of valid samples, where each sample is deemed valid if at least one positive sample is retrieved among its K nearest neighbors.

4.2. Implementation Details

We employed the PyTorch [37] framework to conduct all the experiments. We adopted the ImageNet [42] pretrained ResNet-50 [22] as the base CNN model and added 4 randomly initialized fully connected layers with the output dimension of 128 as the learners. We resized all images to 256 × 256 as inputs to the metric model. During training, we used 8 compositors and randomly initialized them with standard normal distribution. For fair comparisons, we instantiated each subtask as a simple discriminative task with the corresponding metric objective (i.e., $J_{T_m} = J_{dis}$), but more diverse tasks could be used to further improve the performance. We set all the λ_ms to 1.0 and λ_r to 0.05. We augmented the training images with random resized cropping to 224 × 224 and random horizontal flipping with 50% probability. We fixed the batch size to 112 and used the Adam optimizer with learning rates of 10^{-6} for the base CNN, all the learners, and all the compositors. We concatenated the four 128-dimension sub-embeddings for testing.

4.3. Results and Analysis

Diversity of the Compositors: The diversity of the learners is a crucial factor for the generalization performance of ensemble-based methods, as verified by a number of works [3, 27, 36, 64]. The proposed DCML framework achieves diversity by the self-reinforced training of the compositors which adaptively and diversely distribute the training signals from various subtasks. To study the diversity of the compositors, we conducted an experiment on the CUB-200-2011 dataset and analyzed the absolute weights of each compositor for the learners averaged by all the samples in the training dataset, as shown in Figure 4.

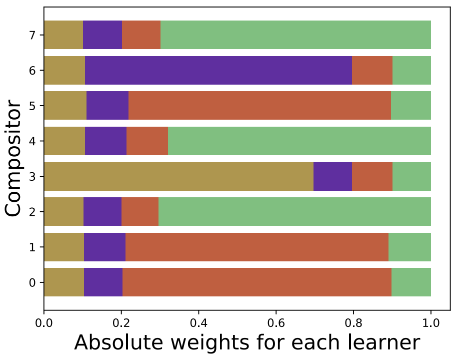

Figure 4. Weight analysis of compositors on CUB-200-2011.

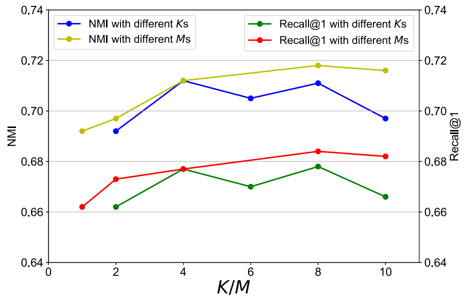

Figure 5. Analysis of different numbers of learners and compositors on CUB-200-2011.

Table 1. Results with different model settings on CUB-200-2011.

Method	R@1	R@2	R@4	R@8	NMI
DCML (w/o compositors)	66.2	76.9	85.8	91.3	70.1
DCML (w/o J_{reinf})	67.2	77.1	85.6	91.6	71.2
DCML (w/o $\hat{J}_T$)	67.4	77.2	85.6	91.5	71.4
DCML (w/o J_{dis})	68.2	77.8	86.0	91.4	71.7
DCML-MDW	**68.4**	**77.9**	**86.1**	**91.7**	**71.8**

Table 2. Results using different compositors on CUB-200-2011.

Method	R@1	R@2	R@4	R@8	NMI
Direcly using an FC layer	64.8	75.7	84.9	90.9	68.9
Exp → ReLU	67.1	77.4	86.0	91.6	71.1
W/o tanh	67.9	77.8	**86.2**	**91.8**	71.7
DCML-MDW	**68.4**	**77.9**	86.1	91.7	**71.8**

We observe that different compositors learn to concentrate on diverse learners and each learner is at least densely selected by one compositor. We also observe that each compositor mainly focuses on one learner but produces different weights for other learners, which is consistent with the self-reinforced training scheme of the compositors. Note that the weights are averaged by all the training samples, and each compositor can emphasize different learners for different samples. We further computed the average normalized distance between each of the sub-embeddings as 0.823 and 0.534 with and without J_{reinf}, respectively, which verifies the diversity of the ensemble.

