Dynamic Metric Learning with Cross-Level Concept Distillation

Wenzhao Zheng[1,2], Yuanhui Huang[1,2],
Borui Zhang[1,2], Jie Zhou[1,2], and Jiwen Lu[1,2,⋆]

[1] Department of Automation, Tsinghua University, China
[2] Beijing National Research Center for Information Science and Technology, China
{zhengwz18, huang-yh18, zhang-br21}@mails.tsinghua.edu.cn;
{jzhou, lujiwen}@tsinghua.edu.cn;

Abstract. A good similarity metric should be consistent with the human perception of similarities: a sparrow is more similar to an owl if compared to a dog but is more similar to a dog if compared to a car. It depends on the semantic levels to determine if two images are from the same class. As most existing metric learning methods push away interclass samples and pull closer intraclass samples, it seems contradictory if the labels cross semantic levels. The core problem is that a negative pair on a finer semantic level can be a positive pair on a coarser semantic level, so pushing away this pair damages the class structure on the coarser semantic level. We identify the negative repulsion as the key obstacle in existing methods since a positive pair is always positive for coarser semantic levels but not for negative pairs. Our solution, cross-level concept distillation (CLCD), is simple in concept: we only pull closer positive pairs. To facilitate the cross-level semantic structure of the image representations, we propose a hierarchical concept refiner to construct multiple levels of concept embeddings of an image and then pull closer the distance of the corresponding concepts. Extensive experiments demonstrate that the proposed CLCD method outperforms all other competing methods on the hierarchically labeled datasets. Code is available at: https://github.com/wzzheng/CLCD.

1 Introduction

Measuring the similarity between images is a crucial step in the field of computer vision. Modern methods use deep neural networks such as Convolutional Neural Networks (CNNs) [43,48,22] or Vision Transformers (ViTs) [12,33,7] to extract an embedding vector to represent an image for similarity computing. The design of the model architecture is crucial, but how to train this model matters equally. As a widely used learning paradigm, deep metric learning aims to learn a discriminative embedding by reducing the distance between samples from the same class and enlarging the distance between samples from different classes, which has benefited various tasks including image retrieval [45,38,30,13], face recognition [23,41], and person re-identification [42,51,66,5].

⋆ Corresponding author.

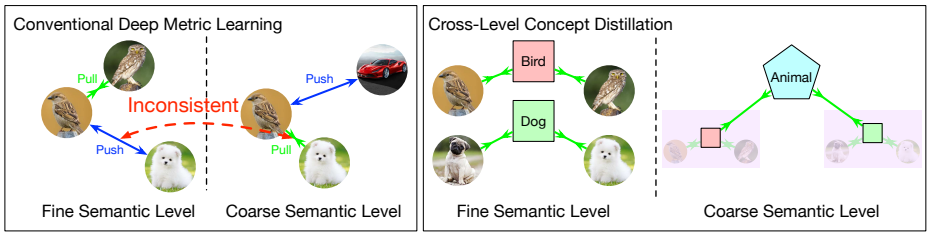

Fig. 1. The motivation of the proposed CLCD method. Conventional deep metric learning pulls closer samples from the same class and pushes away samples from different classes. This results in conflicts if we consider the class of images from different semantic levels. A pair image may be deemed dissimilar at a fine semantic level but similar at a coarse semantic level. To address this, we construct a hierarchy of concept embeddings and propose a CLCD method to distill higher-level concepts using the corresponding lower-level concepts. (Best viewed in color.)

Humans perceive concepts in a hierarchical way. We first recognize a sparrow as an animal, then as a bird, and finally as an owl. When we consider the similarities between images, the result varies at different semantic levels. A sparrow is more similar to an owl when compared to a dog, but is more similar to a dog than to a car. Therefore, the objective of deep metric learning, pulling closer positive pairs and pushing away negative pairs, seems reasonable within a single semantic level, but conflicts emerge when considering multiple semantic levels, as shown in Fig. 1. The sparrow-dog pair should be pushed away in a fine semantic level but instead should be pulled closer in a coarse semantic level. Sun *et al.* [47] recently identified this issue and formulated the dynamic metric learning (DyML) problem, where an image is assigned three labels in the coarse, middle, and fine level, respectively. The goal is to retrieve the correct samples with the same labels in all three semantic levels. They also proposed a recipe for this problem by setting increasing similarity margins to separate the positive and negative pair in the fine, middle, and coarse levels. They need to manually set a fixed margin to separate concepts from different levels, leading to rigid concept scopes at each semantic level.

Our solution, on the other hand, is free of hand-crafted margins. Given that the conflicts result from the positive attraction and negative repulsion across different semantic levels, we propose to completely discard the latter for pure harmony. That is, we only pull closer positive pairs, which is simple in concept but non-trivial to implement. To put this into practice, we propose a cross-level concept distillation (CLCD) method, which simultaneously learns multiple-level concept embeddings to guide the training of the image embedding. Specifically, we employ a hierarchical concept refiner to extract multiple concepts corresponding to different semantic levels for each image. We represent each concept using an embedding vector with the same size as the image embedding and treat the image embedding as a concept of instance level. We propose a cross-level concept distillation method to pull closer the cross-level concepts of two images under

the finest semantic level that they have the same label. The proposed CLCD avoids the cross-level conflict by only pulling closer positive samples and achieves discriminativeness by the hierarchical concept refining. We conduct extensive experiments on the three dynamic metric learning datasets: DyML-Animal, DyML-Vehicle, and DyML-Products [47], which show that our proposed CLCD achieves the best performance. We also demonstrate that a simple positive attraction loss in the proposed manner is effective to learn a discriminative embedding space and achieves comparable performance under the conventional deep metric learning setting on the widely-used CUB-200-2011 [50] dataset for image retrieval.

2 Related Work

Deep Metric Learning: Deep metric learning aims at learning a discriminative embedding space where intraclass distances are small and interclass distances are large. Existing methods achieve this by imposing different restrictions on the embedding space. A number of works directly constrain the distances between sample pairs [41,45,44,53,11,54,59,2,17,15]. For example, Schroff *et al.* [41] employed a triplet loss acting on three samples to enforce a margin on the distance between the positive pair and the negative pair. Sohn *et al.* [44] extended the triplet loss to an N-Pair loss which simultaneously constrains the relations between $N + 1$ samples from different classes. The vast number of combinations of samples causes the sampling of informative tuples to be an important component for deep metric learning. Some methods addressed this by using a carefully designed sampling strategy [16,41,57,24,62,21,60] or synthesis generation method [13,64,31,65,28], while other works reduced the sampling complexity by representing each class using proxies and instead restrict the relations between samples and proxies [34,39,27,11,32,52,9].

