Multi-level Distance Regularization for Deep Metric Learning

Yonghyun Kim[1*], **Wonpyo Park**[2*]
[1]AI Lab, Kakao Enterprise
[2]Kakao Corp.
aiden.d@kakaoenterprise.com, tony.nn@kakaocorp.com

Abstract

We propose a novel distance-based regularization method for deep metric learning called Multi-level Distance Regularization (MDR). MDR explicitly disturbs a learning procedure by regularizing pairwise distances between embedding vectors into multiple levels that represents a degree of similarity between a pair. In the training stage, the model is trained with both MDR and an existing loss function of deep metric learning, simultaneously; the two losses interfere with the objective of each other, and it makes the learning process difficult. Moreover, MDR prevents some examples from being ignored or overly influenced in the learning process. These allow the parameters of the embedding network to be settle on a local optima with better generalization. Without bells and whistles, MDR with simple Triplet loss achieves the-state-of-the-art performance in various benchmark datasets: CUB-200-2011, Cars-196, Stanford Online Products, and In-Shop Clothes Retrieval. We extensively perform ablation studies on its behaviors to show the effectiveness of MDR. By easily adopting our MDR, the previous approaches can be improved in performance and generalization ability.

Introduction

Deep Metric Learning (DML) aims to learn an appropriate metric that measures the semantic difference between a pair of images as a distance between embedding vectors. Many research areas such as image retrieval (Sohn 2016; Yuan, Yang, and Zhang 2017; Oh Song et al. 2017; Duan et al. 2018; Ge 2018) and face recognition (Wang et al. 2017; Liu et al. 2017; Wang et al. 2018; Deng et al. 2019) are based on DML to seek appropriate metrics among instances. Those studies focus on devising a better loss function for DML.

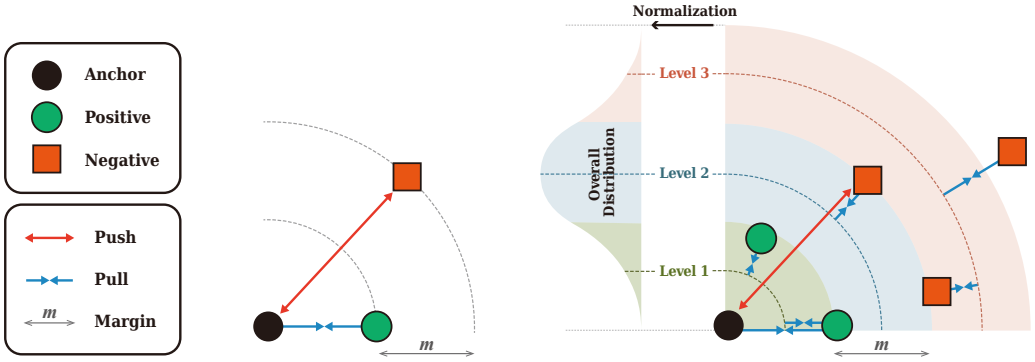
Most of previous loss functions (Sohn 2016; Bromley et al. 1994; Hadsell, Chopra, and LeCun 2006; Yi et al. 2014; Hoffer and Ailon 2015; Schroff, Kalenichenko, and Philbin 2015) use binary supervision that indicates whether a given pair is positive or negative. Their common objective is to minimize the distance between a positive pair and maximize the distance between a negative pair (Figure 1a). However, without any constraints, a model trained with such objective is prone to overfitting on a training set because positive pairs can be aligned too closely while the negative pairs can be aligned too far in the embedding space. Therefore, several loss functions employ additional terms to avoid positive pairs to be too close and negative pairs to be too far, *e.g.*, margin m in Triplet loss (Schroff, Kalenichenko, and Philbin 2015) and Constrastive loss (Hadsell, Chopra, and LeCun 2006). Despite these attempts, they still can suffer from overfitting due to the lack of explicit regularization for the distances.

Our insight is that a learning procedure of DML can be enhanced by explicitly regularizing the distance between pairs to disturb a loss function of DML from optimizing an embedding network; one easy way to constrain a distance is to pull the value of the distance to a predefined level. Conventional loss functions of DML adjust the distance according to its label, on the other hand, explicit distance-based regularization prevents the distance from deviating from the predefined level. Those two interfere with the objective of each other, thus it makes the learning process difficult and allows the embedding network to be more robust for generalization. Additionally, we consider multiple levels with disjoint intervals to regularize distances, not a single level, because a degree of inter-class similarity or intra-class variation can be different depending on classes or instances.

We propose a novel method called Multi-level Distance Regularization (MDR) that makes the conventional loss functions of DML have difficulty in converging by holding each distance so that it does not deviate from the belonging level. At first, MDR normalizes pairwise distances among the embedding vectors of a mini-batch, with their mean and standard deviation to obtain the objective degree of similarity between a pair by considering overall distribution. MDR defines the multiple levels that represent various degrees of similarity for pairwise distances, and the levels and the belonging distances are trained to approach each other (Figure 1b). A conventional loss function of DML struggles to optimize a model by overcoming the disturbance from the proposed regularization. Therefore, the learning process succeeds in learning a model with a better generalization ability. We summarize our contributions:

- We introduce MDR, a novel regularization method for DML. The method disturbs optimizing pairwise distances by preventing them from deviating from its belonging level for better generalization.

[*]Equal Contribution



(a) Conventional Learning (b) Our Learning

Figure 1: Conceptual comparison between the conventional learning scheme and our learning scheme. (a) illustrates the triplet learning (Schroff, Kalenichenko, and Philbin 2015), which is one of the representative conventional learning. It increases the relative difference between distances of a positive pair and that of a negative pair more than margin m. (b) illustrates our learning combined with the triplet learning. It has multiple levels with disjoint intervals to reflect various degrees of similarity between pairs. It disturbs the learning procedure to construct an efficient embedding space by preventing the pairwise distances from deviating from its belonging level.