Effect of the Numbers of Learners and Compositors: We conducted experiments on the CUB-200-2011 dataset to analyze the effect of different numbers of learners and compositors in our DCML framework with the margin loss.

Table 3. Experimental results (%) on the CUB-200-2011 dataset compared with state-of-the-art methods.

Methods	Size	R@1	R@2	R@4	R@8	NMI
Lifted [47]	64	43.6	56.6	68.6	79.6	56.5
Clustering [48]	64	48.2	61.4	71.8	81.9	59.2
N-Pair [47]	512	50.1	63.3	74.3	83.2	60.4
DVML [31]	512	52.7	65.1	75.5	84.3	61.4
Angular [57]	512	53.6	65.0	75.3	83.7	61.0
HDML [70]	512	53.7	65.7	76.7	85.7	62.6
HTL [16]	512	57.1	68.8	78.7	86.5	-
RLL-H [59]	512	57.4	69.7	79.2	86.9	63.6
HTG [69]	512	59.5	71.8	81.3	88.2	-
Margin [63]	128	63.6	74.4	83.1	90.0	69.0
SoftTriple [38]	512	65.4	76.4	84.5	90.4	69.3
Multi-Sim [58]	512	65.7	77.0	86.3	91.2	-
MIC [40]	128	66.1	76.8	85.6	-	69.7
DR [34]	512	66.1	77.0	85.1	91.1	-
CircleLoss [51]	512	66.7	77.4	86.2	91.2	-
RankMI [25]	128	66.7	77.2	85.1	91.0	71.3
PADS [41]	128	67.3	78.0	85.9	-	69.9
Ensemble-based methods:						
HDC [68]	384	53.6	65.7	77.0	85.6	-
A-BIER [36]	512	57.5	68.7	78.3	86.2	-
ABE-8 [27]	512	60.6	71.5	79.8	87.4	-
Ranked [59]	1536	61.3	72.7	82.7	89.4	66.1
DREML [64]	9216	63.9	75.0	83.1	89.7	67.8
D & C [44]	128	65.9	76.6	84.4	90.6	69.6
Triplet-R [56]	512	59.5	71.8	81.8	88.7	65.6
DCML-TR	512	**62.0**	**73.9**	**82.9**	**89.9**	**66.8**
Triplet-SH [45]	512	62.1	74.0	83.5	89.9	67.1
DCML-TSH	512	**64.8**	**75.8**	**84.2**	**90.3**	**67.9**
ProxyNCA [35]	512	64.2	75.5	83.9	89.8	67.9
DCML-PN	512	**65.2**	**76.4**	**84.8**	**90.7**	**68.8**
Margin-DW [63]	512	66.2	77.2	86.0	91.3	69.7
DCML-MDW	512	**68.4**	**77.9**	**86.1**	**91.7**	**71.8**

Table 4. Experimental results (%) on the Cars196 dataset compared with state-of-the-art methods.