Most existing deep metric learning only consider the semantic similarity under a certain semantic level and a direct extension of existing methods leads to cross-level conflicts of pulling closer and pushing away the same pair of samples. Some methods [36,37,26] employ hyperbolic embeddings to effectively represent hierarchically structured data, yet they still cannot avoid the cross-level conflicts during training. This motivates Sun *et al.* [47] to formulate a dynamic metric learning task to consider the similarity measure under different semantic levels. They further proposed a Cross-Scale Learning (CSL) method to enforce increasing margins between the similarities between positive pairs and negative pairs for coarser and coarser semantic levels. They rely on manually set margins to differentiate concepts at different semantic levels. Differently, the proposed CLCD employs a hierarchical concept refiner to adaptively distill concepts by summarizing the corresponding lower-level concepts. In addition, SimSiam learns unsupervised image representations using only positive pairs. We demonstrate that only using positive attraction is also effective for supervised learning and further extend it to dynamic metric learning.

Hierarchical Image Classification: Another related area is the hierarchical image classification (HIC), which aims to predict the correct labels across

different semantic levels for an image [14,10,19,55,58]. It can be seen as a special case for multi-task learning [4] if we regard the multiple classification problems as different tasks. Most methods perform this task during training in order to better leverage the hierarchical annotations provided by various datasets [8,29] to improve the performance on the finest-level classification. For example, Verma *et al.* [49] added coarse-level metric matrices to obtain fine-grained-level metric matrices for hierarchical classification. Dutt *et al.* [14] proposed a partially merged network architecture to jointly learn classifiers at different semantic levels and employed a probability adjustment procedure to improve the performance. Yan *et al.* [58] designed a hierarchical deep convolutional neural network to complete the coarse and fine classification task progressively.

The task of hierarchical image classification is essentially different from dynamic metric learning. While HIC only requires the model to correctly predict the labels of different semantic levels, DyML further requires the model to obtain a single representation for an image so that it can properly reflect the similarities between images across semantic levels.

3 Proposed Approach

In this section, we first formulate the problem of dynamic metric learning and identify the cross-level conflicts caused by existing methods. We then present the proposed hierarchical concept refiner and cross-level concept distillation method as the two main components of our CLCD method.

3.1 Dynamic Metric Learning

For a set of images $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$, conventional metric learning only assumes a single label l_i for each image $\mathbf{x}_i$. Deep metric learning employs a deep network to obtain an n-dimentsion embedding $\mathbf{y} \in \mathbb{R}^d$ and then imposes discriminative constraints on the Euclidean distances between image embeddings:

$$\begin{cases} \text{Positive attraction (PA): } \min d(\mathbf{y}_i, \mathbf{y}_j), & \text{if } \ l_i = l_j, \\ \text{Negative repulsion (NR): } \max d(\mathbf{y}_i, \mathbf{y}_j), & \text{if } \ l_i \neq l_j, \end{cases} \tag{1}$$

where $d(\cdot, \cdot)$ denotes the Euclidean distance.

This seems reasonable for images with a single label, but what if an image is assigned multiple hierarchical labels at different semantic levels? This is common in reality, for example, a Ferrari can be classified as a car or a vehicle if we consider it at different semantic levels. Considering this, Sun *et al.* [47] formulates the dynamic metric learning (DyML) problem, aiming at learning an embedding space where images can be correctly retrieved across multiple semantic labels.

Formally, each image $\mathbf{x}_i$ is assigned a label set of K labels $\{l_i^1, \cdots, l_i^K\}$, where K is the number of the concerned semantic levels. We further assume a hierarchical structure in each label set, i.e., the coarser-level labels of two images are always the same if they share a label at a certain level:

$$l_i^k = l_j^k, \quad \forall k > t, \quad \text{if } \ \exists \ \ l_i^t = l_j^t. \tag{2}$$

This is reasonable since a coarse concept (e.g., animal) should include the fine concepts (e.g., bird, dog). We can then define the finest semantic level $\alpha(\mathbf{x}_i, \mathbf{x}_j)$ where two images share the same label:

$$\alpha(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} \arg\min_k (l_i^k = l_j^k), & \text{if } \exists \ \ l_i^k = l_j^k, \\ K+1, & \text{if } \ l_i \neq l_j \ \ \forall \ \ k, \end{cases} \tag{3}$$

The objective of DyML can be then formulated as:

$$d(\mathbf{y}_a, \mathbf{y}_p) < d(\mathbf{y}_a, \mathbf{y}_n), \quad \text{if } \ \alpha(\mathbf{x}_a, \mathbf{x}_p) < \alpha(\mathbf{x}_a, \mathbf{x}_n). \tag{4}$$

Despite the hierarchical structure of each label set, it is still possible for two images to have different fine-level labels but share a coarse-level label, i.e., $l_i^s \neq l_j^s$ but $l_i^t = l_j^t$ if $s < t$. Therefore, directly extending the objective of conventional deep metric learning (1) to multiple semantic levels would cause the NR under a fine level to be contradictory to the PA under a coarse level, rendering the learning process less effective.

To address this, Sun *et al.* [47] present a recipe by enforcing different margins between for negative pairs at different semantic levels:

$$d(\mathbf{y}_a, \mathbf{y}_p) + m(\alpha(\mathbf{x}_a, \mathbf{x}_n)) \leq d(\mathbf{y}_a, \mathbf{y}_n), \tag{5}$$

where $\alpha(\mathbf{x}_a, \mathbf{x}_p) = 1$, and $m(\cdot)$ is a positive monotonically increasing function.

Intuitively, it requires the dissimilar pairs at coarser levels to be separated with a larger margin. However, it requires a manual setting of the margins and enforces a handcrafted prior on the distances between concepts. Our solution is, on the other hand, free of margins: we only pull closer positive pairs.