- MDR achieves the-state-of-the-art performance on various benchmark datasets (Wah et al. 2011; Krause et al. 2013; Oh Song et al. 2016; Liu et al. 2016) of DML. Moreover, our extensive ablation studies show that MDR can be adopted to any backbone networks and any distance-based loss functions to improve the performance of a model.

Related Work

Loss Function. Improving the loss function is one of the key objectives in recent DML studies. One family of loss functions (Sohn 2016; Bromley et al. 1994; Schroff, Kalenichenko, and Philbin 2015; Oh Song et al. 2016; Wang et al. 2019a; Wu et al. 2017) focuses on optimizing pairwise distance between instances. The common objective of these functions is to minimize the distance between positive pairs and to maximize the distance between negative pairs in an embedding space. Contrastive loss (Bromley et al. 1994) samples pairs of two instances, whereas Triplet loss (Schroff, Kalenichenko, and Philbin 2015) samples triplets of anchor, positive and negative instances; then both losses optimize the distance between the sampled instances. Also, Global Loss (Kumar BG, Carneiro, and Reid 2016) minimizes the mean and variance of all pairwise distances between positive examples and maximizes the mean of pairwise distances between all negative examples; Global Loss helps to optimize examples that are not selected by the example mining of DML. Histogram Loss (Ustinova and Lempitsky 2016) minimizes the probability that a randomly sampled positive pair has a smaller similarity than randomly sampled negative pairs. To extend the number of relations explored at once, NPair (Sohn 2016) samples a positive and all negative instances for each example in a given mini-batch; similar loss functions (Oh Song et al. 2016; Wang et al. 2019a) also sample a large number of instances to fully explore the pairwise relations in the mini-batch. On the other, some loss functions (Cakir et al. 2019; Revaud et al. 2019) focus on learning to rank according to the similarity between pairs. The performance of loss functions optimizing pairwise distance can be changed by a sampling method, thus, several studies focused on the pair sampling (Suh et al. 2019; Schroff, Kalenichenko, and Philbin 2015; Wu et al. 2017) for stable learning and better accuracy. A recent work (Wang et al. 2020) even samples pairs across mini-batches to collect a sufficient number of negative examples. Instead of designing a sampling method manually, a work (Roth, Milbich, and Ommer 2020) employs reinforcement learning to learn the policy for sampling. As a regularizer, MDR can be combined with those loss functions to improve the generalization ability of a model.

Generalization Ability. Another goal of DML is to improve the generalization ability of a given model. An ensemble of multiple heads that share the backbone network (Opitz et al. 2018; Kim et al. 2018; Jacob et al. 2019; Sanakoyeu et al. 2019) has the key objective of diversifying each head to achieve reliable embedding. Boosting can be used to reweight the importance of instances differently on each head (Opitz et al. 2018; Sanakoyeu et al. 2019), or a spatial attention module can be used to differentiate a spatial region on which each head focuses (Kim et al. 2018). HORDE (Jacob et al. 2019) makes each head approximate a different higher-order moment. Those methods focus on changing the architecture of a model, but our MDR, as a regularizer, focuses on making a learning procedure harder to improve generalization ability. Without adding any extra computational costs or changing the architecture of the model, MDR can be easily integrated with those DML methods by simply adding our loss function.

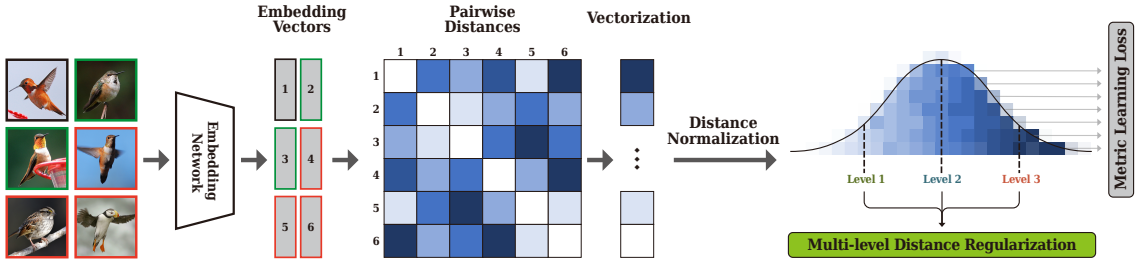


Figure 2: Learning procedure of the proposed MDR. The embedding network generates embedding vectors from given images. Our MDR computes a matrix of pairwise distances for the embedding vectors, and then, the distances are normalized after vectorization. In our learning scheme, a model is trained by simultaneously optimizing the conventional metric learning loss such as Triplet loss (Schroff, Kalenichenko, and Philbin 2015) and the proposed loss, which regularizes the normalized pairwise distances with multiple levels.

Proposed Method

In this section, we introduce a new regularization method called Multi-level Distance Regularization (MDR), which makes the learning procedure difficult by preventing each pairwise distance from deviating from a corresponding level, to learn a robust feature representation.

Multi-level Distance Regularization

We describe the detailed procedure of MDR to regulate pairwise distances in three steps (Figure 2).