Methods	Size	R@1	R@2	R@4	R@8	NMI
Lifted [47]	64	53.0	65.7	76.0	84.3	56.9
Clustering [48]	64	58.1	70.6	80.3	87.8	59.0
N-Pair [47]	512	71.1	79.7	86.5	91.6	64.0
Angular [57]	512	71.3	80.7	87.0	91.8	62.4
RLL-H [59]	512	74.0	83.6	90.1	94.1	65.4
HTG [69]	512	76.5	84.7	90.4	94.0	-
HDML [70]	512	79.1	87.1	92.1	95.5	69.7
Margin [63]	128	79.6	86.5	91.9	95.1	69.1
HTL [16]	512	81.4	88.0	92.7	95.7	-
DVML [31]	512	82.0	88.4	93.3	96.3	67.6
MIC [40]	128	82.6	89.1	93.2	-	68.4
RankMI [25]	128	83.3	89.8	93.8	96.5	69.4
CircleLoss [51]	512	83.4	89.8	94.1	96.5	-
PADS [41]	128	83.5	89.7	93.8	-	68.8
Multi-Sim [58]	512	84.1	90.4	94.0	96.5	-
SoftTriple [38]	512	84.5	90.7	94.5	96.9	70.1
DR [34]	512	85.0	90.5	94.1	96.4	-
Ensemble-based methods:						
HDC [68]	384	73.7	83.2	89.5	93.8	-
A-BIER [36]	512	82.0	89.0	93.2	96.1	-
Ranked [59]	1536	82.1	89.3	93.7	96.7	71.8
D & C [44]	128	84.6	90.7	94.1	96.5	70.3
ABE-8 [27]	512	85.2	90.5	94.0	96.1	-
DREML [64]	9216	86.0	91.7	95.0	97.2	76.4
Triplet-R [56]	512	76.1	85.2	91.2	95.4	67.1
DCML-TR	512	**79.2**	**87.9**	**93.1**	**96.3**	**68.5**
Triplet-SH [45]	512	80.4	88.6	93.8	96.9	70.8
DCML-TSH	512	**82.5**	**90.6**	**95.1**	**97.6**	**72.2**
ProxyNCA [35]	512	79.8	88.7	93.8	97.1	69.3
DCML-PN	512	**81.2**	**89.8**	**94.6**	**97.2**	**70.9**
Margin-DW [63]	512	82.9	89.6	94.7	97.3	71.0
DCML-MDW	512	**85.2**	**91.8**	**96.0**	**98.0**	**73.9**

We first fix the number of compositors M to 4 and used $2, 4, 6, 8, 10$ learners to instantiate the metric. The green and blue lines in Figure 5 show the experimental results in the retrieval and clustering tasks, respectively. We see that using more learners do not necessarily produce better results, and using 4 learners achieve similar performance with using 8 learners. This is because the 4 compositors cannot fully instruct the training of all the learners when $K > 4$, since each compositor usually focuses on one learner.

Similarly, we fix the number of learners K to 4 with various numbers $M = 1, 2, 4, 8, 10$ of compositors and evaluated the retrieval and clustering performance, as shown by the red and yellow lines in Figure 5, respectively. We observe that the performance generally improves as more compositors are deployed. This demonstrates the effectiveness of the proposed compositor which can exploit more information for training by adaptively combining the sub-embeddings to construct diverse composites. Note that the proposed framework can achieve better performance with more compositors even when $M > 4$. This further shows the use of compositors is not trivial and essentially different from directly performing subtasks on the sub-embeddings.

Ablation Studies: We conducted an ablation study on the major components of our DCML framework as showed in Table 1, where DCML (w/o compositors) means we directly perform sub-tasks on the sub-embeddings without the compositors, and DCML (w/o $\hat{J}_T$) means we do not employ J_T to train the compositor. Experimental results verify the effectiveness of each component of our framework.

The objective of the compositors is to distribute the training signals from different sub-tasks and two important components are the magnitudes and signs of the signals, which we use the softmax module and the tanh module to produce, respectively. We also conducted an ablation study of the design of the compositor as shown in Table 2. Still, we acknowledge that other choices are possible as long as they can produce normalized weights and signs.

Comparisons with State-of-the-art Methods: We compared the proposed DCML framework with state-of-the-art deep metric learning methods including ensemble-based methods on both image retrieval and clustering tasks. We instantiated our framework with various loss formulations and sampling strategies, including the triplet loss with random sampling (Triplet-R), the triplet loss with semi-hard

Table 5. Experimental results (%) on the Stanford Online Products dataset compared with state-of-the-art methods.