3.2 Hierarchical Concept Refiner

To address the cross-level positive attraction and negative repulsion conflict dilemma, we propose an alternative solution to completely discard the negative repulsion at all layers. However, directly pulling closer positive pairs without the regularization of the reverse effect of negative repulsion, the trained model will quickly collapse to a trivial model that represents all images in a single point in the embedding space.

To avoid this, we propose to instead restrict the distances between concepts, where each concept corresponds to a label as well as a semantic level. We represent each concept using a vector $\mathbf{c}$ called the concept embedding. As each image is assigned a set of labels with a hierarchy structure, we propose a hierarchical concept refiner R to distill concepts directly from images, as shown in Fig. 2. The refiner R takes as input the image embedding $\mathbf{y}$ and outputs a set of concepts corresponding to each semantic level:

$$R(\mathbf{y}) = \{\mathbf{c}^0, \mathbf{c}^1, \mathbf{c}^2, \cdots, \mathbf{c}^K\}, \tag{6}$$

where $\mathbf{c} \in \mathbb{R}^n$ has the same dimension with the image embedding $\mathbf{y}$. For convenience, we also regard the image (and possibly its variants with different data augmentations) as a concept at the finest semantic level, i.e., $\mathbf{c}^0 = \mathbf{y}$.

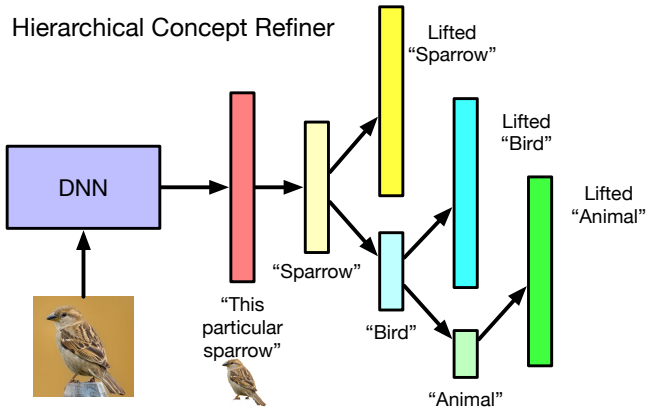

Fig. 2. Illustration of the proposed hierarchical concept refiner. For each image, we first use a deep neural network to obtain an image embedding and then employ a series of encoders to refine a hierarchy of meta-concept embeddings with decreasing dimensions. Finally, we use a set of decoders to map the meta-embeddings to the image embedding space for the sake of direct comparison. (Best viewed in color.)

Considering the hierarchical structure of labels, we design the refiner R accordingly in a hierarchical manner. Since a coarse concept may correspond to multiple concepts at a finer level, we refine the concepts progressively from fine level to coarse level. That is, we first discard certain information to purify an image to a fine concept, and then discard more information to purify the fine concept to a more coarse concept. We continue this process until we obtain a pure coarsest concept. For example, for an image of a sparrow, we first discard the sparrow-specific information to obtain the concept "bird", and then further discard the bird-specific information to obtain the concept "animal". On the other hand, we can add certain information to specify an "animal" concept to a "bird", and further specify it to a "sparrow".

Formally, we employ a series of fully connected layers with decreasing output dimensions to obtain a meta-concept embedding $\widetilde{\mathbf{c}}^i$ at each semantic level.

$$\begin{cases} \widetilde{\mathbf{c}}^0 = \mathbf{c}^0 = \mathbf{y} \in \mathbb{R}^n, \\ E_i(\widetilde{\mathbf{c}}^k) = \widetilde{\mathbf{c}}^{k+1} \in \mathbb{R}^{\beta(k+1)}, \quad k = 0, 1, \cdots, K-1, \end{cases} \tag{7}$$

where $\beta(\cdot)$ is a monotonically decreasingly function. We use $\beta(k) = \frac{n}{2^k}$ in this work, but other choices are also possible.

To enable direct comparison between concepts, we use a set of decoders to map the meta-concept embedding back to the n-dimension embedding space:

$$D_i(\widetilde{\mathbf{c}}^k) = \mathbf{c}^k \in \mathbb{R}^n, \quad k = 1, 2, \cdots, K. \tag{8}$$

Since we do not provide additional specific information to the decoders, the decoders only interpret a concept in a larger space but do not specify a concept to a finer one.

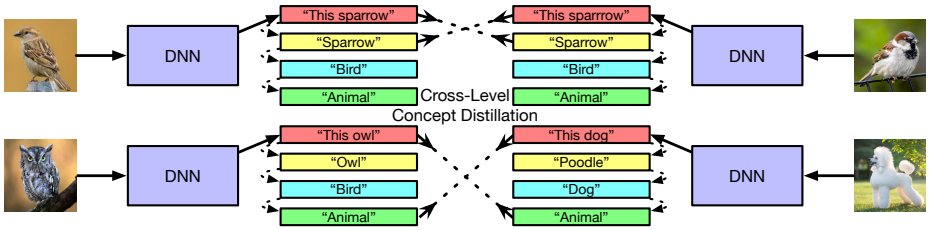

Fig. 3. The framework of the proposed CLCD method. We learn each concept by distilling information from its affiliated lower-level concepts. Having obtained the set of hierarchical concept embeddings for each image, we constrain the distance between the lower-level concept of one image and the higher-level concept of another image if they are from the same class at the higher semantic level. (Best viewed in color.)

Using the proposed hierarchical concept refiner, we can predict the concepts of an image at different semantic levels and represent them in the same space for further relational constraints.

3.3 Cross-Level Concept Distillation

The hierarchical design of the proposed concept refiner naturally constrain finer-level coarser-level concepts to contain less information and thereby to be more general, but how to learn each concept embedding remains challenging.

A straightforward way is to pull closer the distance between the corresponding positive concept embeddings at the same semantic level:

$$L_{naive} = d(\mathbf{c}_i^k, \mathbf{c}_j^k), \quad \text{if} \quad l_i^k = l_j^k. \tag{9}$$

However, the learning process in this way is only aware of same-level concepts and unaware of the lower-level concepts. The concept refiner is then able to bypass the distillation of lower-level concepts and only enforces relations between same-level concepts, ignoring the cross-level concept hierarchically structural relations.

Instead, we think that the formation of a concept requires examining a set of lower-level concepts and then summarizing their common grounds. For example, the "bird" concept should be able to access all the affiliated finer concepts such as "sparrow", "owl", and "pigeon", and then be distilled as a "bird" concept.