(1) Distance Normalization. This step is performed to obtain an objective degree of distance by considering overall distribution for stable regularization. Here, an embedding network f maps an image x into an embedding vector e with a certain dimensionality: $e = f(x)$. A distance is defined as Euclidean distance between two given embedding vectors, $d(e_i, e_j) = \|e_i - e_j\|_2$. We normalize the distance as:

$$\bar{d}(e_i, e_j) = \frac{d(e_i, e_j) - \mu}{\sigma}, \tag{1}$$

where μ is mean of distances and σ is standard deviation of distances a set of pairs, which is $\mathcal{P} = \{(e_i, e_j) | i \neq j\}$ for all instances of a mini-batch. To more widely consider the overall dataset, we employ the momentum updates:

$$\begin{aligned} \mu_t^* &= \gamma \mu_{t-1}^* + (1-\gamma)\mu, \\ \sigma_t^* &= \gamma \sigma_{t-1}^* + (1-\gamma)\sigma, \end{aligned} \tag{2}$$

where μ_t^* and σ_t^* are respectively the momented mean and momented standard deviation at iteration t, and γ is the momentum. With the momented statistics, the normalized distance is re-written:

$$\bar{d}(e_i, e_j) = \frac{d(e_i, e_j) - \mu^*}{\sigma^*}. \tag{3}$$

(2) Level Assignment. MDR designates a level that acts as a regularization goal for each normalized distance. We define a set of levels $s \in \mathcal{S}$, and the levels are initialized with predefined values; each level s is interpreted as a multiplier of the standard deviation of the normalized distance. $g(d; s)$ is an assignment function that outputs whether the given distance d and the given level s are the closest or not, and is defined as:

$$g(d, s) = \begin{cases} 1, & \text{if } \arg\min_{s_i \in \mathcal{S}} |d - s_i| \text{ is } s \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

By adopting the assignment function, MDR selects valid regularization levels for each distance with the consideration of various degrees of similarities.

(3) Regularization. Finally, this step is performed to prevent pairwise distances from deviating from its belonging level. MDR minimizes the difference between a given normalized pairwise distance and the assigned level:

$$\mathcal{L}_{\text{MDR}} = \frac{1}{\mathcal{P}} \sum_{(e_i, e_j) \in \mathcal{P}} \sum_{s \in \mathcal{S}} g\left(\bar{d}(e_i, e_j), s\right) \cdot \left|\bar{d}(e_i, e_j) - s\right|. \tag{5}$$

The levels are learnable parameters and are updated to optimally regularize the pairwise distances. Each normalized distance is trained to become closer to the assigned level; the assigned level is also trained to become closer to the corresponding distances. As iterations pass, the levels are trained to properly divide the normalized distances into multiple intervals. Each level is a representative value of a certain interval in the normalized distance. We describe the initial configuration of the levels in Section .

In conclusion, MDR has two functional effects of regularization: (1) the multiple levels of MDR disturbs optimizing the pairwise distances among examples, (2) the outermost levels of MDR prevents the positive pairs from getting too close and the negative pairs from getting too far. By the formal effect, the learning process does not easily suffer from overfitting. By the latter effect, the learning process does not suffer from diminishing of the loss from easy examples, and also, does not suffer from being too biased to certain examples such as hard examples. Therefore, MDR stabilizes the learning procedure to achieve a better generalization ability on a test dataset.

Learning

Loss Function. The proposed MDR can be applied to any loss functions $\mathcal{L}_{\text{DML}}$ such as Contrastive loss (Bromley

	CUB-200				Cars-196			
Recall@K	1	2	4	8	1	2	4	8
HTL (Ge 2018)	57.1	68.8	78.7	86.5	81.4	88.0	92.7	95.7
RLL-H (Wang et al. 2019b)	57.4	69.7	79.2	86.9	74.0	83.6	90.1	94.1
NSM (Zhai and Wu 2019)	59.6	72.0	81.2	88.4	81.7	88.9	93.4	96.0
MS (Wang et al. 2019a)	65.7	77.0	86.3	91.2	84.1	90.4	94.0	96.5
SoftTriple (Qian et al. 2019)	65.4	76.4	84.5	90.4	84.5	90.7	94.5	96.9
HORDE† (Jacob et al. 2019)	66.3	76.7	84.7	90.6	83.9	90.3	94.1	96.3
DiVA (Milbich et al. 2020)	66.8	77.7	-	-	84.1	90.7	-	-
Triplet	$57.3_{\pm 0.7}$	$68.7_{\pm 0.8}$	$78.4_{\pm 0.6}$	$86.1_{\pm 0.4}$	$76.2_{\pm 0.6}$	$84.4_{\pm 0.3}$	$90.0_{\pm 0.2}$	$93.7_{\pm 0.2}$
Triplet+L_2Norm	$65.1_{\pm 0.3}$	$76.1_{\pm 0.2}$	$84.2_{\pm 0.2}$	$90.3_{\pm 0.1}$	$79.8_{\pm 0.3}$	$87.1_{\pm 0.3}$	$91.9_{\pm 0.4}$	$95.1_{\pm 0.1}$
Triplet+MDR	$\mathbf{68.8}_{\pm 0.5}$	$\mathbf{78.8}_{\pm 0.3}$	$\mathbf{86.6}_{\pm 0.2}$	$\mathbf{91.8}_{\pm 0.1}$	$\mathbf{88.5}_{\pm 0.3}$	$\mathbf{93.0}_{\pm 0.2}$	$\mathbf{95.6}_{\pm 0.2}$	$\mathbf{97.5}_{\pm 0.1}$
Triplet+MDR†	$\mathbf{71.4}_{\pm 0.4}$	$\mathbf{81.2}_{\pm 0.3}$	$\mathbf{88.0}_{\pm 0.2}$	$\mathbf{92.6}_{\pm 0.3}$	$\mathbf{90.4}_{\pm 0.2}$	$\mathbf{94.3}_{\pm 0.1}$	$\mathbf{96.6}_{\pm 0.1}$	$\mathbf{98.0}_{\pm 0.1}$

(a) CUB-200 (Wah et al. 2011) and Cars-196 (Krause et al. 2013)