Methods	Size	R@1	R@10	R@100	NMI
Lifted [47]	64	62.5	80.8	91.9	88.7
Clustering [48]	64	67.0	83.7	93.2	89.5
N-Pair [47]	512	67.7	83.8	93.0	88.1
Angular [57]	512	67.9	83.2	92.2	87.8
HDML [70]	512	68.7	83.2	92.4	89.3
DVML [31]	512	70.2	85.2	93.8	90.8
Margin [63]	128	72.7	86.2	93.8	90.7
RankMI [25]	128	74.3	87.9	94.9	90.5
HTL [16]	512	74.8	88.3	94.8	-
RLL-H [59]	512	76.1	89.1	95.4	89.7
FastAP [5]	512	76.4	89.1	95.4	-
PADS [41]	128	76.5	89.0	95.4	89.9
MIC [40]	128	77.2	89.4	95.6	90.0
Multi-Sim [58]	512	78.2	90.5	96.0	-
SoftTriple [38]	512	78.3	90.3	95.9	92.0
CircleLoss [51]	512	78.3	90.5	96.1	-
Ensemble-based methods:					
HDC [68]	384	70.1	84.9	93.2	-
A-BIER [36]	512	74.2	86.9	94.0	-
D & C [44]	128	75.9	88.4	94.9	90.2
ABE-8 [27]	512	76.3	88.4	94.8	-
Ranked [59]	1536	79.8	91.3	96.3	90.4
Triplet-R [56]	512	70.3	84.2	92.7	88.5
DCML-TR	512	**71.9**	**85.4**	**92.8**	**89.4**
Triplet-SH [45]	512	75.1	87.7	94.3	89.3
DCML-TSH	512	**76.0**	**88.5**	**94.5**	**90.1**
Margin-DW [63]	512	78.4	90.2	95.4	90.3
DCML-MDW	512	**79.8**	**90.8**	**95.8**	**90.8**

sampling (Triplet-SH), the ProxyNCA loss, and the margin loss with distance-weighted sampling (Margin-DW). Tables 3, 4, and 5 shows the experimental results on the widely used CUB-200-2011, Cars196, and Stanford Online Products datasets, respectively. We indicate the best results using red colors and the second best results using blue colors. We use bold numbers to highlight the improvement of our framework over the original method.

We observe a constant performance boost to different losses and sampling strategies with the proposed DCML framework. In particular, our framework combined with the margin loss with distance-weighted sampling achieves the best or second best results for both tasks on on all the datasets. Note that some ensemble-based methods obtain the best results by using a larger embedding size, but our method can still achieve comparable or even better performance with a relatively constrained embedding size. This is because the use of various compositors enables the embeddings to encode more diverse characteristics which can take better advantage of the embedding capacity.

Qualitative Results: We qualitatively demonstrate several retrieved examples from the CUB-200-2011, Cars196, and Stanford Online Products datasets in Figure 6. We see that our method can successfully retrieve positive samples despite various poses, backgrounds, colors, and viewpoints.