Motivated by this, we propose cross-level concept distillation to constrain the relations between cross-level concepts to learn the hierarchical concept refiner, as shown in Fig. 3. We further propose two strategies to refine the concept: adjacent concept refining (ACR) and instance-based concept refining (ICR). For ACR, we distill a higher-level concept by considering the corresponding adjacent level concepts. For ICR, we distill a high-level concept from all the corresponding instances that share this concept. Both strategies exploit the cross-level concept relations to distill new concepts, which only differ in extracting concepts from already constructed concepts or directly from instances.

For both strategies, we impose two constraints to learn the concept refiner. The first self-aware loss requires the multi-level concept embeddings of an image to reconstruct its lower-level concept embedding:

$$L_{self} = \sum_{k=1}^{K} d(\mathbf{c}^{\gamma(k)}, \mathbf{c}^{k}), \tag{10}$$

where $d(\cdot,\cdot)$ denotes the Euclidean distance,, and $\gamma(k) = k-1$ for ACR and $\gamma(k) = 0$ for ICR. The self-aware loss requires the concept of each level to reconstruct concepts of lower levels of the same image as much as possible so that only minimum irrelevant information is discarded during concept refining.

The self-aware loss alone is not enough to distill a concept, since we need to discard more information to construct a higher-level concept. Therefore, we further employ an inter-distillation loss to force a concept to only preserve the common knowledge that defines itself. We reduce the distance between the cross-level concepts of two images if they share the same label at a certain level:

$$L_{inter} = \sum_{j:\alpha(\mathbf{x}_i,\mathbf{x}_j)=k} d(\mathbf{c}_i^{\gamma(k)}, \mathbf{c}_j^{k}) + d(\mathbf{c}_i^{k}, \mathbf{c}_j^{\gamma(k)}). \tag{11}$$

The inter-distillation loss encourage each concept to discard more information to purify irrelevant knowledge, while the self-aware loss constrains each concept to preserve the instance information as much as possible. The two losses enforce the concept refiner to extract the most relevant information that defines a concept.

The abandonment of negative repulsion avoids the cross-level conflicts, but the absence of a counter-force could easily cause the model to collapse, bringing challenges to the optimization process. Motivated by the stop-gradient technique employed in a number of self-supervised methods [3,18,6], we detach the lower-level concept embeddings in the loss and only use them as targets: The overall objective of the CLCD can be formulated as:

$$\begin{aligned} L &= L_{self} + L_{inter}, \\ &= \sum_{k=1}^{K} d(\text{detach}(\mathbf{c}^{\gamma(k)}), \mathbf{c}^{k}) + \sum_{j:\alpha(\mathbf{x}_i,\mathbf{x}_j)=k} (d(\text{detach}(\mathbf{c}_i^{\gamma(k)}), \mathbf{c}_j^{k}) + d(\text{detach}(\mathbf{c}_i^{k}), \mathbf{c}_j^{\gamma(k)})), \end{aligned} \tag{12}$$

where $\text{detach}(x)$ denotes the detach operation where the gradients do not pass through x during back-propagation.

Our CLCD refines the multi-level concepts progressively in a hierarchical way and employs a cross-level distillation method to learn the concepts. The concepts at all semantic levels are learned jointly to preserve the hierarchical structure of the labels, which implicitly constrain the image embedding to share similar hierarchical distances with other semantically varied images.

3.4 Discussions

The preventing of collapsing: Why using a stop-gradient operation can prevent collapsing remains a mystery in the literature [3,18,6]. One hypothesis is

that the stop-gradient operation transforms the optimization into an implicit alternating optimization between two sets of variables. Applying the stop-gradient to the lower-level concept in our case converts it into a fixed target during each iteration, where the targets are probably different for different concepts thanks to the curse of dimensionality [1]. See Chen *et al.* [6] for more details.

Adaptive learning of the concept scope: Though our method does not explicitly push away negative pairs of concepts, the distances between negative pairs are naturally increased due to the clustering of intraclass concepts. Existing methods manually set a margin to control the scope of each concept. We argue that different concepts may occupy regions with different areas in the embedding space. For example, even at the same semantic level, the concept "animal" contains more diverse lower-level concepts than the concept "vehicle", and thus should spread out more. The proposed CLCD method adaptively learns the scope for each concept by using the cross-level concept distillation to train the concept encoders and decoders. The more diverse concepts are more difficult to compress, thereby being encoded to a larger area in order for the decoder to (attempt to) reconstruct the lower-level concepts.

Sampling of mini-batches: The proposed inter-distillation loss (11) acts on pairs of positive concepts at multiple semantic levels, yet the number of negative pairs is far larger than that of the positive pairs, bringing challenges to the sampling process. To achieve balanced learning of all concepts, we employ a hierarchical sampling strategy to guarantee the existence of positive pairs across all the semantic levels. To sample a mini-batch of B images with a label set of K levels, we first sample $\frac{B}{2^K}$ K-level labels, and then for each k-level label, we randomly select two $(k-1)$-level labels until reaching the first (finest) level, where we sample two images.

4 Experiments

In this section, we conduct extensive experiments to evaluate the performance of the proposed CLCD method on the dynamic metric learning task, which aims to learn a versatile similarity metric that is able to perform well across different semantic sales. We demonstrate that using a simple positive attraction loss under our framework achieves comparable performance on the conventional deep metric learning setting. We additional provide an in-depth experimental analysis to demonstrate the effectiveness of our framework.

4.1 Datasets

We follow existing work to conduct experiments on the three dynamic metric learning datasets: DyML-Vehicle, DyML-Animal, and DyML-Product [47]. The images in each dataset are labeled with three hierarchical labels corresponding to three semantic scales (i.e., coarse, middle, and fine). We follow existing work to perform the dataset split for fair comparisons. Specifically, the class labels for the coarse scale on the training and test split have a low intersection, while the

training and test labels are disjoint for the middle and fine scale. We detail the dataset setting in the supplementary material.