	SOP				In-Shop			
Recall@K	1	10	100	1000	1	10	20	40
NSM (Zhai and Wu 2019)	73.8	88.1	95.0	-	-	-	-	-
MS (Wang et al. 2019a)	78.2	90.5	96.0	98.7	89.7	97.9	98.5	99.1
SoftTriple (Qian et al. 2019)	78.3	90.3	95.9	-	-	-	-	-
HORDE† (Jacob et al. 2019)	80.1	91.3	96.2	98.7	90.4	97.8	98.4	98.9
DiVA (Milbich et al. 2020)	78.1	90.6	-	-	-	-	-	-
Triplet	$75.8_{\pm 0.1}$	$87.9_{\pm 0.1}$	$94.1_{\pm 0.1}$	$97.6_{\pm 0.1}$	$88.2_{\pm 0.1}$	$96.7_{\pm 0.1}$	$97.6_{\pm 0.1}$	$98.3_{\pm 0.1}$
Triplet+L_2Norm	$79.1_{\pm 0.1}$	$90.9_{\pm 0.1}$	$96.3_{\pm 0.1}$	$98.8_{\pm 0.1}$	$90.1_{\pm 0.1}$	$97.8_{\pm 0.1}$	$98.6_{\pm 0.0}$	$99.1_{\pm 0.1}$
Triplet+MDR	$\mathbf{80.1}_{\pm 0.0}$	$\mathbf{91.4}_{\pm 0.1}$	$\mathbf{96.4}_{\pm 0.1}$	$\mathbf{98.8}_{\pm 0.1}$	$\mathbf{90.5}_{\pm 0.1}$	$\mathbf{97.9}_{\pm 0.1}$	$\mathbf{98.5}_{\pm 0.1}$	$\mathbf{99.1}_{\pm 0.1}$
Triplet+MDR†	$\mathbf{80.8}_{\pm 0.1}$	$\mathbf{91.9}_{\pm 0.1}$	$\mathbf{96.7}_{\pm 0.0}$	$\mathbf{98.9}_{\pm 0.0}$	$\mathbf{91.3}_{\pm 0.1}$	$\mathbf{98.2}_{\pm 0.1}$	$\mathbf{98.8}_{\pm 0.0}$	$\mathbf{99.3}_{\pm 0.0}$

(b) SOP (Oh Song et al. 2016) and In-Shop (Liu et al. 2016)

Table 1: Recall@K comparison with state-of-the-art methods. The baseline methods and MDR are grouped in the gray-colored rows. † indicates that the model is trained and tested with large images of 256×256 following the setting of (Jacob et al. 2019). We round reported values to the first decimal place.

et al. 1994), Triplet loss (Schroff, Kalenichenko, and Philbin 2015) and Margin loss (Wu et al. 2017). We mostly adopted Triplet loss as baseline for our experiments:

$$\mathcal{L}_{\text{Triplet}} = \frac{1}{|\mathcal{T}|} \sum_{(e^a, e^p, e^n) \in \mathcal{T}} [d(e^a, e^p) - d(e^a, e^n) + m]_+ , \tag{6}$$

where $\mathcal{T}$ is a set of triplets of an anchor e^a, a positive e^p, and a negative e^n sampled from a mini-batch. m is a margin. The final loss function $\mathcal{L}$ is defined as the sum of $\mathcal{L}_{\text{DML}}$ and $\mathcal{L}_{\text{MDR}}$ with a multiplier λ that balances the losses:

$$\mathcal{L} = \mathcal{L}_{\text{DML}} + \lambda \mathcal{L}_{\text{MDR}}. \tag{7}$$

$\mathcal{L}_{\text{DML}}$ optimizes the model by minimizing the distance of positive pairs and maximizing the distance of negative pairs. $\mathcal{L}_{\text{MDR}}$ regularize the pairwise distances by constraining the distances with multiple levels. The embedding network is trained simultaneously with different objectives.

Embedding Normalization Trick for MDR. In our learning procedure, L_2 Normalization (L_2 Norm) is not adopted because it can disturb the proper regularization effect of MDR. However, the lack of L_2 Norm can cause difficulty in finding appropriate hyper-parameters of $\mathcal{L}_{\text{DML}}$ such as margin m in Triplet loss, because any prior knowledge of the scale of embedding vectors is not given. To overcome the difficulty, we normalize $\mathcal{L}_{\text{DML}}$ by dividing the embedding vectors e by μ during the training stage, such that the expected pairwise distance is one: $\mathbb{E}\left[d(\frac{e_i}{\mu}, \frac{e_j}{\mu})\right] = 1$. We adopt this trick on several loss functions such as Constrastive loss (Hadsell, Chopra, and LeCun 2006), Margin loss (Wu et al. 2017), and Triplet loss in our experiments.

Experiments

To show the effectiveness of MDR and its behaviors, we extensively perform ablation studies and experiments. We follow the standard evaluation protocol and data splits proposed in (Oh Song et al. 2016). For an unbiased evaluation, we conduct 5 independent runs for each experiment and report the mean and the standard deviation of them.

Datasets. We employ the four standard datasets of deep metric learning for evaluations: CUB-200-2011 (Wah et al. 2011) (CUB-200), Cars-196 (Krause et al. 2013), Stanford Online Product (Oh Song et al. 2016) (SOP) and In-Shop Clothes Retrieval (Liu et al. 2016) (In-Shop). CUB-200 has 5,864 images of first 100 classes for training and 5,924 im-

ages of the rest classes for evaluation. Cars-196 has 8,054 images of first 98 classes for training and 8,131 images of the rest classes for evaluation. SOP has 59,551 images of 11,318 classes for training and 60,502 images of the rest classes for evaluation. In-Shop has 25,882 images of 3,997 classes for training, and the remaining 7,970 classes with 26,830 images are partitioned into two subsets (query set and gallery set) for evaluation.