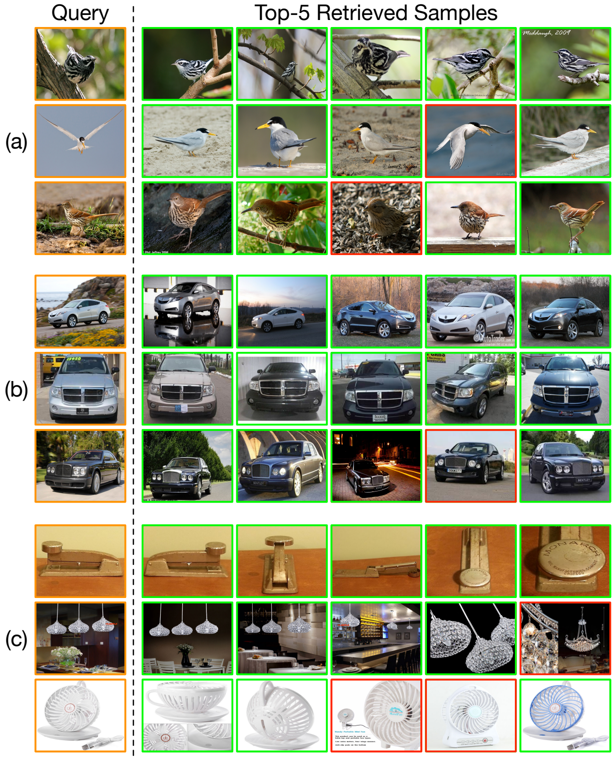

Figure 6. Qualitative retrieval results of the proposed DCML-MDW method on (a) CUB-200-2011, (b) Cars196, and (c) Stanford Online Products datasets. The green and red color at the border denotes a successful and failed retrieved sample, respectively.

5. Conclusion

In this paper, we have presented a deep compositional metric learning (DCML) framework to improve the generalization of the metric without compromising on the discriminativeness. We use an ensemble of sub-embeddings to represent an image and employ a set of compositors to diversely and adaptively combine the sub-embeddings to obtain composites, on which we impose auxiliary constraints to preserve more generalizable characteristics. We have performed experiments on three widely used datasets to analyze the effectiveness of our framework, which have demonstrated a constant performance boost to various losses and sampling strategies. It is interesting to employ the proposed compositional scheme in other directions such as deep hashing and self-supervised learning as future works.

Acknowledgement

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFA0700802, in part by the National Natural Science Foundation of China under Grant Grant U1813218, 61822603, Grant U1713214, in part by Beijing Academy of Artificial Intelligence (BAAI), and in part by a grant from the Institute for Guo Qiang, Tsinghua University.

References

[1] Nicolas Aziere and Sinisa Todorovic. Ensemble deep manifold similarity learning using hard proxies. In *CVPR*, pages 7299–7307, 2019. 2

[2] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv*, abs/1308.3432, 2013. 4

[3] Leo Breiman. Random forests. *ML*, 45(1):5–32, 2001. 6

[4] Jacob Buckman, Danijar Hafner, George Tucker, Eugene Brevdo, and Honglak Lee. Sample-efficient reinforcement learning with stochastic ensemble value expansion. In *NIPS*, pages 8224–8234, 2018. 2

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

[6] Guangyi Chen, Tianren Zhang, Jiwen Lu, and Jie Zhou. Deep meta metric learning. In *ICCV*, pages 9547–9556, 2019. 1

[7] 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

[8] 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

[9] Jason V Davis, Brian Kulis, Prateek Jain, Suvrit Sra, and Inderjit S Dhillon. Information-theoretic metric learning. In *ICML*, pages 209–216, 2007. 1

[10] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *CVPR*, pages 4690–4699, 2019. 2

[11] 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

[12] Thanh-Toan Do, Toan Tran, Ian Reid, Vijay Kumar, Tuan Hoang, and Gustavo Carneiro. A theoretically sound upper bound on the triplet loss for improving the efficiency of deep distance metric learning. In *CVPR*, pages 10404–10413, 2019. 2

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

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

[15] Haytham Elghazel and Alex Aussem. Unsupervised feature selection with ensemble learning. *ML*, 98(1-2):157–180, 2015. 2

[16] 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

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

[18] Senhui Guo, Jing Xu, Dapeng Chen, Chao Zhang, Xiaogang Wang, and Rui Zhao. Density-aware feature embedding for face clustering. In *CVPR*, pages 6698–6706, 2020. 1

[19] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, pages 1735–1742, 2006. 1

[20] Lars Kai Hansen and Peter Salamon. Neural network ensembles. *TPAMI*, 12(10):993–1001, 1990. 2

[21] 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

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1, 6

[23] Yi Hong, Sam Kwong, Yuchou Chang, and Qingsheng Ren. Unsupervised feature selection using clustering ensembles and population based incremental learning algorithm. *PR*, 41(9):2742–2756, 2008. 2