4.2 Evaluation Protocol

To evaluate the performance of the learned metric across all the semantic levels, we first test the performance under each semantic level and then compute the average of all levels. Specifically, we adopt the widely used Recall@Ks and mean Average Precision (mAP) for the image retrieval tasks under each level. The recall@Ks compute the percentage of images in the query set that has at least one correct retrieved sample with the sample label from the K nearest neighbors in the gallery set. The mAP first computes the average precision score for each correct retrieved sample for a ranked list in the query set and then takes the mean across all the samples in the query set. Note we omit the average set intersection (ASI) metric used in the original paper [47] as the computing requires the ground truth ranking list of each image which is not provided in the datasets.

4.3 Implementation Details

We conducted all the experiments using the PyTorch package. We followed Sun *et al.* [47] to adopt the ResNet-34 as the backbone CNN model, where uses the ImageNet-1K [40] pretrained weights on the DyML-Vehicle and DyML-Product datasets and randomly initialized weights on the DyML-Animal dataset. Following the backbone CNN, we added an adaptive max pooling layer a randomly initialized fully connected layer to obtain a 512-dimension image embedding, and set the concept embedding sizes to 256, 128, and 64 for the fine, middle, and coarse semantic levels, respectively. We then added an L2-normalization layer after each image embedding and concept embedding before distance computation. We normalized all the images to 256×256 as inputs to the CNN model. For training, we performed data augmentation to images with random cropping to 227×227 and random horizontal mirror with a possibility of 0.5. We set the learning rate to 10^{-5} for the backbone CNN, 10^{-4} for the following fully connected layer, and 10^{-2} for the encoders and decoders. We only use the refiner and the multi-level concepts during training and simply use the image representation $\mathbf{y}$ from the backbone during evaluation. The multi-level concepts serve as targets to train the image representation and are discarded after training.

4.4 Main Results

We compare our CLCD with all the methods provided by the dynamic metric learning benchmark [47] as shown in Table 1, including the cross-level deep metric learning method CSL [47], conventional deep metric learning methods (the triplet loss [41], the Multi-Sim loss [54], and the N-Pair loss [44]), and classification methods (the softmax loss, CosFace [52], and the circle loss [46]).

Table 1. Experimental results (%) of the proposed CLCD method compared with existing methods on the DyML task.

	DyML-Vehicle				DyML-Animal				DyML-Product			
	mAP	R@1	R@10	R@20	mAP	R@1	R@10	R@20	mAP	R@1	R@10	R@20
Triplet	10.0	13.8	52.6	65.1	11.0	18.2	55.5	66.3	9.3	11.2	43.6	53.3
Multi-Sim	10.4	17.4	56.0	67.9	11.6	16.7	53.5	64.8	10.0	12.7	45.7	56.4
N-Pair	10.5	16.4	55.7	68.1	30.3	39.6	69.6	78.8	15.3	20.3	55.5	65.6
Softmax	12.0	22.9	61.6	72.9	25.8	49.6	81.7	88.8	26.1	50.2	81.6	87.7
Cosface	12.0	22.9	62.1	73.4	28.4	45.1	75.7	83.3	25.0	49.3	81.3	87.7
Circle	12.1	23.5	62.0	73.3	30.6	41.5	72.2	80.3	15.0	26.7	61.5	70.3
CSL	12.1	25.2	64.2	75.0	31.0	52.3	81.7	88.3	28.7	54.3	83.1	89.4
CLCD-ACR	16.0	42.9	74.0	84.1	**36.0**	**57.1**	**85.2**	**90.1**	29.4	58.8	86.2	90.7
CLCD-ICR	**16.6**	**43.7**	**75.4**	**86.3**	35.7	56.0	84.8	89.7	**30.2**	**59.5**	**87.1**	**92.1**

Table 2. Experimental results using pretrained weights on the DyML-Animal dataset.

Methods	mAP	R@1	R@10	R@20
CLCD-ACR pretrained	55.1	83.0	96.8	98.6
CLCD-ICR pretrained	**55.4**	**83.3**	**97.0**	**98.7**

We see that the proposed method achieves the best performance on all three DyML datasets without negative repulsion. This is because our CLCD only imposes positive attraction on the concepts from different semantic levels, which avoids the cross-level conflicts and is able to adaptively learn the concept scope at each semantic level. Also, we observe that the ICR strategy for concept distillation attains better performance on the DyML-Vehicle and DyML-Product datasets but lower performance on the DyML-Animal dataset.

4.5 Experimental Analysis

Analysis of ACR & ICR: We first studied why ICR performs worse than ACR on DyML-Animal but better on DyML-Vehicle and DyML-Product. The hypothesized factor is whether to use pre-trained weights, as we followed the benchmark setting to use randomly initialized weights on DyML-Animal. We thus conducted an experiment to also use pre-trained weights on DyML-Animal, as shown in Table 2. We see that ICR outperforms ACR in this case, which is the same to the results on the other datasets.

Performance Analysis at Different Semantic Levels: To further analyze the effectiveness of our method, we present the results of the proposed CLCD-ICR on each semantic level compared with CosFace [52] and CSL [47], as shown in Table 3. We see that despite achieving better overall performance, our method does not perform the best on all the semantic levels. Specifically, the CSL method outperforms our method at the fine level on the DyML-Product dataset, while our method achieves higher results on the middle and coarse levels. We think this is because the absence of the negative repulsion in our method

Table 3. Experimental results (%) at all the semantic levels of the proposed CLCD method compared with existing methods.

Method	Level	DyML-Vehicle		DyML-Animal		DyML-Product	
		mAP	R@1	mAP	R@1	mAP	R@1
Cosface	Fine	-	-	8.7	18.3	11.1	20.3
	Middle	-	-	28.4	46.6	16.9	47.6
	Coarse	-	-	48.2	70.5	47.1	80.0
	Overall	-	-	28.4	45.1	25.0	49.3
CSL	Fine	-	-	10.3	25.3	15.6	26.2
	Middle	-	-	30.1	53.9	20.1	53.2
	Coarse	-	-	52.7	77.7	50.4	83.7
	Overall	-	-	31.0	52.3	28.7	54.3
CLCD	Fine	3.8	12.6	13.8	28.9	13.9	29.4
	Middle	10.5	30.7	35.6	59.0	22.4	59.2
	Coarse	35.6	75.3	57.7	80.1	54.2	89.8
	Overall	16.6	43.7	35.7	56.0	30.2	59.5

Table 4. Comparisons of whether to use negative repulsion on DyML-Product.