Implementation Details

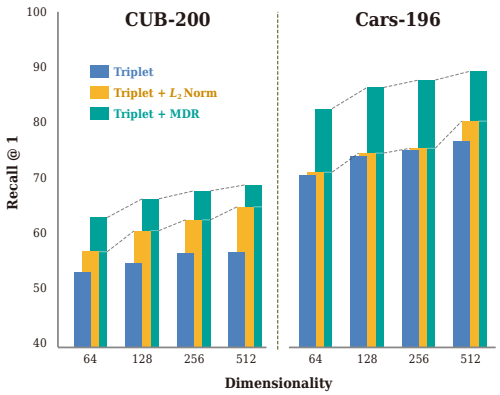
Embedding Network. All the compared methods and our method use the Inception architecture with Batch Normalization (IBN) (Ioffe and Szegedy 2015) as a backbone network. IBN is pre-trained for ImageNet ILSVRC 2012 dataset (Deng et al. 2009) and then fine-tuned on the target dataset. We attach a fully-connected layer, where its output activation is used as an embedding vector, after the last pooling layer of IBN. For models trained with MDR, L_2 Norm is not applied to the embedding vectors because it disturbs the effect of the regularization. For a fair comparison with the conventional implementation of Triplet loss (Schroff, Kalenichenko, and Philbin 2015) that is used as a baseline, we apply L_2 Norm to those models.

Learning. We employ Adam (Kingma and Ba 2014) optimizer with a weight decay of 10^{-5}. For CUB-200 and Cars-196, a learning rate and the size of mini-batch are set to $5 \cdot 10^{-5}$ and 128. For SOP and In-Shop, a learning rate and the size of mini-batch are set to 10^{-4} and 256. We mainly apply our method to Triplet loss (Schroff, Kalenichenko, and Philbin 2015). As a triplet sampling method, we employ the distance weighted sampling (Wu et al. 2017). The margin m of Triplet loss is set to 0.2. We summarized the hyper-parameters of MDR: the configuration of the levels is initialized to three levels of $\{-3, 0, 3\}$, and the momentum γ is set to 0.9. λ is set differently for each dataset: 0.6 for CUB-200, 0.2 for Cars-196 and 0.1 for SOP and In-Shop. For most of the datasets, λ of 0.1 is enough to improve a given model; on CUB-200, a strong regularization is more effective because it is a small dataset with only 5,864 training images where a model may easily suffer from overfitting. Those hyper-parameters are not very sensitive to tune, and we explain the effects of each hyper-parameter in the ablation studies at Section .

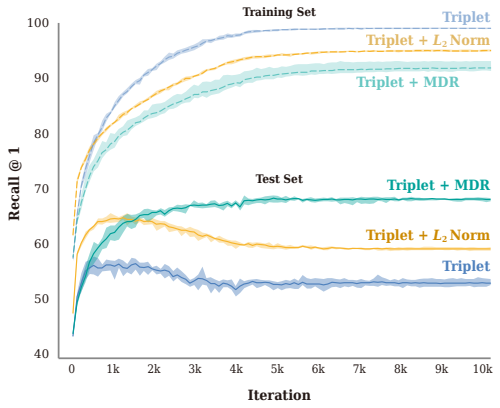
Image Setting. During training, we follow the standard image augmentation process (Oh Song et al. 2016; Wang et al. 2019a) with the following order: resizing to 256×256, random cropping, random horizontal flipping, and resizing to 224×224. For evaluation, images are center-cropped.

Comparison with State-of-the-art Methods

We show the comparison of MDR and the recent state-of-the-art methods (Table 1). All compared methods use embedding vectors of 512 dimensionality. Our baseline model is trained by Triplet loss without L_2 Norm (Triplet) and we also report the conventional Triplet with L_2 Norm (Triplet+L_2 Norm). The lack of constraints of L_2 Norm on the embedding space results in poor generalization performance, and it is known that Triplet loss is effective when L_2 Norm is applied (Schroff, Kalenichenko, and Philbin 2015). However, the models with MDR outperform the Triplet+L_2 Norm models on all the datasets. Those results prove the effectiveness of the proposed distance-based regularization.



(a) Dimensionality



(b) Learning Curves

Figure 3: (a) compares the three methods on various dimensionalities of the embedding vector on CUB-200 and Cars-196. (b) shows the learning curves of the three methods for the training and test set on CUB-200.

Experimental Results. MDR improves performance on all the datasets, and, in particular, the improvements are significantly high on the small-sized datasets. For CUB-200, MDR improves 3.7 percentage points on Recall@1 compared to the conventional Triplet+L_2 Norm; the result is 11.5 percentage points higher than Recall@1 of the Triplet. For Cars-196, MDR improves 8.7 percentage points on Recall@1 compared to the conventional Triplet+L_2 Norm; the result is 12.3 percentage points higher than Recall@1 of the Triplet. MDR also improves the recall performance compared to the baselines on SOP and In-Shop. Moreover, our method significantly outperforms the other state-of-the-art methods in all recall criteria for all datasets.