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

[25] 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

[26] Hyo Jin Kim, Enrique Dunn, and Jan-Michael Frahm. Learned contextual feature reweighting for image geo-localization. In *CVPR*, pages 3251–3260, 2017. 3

[27] 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, 3, 4, 6, 7, 8

[28] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *ICCVW*, pages 554–561, 2013. 5

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

[30] Thanard Kurutach, Ignasi Clavera, Yan Duan, Aviv Tamar, and Pieter Abbeel. Model-ensemble trust-region policy optimization. In *ICLR*, 2018. 2

[31] Xudong Lin, Yueqi Duan, Qiyuan Dong, Jiwen Lu, and Jie Zhou. Deep variational metric learning. In *ECCV*, pages 689–704, 2018. 2, 7, 8

[32] 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

[33] 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, 5

[34] 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

[35] 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, 7

[36] Michael Opitz, Georg Waltner, Horst Possegger, and Horst Bischof. Deep metric learning with bier: Boosting independent embeddings robustly. *TPAMI*, 2018. 2, 3, 4, 6, 7, 8

[37] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming

Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *NIPS*, pages 8026–8037, 2019. 6

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

[39] Lior Rokach. Ensemble methods in supervised learning. In *Data mining and knowledge discovery handbook*, pages 959–979. 2009. 2

[40] Karsten Roth, Biagio Brattoli, and Bjorn Ommer. Mic: Mining interclass characteristics for improved metric learning. In *ICCV*, pages 8000–8009, 2019. 3, 4, 5, 7, 8

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

[42] 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

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

[44] 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, 3, 5, 7, 8

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

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

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

[48] Hyun Oh Song, Stefanie Jegelka, Vivek Rathod, and Kevin Murphy. Deep metric learning via facility location. In *CVPR*, pages 2206–2214, 2017. 7, 8

[49] 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, 5, 6

[50] 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

[51] 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. 7, 8

[52] 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

[53] Apoorv Vyas, Nataraj Jammalamadaka, Xia Zhu, Dipankar Das, Bharat Kaul, and Theodore L Willke. Out-of-distribution detection using an ensemble of self supervised leave-out classifiers. In *ECCV*, pages 550–564, 2018. 2

[54] 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. 5

[55] 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. 2

[56] 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, 7, 8

[57] 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

[58] 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, 6, 7, 8

[59] 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

[60] Zhongdao Wang, Liang Zheng, Yali Li, and Shengjin Wang. Linkage based face clustering via graph convolution network. In *CVPR*, pages 1117–1125, 2019. 1

[61] Kilian Q Weinberger and Lawrence K Saul. Distance metric learning for large margin nearest neighbor classification. *JMLR*, 10(2):207–244, 2009. 1, 2

[62] Marco A Wiering and Hado Van Hasselt. Ensemble algorithms in reinforcement learning. *TSMC*, 38(4):930–936, 2008. 2

[63] Chao-Yuan Wu, R Manmatha, Alexander J Smola, and Philipp Krähenbühl. Sampling matters in deep embedding learning. In *ICCV*, pages 2859–2867, 2017. 2, 5, 7, 8

[64] Hong Xuan, Richard Souvenir, and Robert Pless. Deep randomized ensembles for metric learning. In *ECCV*, pages 723–734, 2018. 2, 3, 5, 6, 7

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

[66] Lei Yang, Xiaohang Zhan, Dapeng Chen, Junjie Yan, Chen Change Loy, and Dahua Lin. Learning to cluster faces on an affinity graph. In *CVPR*, pages 2298–2306, 2019. 1

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

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

[69] 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

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

[71] Wenzhao Zheng, Jiwen Lu, and Jie Zhou. Deep metric learning via adaptive learnable assessment. In *CVPR*, pages 2960–2969, 2020. 2

[72] 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