Method	Fine level		Middle level		Coarse level		Overall	
	mAP	R@1	mAP	R@1	mAP	R@1	mAP	R@1
CLCD w/ NP	**16.2**	**27.5**	19.7	51.8	52.1	86.9	29.3	55.4
CLCD	13.9	29.4	**22.4**	**59.2**	**54.2**	**89.8**	**30.2**	**59.5**

compromises the discriminativeness of the image embedding space for a more flexible scope of each concept. To validate this hypothesis, we add the negative repulsion only on the fine level, as shown in Table 4. We see that negative repulsion helps on the fine level but reduces the performance on the other levels. Therefore, the adaptively learned concept scopes are more important on higher semantic levels which are more probable to contain different numbers of sub-concepts. The use of fixed hand-crated margins in CSL enforces each concept to occupy the same area of region in the embedding space regardless of the concept scope, which may damage the generalization ability of the learned metric.

Conventional Metric Learning without Negative Repulsion: To demonstrate the effectiveness of only using a positive attraction loss to learn the embedding space, we applied our method to the conventional metric learning setting on the CUB-200-2011 [50] dataset, where only one level of concept is present in the data. We simplified the proposed CLCD method to a vanilla version (CLCD-V), where only one encoder and decoder are used to refine a single concept embedding. We then simply use the distance between an image embedding with its positive concept embedding as the loss function to train the model.

For fair comparisons with existing deep metric learning losses, we adopted the recent proposed experimental settings [35] including using the ImageNet-1K [40] pretrained BN-Inception [25], smaller batch size, and strict dataset split. See Musgrave *et al.* [35] for more details. We strictly followed these protocols

Table 5. Experimental results (%) of for conventional deep metric learning.

	Concatenated (512-dim)			Separated (128-dim)		
	P@1	**RP**	**MAP@R**	**P@1**	**RP**	**MAP@R**
Pretrained	51.05	24.85	14.21	50.54	25.12	14.53
Contrastive[20]	**68.13 ± 0.31**	37.24 ± 0.28	26.53 ± 0.29	59.73 ± 0.40	31.98 ± 0.29	21.18 ± 0.28
Triplet [56]	64.24 ± 0.26	34.55 ± 0.24	23.69 ± 0.23	55.76 ± 0.27	29.55 ± 0.16	18.75 ± 0.15
ProxyNCA [34]	65.69 ± 0.43	35.14 ± 0.26	24.21 ± 0.27	57.88 ± 0.30	30.16 ± 0.22	19.32 ± 0.21
Margin [57]	64.37 ± 0.18	34.59 ± 0.16	23.71 ± 0.16	55.56 ± 0.16	29.32 ± 0.15	18.51 ± 0.13
N. Softmax [63]	65.65 ± 0.30	35.99 ± 0.15	25.25 ± 0.13	58.75 ± 0.19	31.75 ± 0.12	20.96 ± 0.11
CosFace [52]	67.32 ± 0.32	**37.49 ± 0.21**	**26.70 ± 0.23**	59.63 ± 0.36	31.99 ± 0.22	21.21 ± 0.22
ArcFace [9]	67.50 ± 0.25	37.31 ± 0.21	26.45 ± 0.20	**60.17 ± 0.32**	**32.37 ± 0.17**	**21.49 ± 0.16**
FastAP [2]	63.17 ± 0.34	34.20 ± 0.20	23.53 ± 0.20	55.58 ± 0.31	29.72 ± 0.16	19.09 ± 0.16
SNR [61]	66.44 ± 0.56	36.56 ± 0.34	25.75 ± 0.36	58.06 ± 0.39	31.21 ± 0.28	20.43 ± 0.28
MS [54]	65.04 ± 0.28	35.40 ± 0.12	24.70 ± 0.13	57.60 ± 0.24	30.84 ± 0.13	20.15 ± 0.14
MS+Miner [54]	67.73 ± 0.18	37.37 ± 0.19	26.52 ± 0.18	59.41 ± 0.30	31.93 ± 0.15	21.01 ± 0.14
SoftTriple [39]	67.27 ± 0.39	37.34 ± 0.19	26.51 ± 0.20	59.94 ± 0.33	32.12 ± 0.14	21.31 ± 0.14
CLCD-V	67.13 ± 0.24	37.17 ± 0.17	26.49 ± 0.25	59.97 ± 0.24	31.33 ± 0.11	21.26 ± 0.13

by implementing our method using the provided code [3]. Table 5 shows the performance of various loss functions. We observe that using a simple positive attraction loss achieves comparable performance with the other losses, which all impose both positive attraction and negative repulsion on the image embeddings. The results demonstrate that the proposed CLCD method is able to learn a discriminative embedding space despite the absence of negative repulsion. Our method implicitly pushes away negative pairs in an adaptive manner free from hand-crafted margins, which we found affect the performance of the contrastive loss, the triplet loss, the margin loss largely.

Ablation Study: We conduct an ablation study to analyze the effectiveness of each component in the proposed CLCD method on the DyML-Vehicle dataset, as shown in Table 6. Asymmetry denotes we only pull closer the concept embedding of one image to the image embedding of another positive sample but not always the other way around. Random sampling means that we randomly select images from the dataset to construct a mini-batch. Intra-level represents using (9) as the loss function to pull closer positive concept embeddings at the same semantic level. W/o stop-gradient means we do not use the stop-gradient operation in our method.

We see that Asymmetry attains slightly lower performance resulting from the possible inaccurate estimates of the backward gradient due to the lack of comparisons. Random sampling also leads to compromised performance and much lower convergence speed, since we can only find very few positive pairs in each mini-batch especially for the finest level, due to the vast number of classes. Using the intra-level positive pulling loss also achieves poor performance as each concept cannot see the relevant concepts from the lower levels and thus is not able to reflect their common grounds. The absence of the stop-gradient operation leads to model collapse. To further understand how the stop-gradient operation

[3] https://github.com/KevinMusgrave/pytorch-metric-learning

Table 6. Ablation study of different settings on DyML-Vehicle.