MDR	CUB-200		Cars-196	
		✓		✓
R18	$51.2_{\pm 0.5}$	$63.6_{\pm 0.5}$	$63.4_{\pm 1.0}$	$82.3_{\pm 0.2}$
R50	$58.5_{\pm 0.4}$	$65.8_{\pm 0.3}$	$77.5_{\pm 0.4}$	$87.6_{\pm 0.2}$
IBN	$57.3_{\pm 0.7}$	$68.8_{\pm 0.5}$	$76.2_{\pm 0.6}$	$88.5_{\pm 0.3}$

(a) Backbone Network

MDR	CUB-200		Cars-196	
		✓		✓
Contrastive	$63.9_{\pm 0.3}$	$65.6_{\pm 0.2}$	$83.2_{\pm 0.1}$	$86.1_{\pm 1.0}$
Margin	$59.3_{\pm 0.5}$	$67.5_{\pm 0.3}$	$79.1_{\pm 0.3}$	$88.2_{\pm 0.4}$
Triplet	$57.3_{\pm 0.7}$	$68.8_{\pm 0.5}$	$76.2_{\pm 0.6}$	$88.5_{\pm 0.3}$

(b) Loss Function

	CUB-200	
	Fixed	Learnable
$\{-1, 0, 1\}$	$64.4_{\pm 0.4}$	$64.9_{\pm 0.4}$
$\{-2, 0, 2\}$	$67.9_{\pm 0.2}$	$68.2_{\pm 0.5}$
$\{-3, 0, 3\}$	$68.2_{\pm 0.1}$	$68.8_{\pm 0.5}$
$\{-3, -1, 0, 1, 3\}$	$64.0_{\pm 0.4}$	$64.9_{\pm 0.1}$
$\{-3, -2, 0, 2, 3\}$	$67.8_{\pm 0.4}$	$67.9_{\pm 0.3}$
$\{-4, -2, 0, 2, 4\}$	$67.8_{\pm 0.3}$	$67.8_{\pm 0.2}$
$\{-6, -3, 0, 3, 6\}$	$68.4_{\pm 0.1}$	$68.7_{\pm 0.5}$

(c) Level Configuration

Table 2: Recall@1 comparison with various backbone networks, loss functions, and level configurations. The models of (a) are trained with Triplet loss. The models of (b) use IBN as the backbone network. In (a) and (b), a column with ✓ indicates that the models are trained with MDR.

Ablation Studies

We extensively perform ablation studies on the behaviors of the proposed MDR.

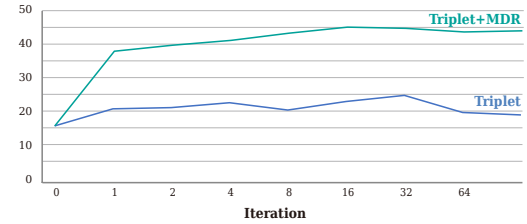
Backbone Network. MDR can be widely applicable to any backbone networks (Table 2a). We apply MDR on IBN (Ioffe and Szegedy 2015), ResNet18 (R18) and ResNet50 (He et al. 2016) (R50), and achieve significant improvements for all backbone networks. Especially, a light-weight backbone, R18, with MDR even outperforms the baseline models with a heavy-weight backbone such as R50 and IBN on both datasets.

Loss Function. Our MDR also can be widely applicable to any distance-based loss function (Table 2b). We apply MDR on Constrastive loss (Hadsell, Chopra, and LeCun 2006), Margin loss (Wu et al. 2017) and Triplet loss. MDR achieves significant improvements for all loss functions.

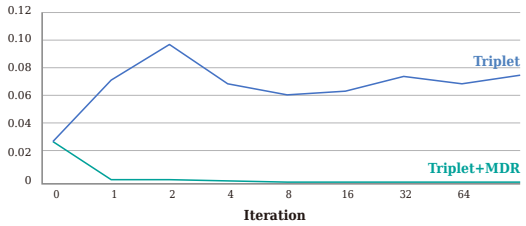
Level Configuration $\mathcal{S}$. Even though the levels are learnable, we should properly set the number of levels and the initial values of levels. We perform experiments on various initial configurations of levels and validate the importance of the learnability of levels (Table 2c). From the experiments, we find that a sufficiently spaced configuration is better than a tightly spaced configuration; $\{-3, 0, 3\}$ is better than $\{-1, 0, 1\}$, and a configuration of three levels is sufficient.

L_2 Norm at Inference	CUB-200	
		✓
Triplet	$57.3_{\pm 0.7}$	$51.5_{\pm 1.0}$
Triplet+MDR	$68.8_{\pm 0.5}$	$68.2_{\pm 0.4}$

Table 3: Recall@1 comparison with the effect of L_2 Norm at inference time for the models trained without L_2 Norm. A column with ✓ indicates that the trained models are evaluated with L_2 Norm.



(a) Expectation of Two-Norm: $E\left[\|e\|_2\right]$



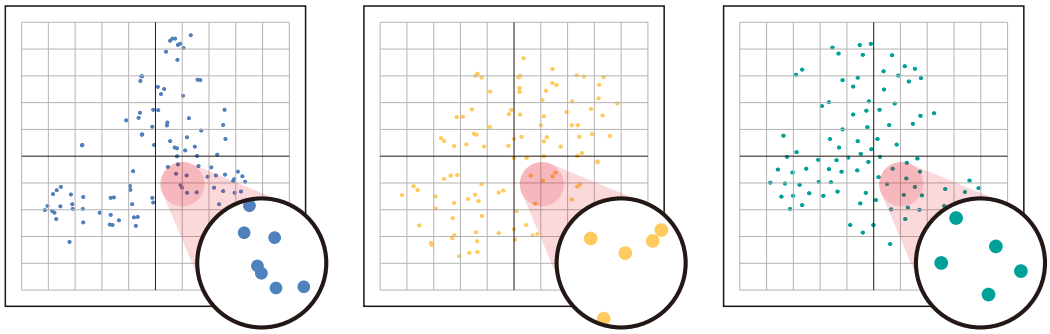
(b) Coefficient of Variation of Two-Norm: $\frac{Var\left[\|e\|_2\right]}{E\left[\|e\|_2\right]}$

Figure 4: (a) compares the expectation of the two-norm of the embedding vectors for the test set on CUB-200. (b) compares the coefficient of variation of the two-norm of the embedding vectors for the test set on CUB-200.

Discussion

Effectiveness in Small Dimensionality. We perform an experiment on various dimensionalities of embedding vector such as 64, 128, 256, and 512. MDR significantly improves the Recall@1 of the models, especially in small dimensionality. In the experiment, our MDR only with 64 dimensionality is similar to or surpasses the performance of other methods with 512 dimensionality (Figure 3a). The result indicates that our MDR constructs a highly efficient embedding space in compact dimensionality. Moreover, the improvements are larger compared to Triplet+L_2 Norm for all dimensionality.

Prevention of Overfitting as Regularizer. We investigate the learning curves of three models: Triplet, Triplet+L_2 Norm and Triplet+MDR (Figure 3b). There are two crucial observations: (1) on the training set, Triplet+MDR is less overfitted than the other two methods, but it shows the most high performance on the test set., (2) the recall of Triplet+MDR does not drop until the end of learning, un-



(a) Triplet (b) Triplet + L_2 Norm (c) Triplet + MDR

Figure 5: Class centers in the embedding space of two models trained without MDR (Triplet & Triplet+L_2 Norm) and one model trained with MDR (Triplet+MDR). We visualize using t-SNE (Maaten and Hinton 2008) on CUB-200.

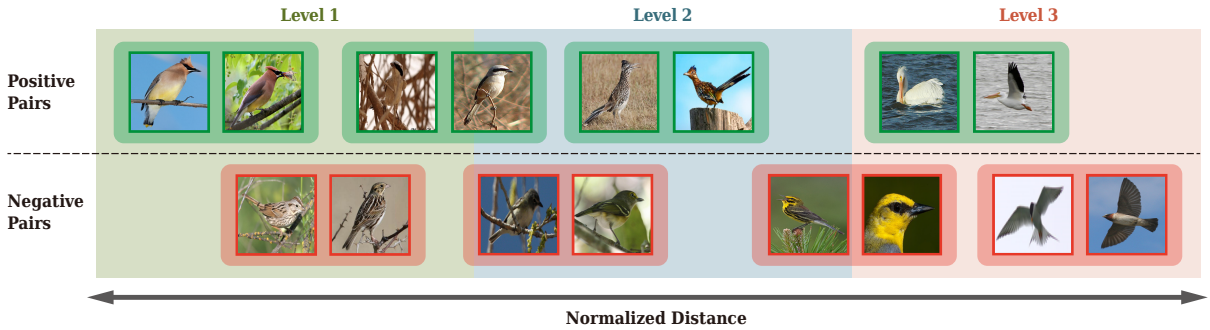


Figure 6: Visualization of assigned positive and negative pairs at each level on CUB-200. Regardless of positive or negative pair, a visually close pair is assigned to level 1, and a visually distant pair is assigned to level 3; even the same birds of the same species can be varying in appearance by the difference in perspectives, poses, and environments.

like the other methods, which suffer from severe overfitting. These observations indicate that our MDR is an effective regularizer for DML.

Equalizing the Two-Norm of Embedding Vectors. We find that the embedding vectors of a model trained with MDR have almost the same two-norm (Figure 4a and 4b). This shows that the embedding vectors are almost located on a hypersphere, even though the model is trained without L_2 Norm. Therefore, the model trained with MDR achieves similar performance even if L_2 Norm is applied at inference time (Table 3). This observation implies that MDR has similar effects of L_2 Norm at the end of the training, even though MDR is a distance-based regularization and L_2 Norm is norm-based regularization.

Discriminative Representation. To show the effectiveness of our method, we visualize how MDR constructs an embedding space. In the embedding space of Triplet and Triplet+L_2 Norm, the class centers are often aligned closely to each other (Figure 5a and 5b). However, in an embedding space of Triplet+MDR, the class centers are evenly spaced with a large margin (Figure 5c). This result indicates that MDR constructs a more discriminative representation than the conventional methods.

Qualitative Analysis on Level Assignment. In the step of the level assignment, a lower level indicates that the pairs are closely aligned in the embedding space and vice versa. Most of the positive pairs are belonging to between level 1 and 2, and most of the negative pairs are belonging to between level 2 and 3. However, hard-positive pairs may belong to level 3 while hard-negative also may belong to level 1 (Figure 6). Therefore, levels are assigned to each pair regardless of given binary supervision. The learning procedure tried to overcome the disturbance that pulls the distances to belonging levels by considering the various degrees of distances; this multi-level disturbance leads to the improvement of the generalization ability.

Conclusion

We introduce a new distance-based regularization method that elaborately adjusts the pairwise distance into multiple levels for better generalization. We prove the effectiveness of MDR by showing the improvements that greatly exceed the existing methods, and by extensively performing the ablation studies of its behaviors. By applying our MDR, many methods can be significantly improved without any extra burdens at inference time.

Acknowledgements

We would like to thank AI R&D team of Kakao Enterprise for the helpful discussion. In particular, we would like to thank Yunmo Park who designed the visual materials.

Potential Ethical Impact

Due to the gap between a training dataset and real-world data, it is important to build a reliable model with better generalization ability across the unseen dataset, *e.g.*test set, for its practicality. Our MDR is a regularization method to improve the generalization ability of a deep neural network on the task of deep metric learning. As positive aspects, our method can be applied to many practical applications such as image retrieval and item recommendation. These applications are utilized for our conveniences and the proposed MDR can improve their performance more reliably. We believe that our method does not have particular negative aspects because it is a fundamental method that assists conventional approaches to improve reliability on unseen datasets.

References

Bromley, J.; Guyon, I.; LeCun, Y.; Säckinger, E.; and Shah, R. 1994. Signature verification using a" siamese" time delay neural network. In *Advances in Neural Information Processing Systems*.