Setting	mAP	R@1	R@10	R@20
Asymmetry	14.9	40.1	72.3	83.2
Random sampling	10.2	30.6	62.6	78.8
Intra-level	12.4	34.3	67.9	80.0
W/o stop-gradient	1.3	10.2	23.6	73.3
CLCD	**16.6**	**43.7**	**75.4**	**86.3**

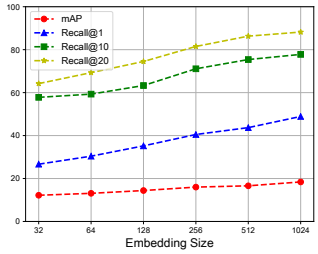

Fig. 4. Effect of different embedding sizes.

works, we conducted an experiment where we initialized all the embeddings to a fixed point so that the targets are the same for different concepts. We observe that the training collapses even with the stop-gradient operation. This verifies the significance of using different targets and further backs up the hypothesis [6].

Effect of Embedding Dimension: We conduct an experiment on the DyML-Vehicle dataset with different dimensions of the image embedding size, as shown in Fig. 4. The dimension of each meta-concept embedding is proportionally resized according to that of the image embedding. We see that using a larger embedding dimension generally improves the performance across all the semantic levels due to the better representation ability. Note that the output feature after the pooling layer of the ResNet-34 model used in the experiments had a dimension of 512, but uplifting it into a 1024-dimension embedding as the image representation still improves the performance.

5 Conclusion

In this paper, we have presented a cross-level concept distillation method for dynamic metric learning. We employ a hierarchical concept refiner to obtain a series of concept embeddings for an image and distill higher-level concepts using lower-level concepts. We only impose constraints on the cross-level positive concept pairs to avoid the possible conflicts across semantic levels. We have evaluated our method under the dynamic metric learning setting which shows that the proposed CLCD outperforms all other existing methods. We also conducted experiments under the conventional deep metric learning setting to further verify the effectiveness of only pulling closer positive pairs. In the future, it is interesting to apply our method to semi-supervised learning, where we can regard the instance-level and class-level labels as concepts from different semantic levels.

Acknowledgements

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 62125603 and Grant U1813218, in part by a grant from the Beijing Academy of Artificial Intelligence (BAAI).

References

1. Bellman, R.: Dynamic programming. Science **153**(3731), 34–37 (1966) 9
2. Cakir, F., He, K., Xia, X., Kulis, B., Sclaroff, S.: Deep metric learning to rank. In: CVPR. pp. 1861–1870 (2019) 3, 13
3. Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., Joulin, A.: Unsupervised learning of visual features by contrasting cluster assignments. In: NeurIPS (2020) 8
4. Caruana, R.: Multitask learning. Machine learning **28**(1), 41–75 (1997) 4
5. Chen, W., Chen, X., Zhang, J., Huang, K.: Beyond triplet loss: a deep quadruplet network for person re-identification. In: CVPR. pp. 1320–329 (2017) 1
6. Chen, X., He, K.: Exploring simple siamese representation learning. In: CVPR. pp. 15750–15758 (2021) 8, 9, 14
7. Chu, X., Tian, Z., Wang, Y., Zhang, B., Ren, H., Wei, X., Xia, H., Shen, C.: Twins: Revisiting the design of spatial attention in vision transformers (2021) 1
8. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR. pp. 248–255 (2009) 4
9. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: Arcface: Additive angular margin loss for deep face recognition. In: CVPR. pp. 4690–4699 (2019) 3, 13
10. Dhall, A., Makarova, A., Ganea, O., Pavllo, D., Greeff, M., Krause, A.: Hierarchical image classification using entailment cone embeddings. In: CVPRW. pp. 836–837 (2020) 4
11. Do, T.T., Tran, T., Reid, I., Kumar, V., Hoang, T., Carneiro, G.: A theoretically sound upper bound on the triplet loss for improving the efficiency of deep distance metric learning. In: CVPR. pp. 10404–10413 (2019) 3
12. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2020) 1
13. Duan, Y., Zheng, W., Lin, X., Lu, J., Zhou, J.: Deep adversarial metric learning. In: CVPR. pp. 2780–2789 (2018) 1, 3
14. Dutt, A., Pellerin, D., Quénot, G.: Improving hierarchical image classification with merged cnn architectures. In: Proceedings of the 15th International Workshop on Content-Based Multimedia Indexing. pp. 1–7 (2017) 4
15. Elezi, I., Vascon, S., Torcinovich, A., Pelillo, M., Leal-Taixe, L.: The group loss for deep metric learning. In: ECCV (2019) 3
16. Ge, W., Huang, W., Dong, D., Scott, M.R.: Deep metric learning with hierarchical triplet loss. In: ECCV. pp. 269–285 (2018) 3
17. Ghosh, S., Singh, R., Vatsa, M.: On learning density aware embeddings. In: CVPR. pp. 4884–4892 (2019) 3
18. Grill, J.B., Strub, F., Altché, F., Tallec, C., Richemond, P.H., Buchatskaya, E., Doersch, C., Pires, B.A., Guo, Z.D., Azar, M.G., et al.: Bootstrap your own latent: A new approach to self-supervised learning. arXiv **abs/2006.07733** (2020) 8
19. Guo, Y., Liu, Y., Bakker, E.M., Guo, Y., Lew, M.S.: Cnn-rnn: a large-scale hierarchical image classification framework. Multimedia tools and applications **77**(8), 10251–10271 (2018) 4
20. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: CVPR. pp. 1735–1742 (2006) 13
21. Harwood, B., Kumar B G, V., Carneiro, G., Reid, I., Drummond, T.: Smart mining for deep metric learning. In: ICCV. pp. 2840–2848 (2017) 3

22. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016) 1
23. Hu, J., Lu, J., Tan, Y.P.: Discriminative deep metric learning for face verification in the wild. In: CVPR. pp. 1875–1882 (2014) 1
24. Huang, C., Loy, C.C., Tang, X.: Local similarity-aware deep feature embedding. In: NeurIPS. pp. 1262–1270 (2016) 3
25. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: ICML. pp. 448–456 (2015) 12
26. Khrulkov, V., Mirvakhabova, L., Ustinova, E., Oseledets, I., Lempitsky, V.: Hyperbolic image embeddings. In: CVPR. pp. 6418–6428 (2020) 3
27. Kim, S., Kim, D., Cho, M., Kwak, S.: Proxy anchor loss for deep metric learning. In: CVPR. pp. 3238–3247 (2020) 3
28. Ko, B., Gu, G.: Embedding expansion: Augmentation in embedding space for deep metric learning. In: CVPR. pp. 7255–7264 (2020) 3
29. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009) 4
30. Law, M.T., Urtasun, R., Zemel, R.S.: Deep spectral clustering learning. In: ICML. pp. 1985–1994 (2017) 1
31. Lin, X., Duan, Y., Dong, Q., Lu, J., Zhou, J.: Deep variational metric learning. In: ECCV. pp. 689–704 (2018) 3
32. Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B., Song, L.: Sphereface: Deep hypersphere embedding for face recognition. In: CVPR. pp. 6738–6746 (2017) 3
33. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows (2021) 1
34. Movshovitz-Attias, Y., Toshev, A., Leung, T.K., Ioffe, S., Singh, S.: No fuss distance metric learning using proxies. In: ICCV. pp. 360–368 (2017) 3, 13
35. Musgrave, K., Belongie, S., Lim, S.N.: A metric learning reality check. In: ECCV (2020) 12
36. Nickel, M., Kiela, D.: Poincaré embeddings for learning hierarchical representations. NeurIPS **30** (2017) 3
37. Nickel, M., Kiela, D.: Learning continuous hierarchies in the lorentz model of hyperbolic geometry. In: ICML. pp. 3779–3788 (2018) 3
38. Opitz, M., Waltner, G., Possegger, H., Bischof, H.: Deep metric learning with bier: Boosting independent embeddings robustly. TPAMI (2018) 1
39. Qian, Q., Shang, L., Sun, B., Hu, J.: Softtriple loss: Deep metric learning without triplet sampling. In: ICCV (2019) 3, 13
40. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. IJCV **115**(3), 211–252 (2015) 10, 12
41. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: CVPR. pp. 815–823 (2015) 1, 3, 10
42. Shi, H., Yang, Y., Zhu, X., Liao, S., Lei, Z., Zheng, W., Li, S.Z.: Embedding deep metric for person re-identification: A study against large variations. In: ECCV. pp. 732–748 (2016) 1
43. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv **abs/1409.1556** (2014) 1
44. Sohn, K.: Improved deep metric learning with multi-class n-pair loss objective. In: NeurIPS. pp. 1857–1865 (2016) 3, 10
45. Song, H.O., Xiang, Y., Jegelka, S., Savarese, S.: Deep metric learning via lifted structured feature embedding. In: CVPR. pp. 4004–4012 (2016) 1, 3

46. Sun, Y., Cheng, C., Zhang, Y., Zhang, C., Zheng, L., Wang, Z., Wei, Y.: Circle loss: A unified perspective of pair similarity optimization. In: CVPR. pp. 6398–6407 (2020) 10
47. Sun, Y., Zhu, Y., Zhang, Y., Zheng, P., Qiu, X., Zhang, C., Wei, Y.: Dynamic metric learning: Towards a scalable metric space to accommodate multiple semantic scales. In: CVPR. pp. 5393–5402 (2021) 2, 3, 4, 5, 9, 10, 11
48. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S.E., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: CVPR. pp. 1–9 (2015) 1
49. Verma, N., Mahajan, D., Sellamanickam, S., Nair, V.: Learning hierarchical similarity metrics. In: CVPR. pp. 2280–2287 (2012) 4
50. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.J.: The Caltech-UCSD Birds-200-2011 dataset. Tech. Rep. CNS-TR-2011-001, California Institute of Technology (2011) 3, 12
51. Wang, F., Zuo, W., Lin, L., Zhang, D., Zhang, L.: Joint learning of single-image and cross-image representations for person re-identification. In: CVPR. pp. 1288–1296 (2016) 1
52. Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., Li, Z., Liu, W.: Cosface: Large margin cosine loss for deep face recognition. In: CVPR. pp. 5265–5274 (2018) 3, 10, 11, 13
53. Wang, J., Zhou, F., Wen, S., Liu, X., Lin, Y.: Deep metric learning with angular loss. In: ICCV. pp. 2593–2601 (2017) 3
54. Wang, X., Han, X., Huang, W., Dong, D., Scott, M.R.: Multi-similarity loss with general pair weighting for deep metric learning. In: CVPR. pp. 5022–5030 (2019) 3, 10, 13
55. Wang, Y., Hu, B.G.: Hierarchical image classification using support vector machines. In: ACCV. pp. 23–25 (2002) 4
56. Weinberger, K.Q., Saul, L.K.: Distance metric learning for large margin nearest neighbor classification. JMLR **10**(2), 207–244 (2009) 13
57. Wu, C.Y., Manmatha, R., Smola, A.J., Krähenbühl, P.: Sampling matters in deep embedding learning. In: ICCV. pp. 2859–2867 (2017) 3, 13
58. Yan, Z., Zhang, H., Piramuthu, R., Jagadeesh, V., DeCoste, D., Di, W., Yu, Y.: Hd-cnn: hierarchical deep convolutional neural networks for large scale visual recognition. In: ICCV. pp. 2740–2748 (2015) 4
59. Yu, B., Tao, D.: Deep metric learning with tuplet margin loss. In: ICCV. pp. 6490–6499 (2019) 3
60. Yu, R., Dou, Z., Bai, S., Zhang, Z., Xu, Y., Bai, X.: Hard-aware point-to-set deep metric for person re-identification. In: ECCV. pp. 188–204 (2018) 3
61. Yuan, T., Deng, W., Tang, J., Tang, Y., Chen, B.: Signal-to-noise ratio: A robust distance metric for deep metric learning. In: CVPR. pp. 4815–4824 (2019) 13
62. Yuan, Y., Yang, K., Zhang, C.: Hard-aware deeply cascaded embedding. In: ICCV. pp. 814–823 (2017) 3
63. Zhai, A., Wu, H.Y.: Classification is a strong baseline for deep metric learning. arXiv **abs/1811.12649** (2018) 13
64. Zhao, Y., Jin, Z., Qi, G.j., Lu, H., Hua, X.s.: An adversarial approach to hard triplet generation. In: ECCV. pp. 501–517 (2018) 3
65. Zheng, W., Chen, Z., Lu, J., Zhou, J.: Hardness-aware deep metric learning. In: CVPR. pp. 72–81 (2019) 3
66. Zhou, J., Yu, P., Tang, W., Wu, Y.: Efficient online local metric adaptation via negative samples for person re-identification. In: ICCV. pp. 2420–2428 (2017) 1