Cakir, F.; He, K.; Xia, X.; Kulis, B.; and Sclaroff, S. 2019. Deep metric learning to rank. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Deng, J.; Guo, J.; Xue, N.; and Zafeiriou, S. 2019. ArcFace: Additive Angular Margin Loss for Deep Face Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Duan, Y.; Zheng, W.; Lin, X.; Lu, J.; and Zhou, J. 2018. Deep adversarial metric learning. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Ge, W. 2018. Deep metric learning with hierarchical triplet loss. In *European Conference on Computer Vision*.

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

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hoffer, E.; and Ailon, N. 2015. Deep metric learning using triplet network. In *International Workshop on Similarity-Based Pattern Recognition*.

Ioffe, S.; and Szegedy, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167* .

Jacob, P.; Picard, D.; Histace, A.; and Klein, E. 2019. Metric Learning With HORDE: High-Order Regularizer for Deep Embeddings. In *International Conference on Computer Vision*.

Kim, W.; Goyal, B.; Chawla, K.; Lee, J.; and Kwon, K. 2018. Attention-based Ensemble for Deep Metric Learning. In *European Conference on Computer Vision*.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* .

Krause, J.; Stark, M.; Deng, J.; and Fei-Fei, L. 2013. 3d object representations for fine-grained categorization. In *IEEE International Conference on Computer Vision Workshops*.

Kumar BG, V.; Carneiro, G.; and Reid, I. 2016. Learning local image descriptors with deep siamese and triplet convolutional networks by minimising global loss functions. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Liu, W.; Wen, Y.; Yu, Z.; Li, M.; Raj, B.; and Song, L. 2017. SphereFace: Deep Hypersphere Embedding for Face Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Liu, Z.; Luo, P.; Qiu, S.; Wang, X.; and Tang, X. 2016. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Maaten, L. v. d.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research* .

Milbich, T.; Roth, K.; Bharadhwaj, H.; Sinha, S.; Bengio, Y.; Ommer, B.; and Cohen, J. P. 2020. DiVA: Diverse Visual Feature Aggregation for Deep Metric Learning. In *European Conference on Computer Vision*.

Oh Song, H.; Jegelka, S.; Rathod, V.; and Murphy, K. 2017. Deep metric learning via facility location. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Oh Song, H.; Xiang, Y.; Jegelka, S.; and Savarese, S. 2016. Deep Metric Learning via Lifted Structured Feature Embedding. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Opitz, M.; Waltner, G.; Possegger, H.; and Bischof, H. 2018. Deep Metric Learning with BIER: Boosting Independent Embeddings Robustly. *IEEE Transactions on Pattern Analysis and Machine Intelligence* .

Qian, Q.; Shang, L.; Sun, B.; Hu, J.; Li, H.; and Jin, R. 2019. SoftTriple Loss: Deep Metric Learning Without Triplet Sampling. *arXiv preprint arXiv:1909.05235* .

Revaud, J.; Almazán, J.; Rezende, R. S.; and Souza, C. R. d. 2019. Learning with average precision: Training image retrieval with a listwise loss. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Roth, K.; Milbich, T.; and Ommer, B. 2020. PADS: Policy-Adapted Sampling for Visual Similarity Learning. In *Proceedings of the IEEE International Conference on Computer Vision*.

Sanakoyeu, A.; Tschernezki, V.; Buchler, U.; and Ommer, B. 2019. Divide and Conquer the Embedding Space for Metric Learning. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Schroff, F.; Kalenichenko, D.; and Philbin, J. 2015. Facenet: A unified embedding for face recognition and clustering. In *IEEE Conference on Computer Vision and Pattern Recognition*.

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

Suh, Y.; Han, B.; Kim, W.; and Lee, K. M. 2019. Stochastic class-based hard example mining for deep metric learning. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Ustinova, E.; and Lempitsky, V. 2016. Learning deep embeddings with histogram loss. In *Advances in Neural Information Processing Systems*.

Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S. 2011. The caltech-ucsd birds-200-2011 dataset. *Technical Report* .

Wang, F.; Xiang, X.; Cheng, J.; and Yuille, A. L. 2017. NormFace: L2 Hypersphere Embedding for Face Verification. In *ACM International Conference on Multimedia*.

Wang, H.; Wang, Y.; Zhou, Z.; Ji, X.; Gong, D.; Zhou, J.; Li, Z.; and Liu, W. 2018. CosFace: Large Margin Cosine Loss for Deep Face Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Wang, X.; Han, X.; Huang, W.; Dong, D.; and Scott, M. R. 2019a. Multi-Similarity Loss with General Pair Weighting for Deep Metric Learning. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Wang, X.; Hua, Y.; Kodirov, E.; Hu, G.; Garnier, R.; and Robertson, N. M. 2019b. Ranked list loss for deep metric learning. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Wang, X.; Zhang, H.; Huang, W.; and Scott, M. R. 2020. Cross-Batch Memory for Embedding Learning. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Wu, C.-Y.; Manmatha, R.; Smola, A. J.; and Krahenbuhl, P. 2017. Sampling Matters in Deep Embedding Learning. In *IEEE International Conference on Computer Vision*.

Yi, D.; Lei, Z.; Liao, S.; and Li, S. Z. 2014. Deep metric learning for person re-identification. In *International Conference on Pattern Recognition*.

Yuan, Y.; Yang, K.; and Zhang, C. 2017. Hard-aware deeply cascaded embedding. In *IEEE International Conference on Computer Vision*.

Zhai, A.; and Wu, H.-Y. 2019. Classification is a Strong Baseline for Deep Metric Learning. In *British Machine Vision Conference*.