Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence (IJCAI-25)
Sister Conferences Best Papers Track

SEE: Spherical Embedding Expansion for Improving Deep Metric Learning (Extended Abstract)*

Binh M. Le , Simon S. Woo
Department of Computer Science & Engineering
Sungkyunkwan University, Suwon, South Korea
{bmle, swoo}@g.skku.edu

Abstract

The primary goal of deep metric learning is to construct a comprehensive embedding space that can effectively represent samples originating from both intra- and inter-classes. Although extensive prior work has explored diverse metric functions and innovative training strategies, much of this work relies on default training data. Consequently, the potential variations inherent within this data remain largely unexplored, constraining the model's robustness to unseen images. In this context, we introduce the **S**pherical **E**mbedding **E**xpansion (SEE) method. SEE aims to uncover the latent semantic variations in training data. Especially, our method augments the embedding space with synthetic representations based on Max-Mahalanobis distribution (MMD) centers, which maximize the dispersion of these synthetic features without increasing computational costs. We evaluated the efficacy of SEE on four renowned standard benchmarks for the image retrieval task. The results demonstrate that SEE consistently enhances the performance of conventional methods when integrated with them, setting a new benchmark for deep metric learning performance across all settings.

1 Introduction

Learning to create a semantic embedding space that possesses both discriminative and generalized properties has been extensively studied across a variety of machine learning tasks. Such tasks encompass image retrieval [Kim *et al.*, 2019], face verification [Deng *et al.*, 2019], person re-identification [Chen *et al.*, 2017], few-shot learning [Qiao *et al.*, 2019], and representation learning [Grill *et al.*, 2020]. Consequently, deep metric learning, facilitated by neural networks, has garnered significant attention. Its objective is to learn an efficient embedding space in which semantically similar sample are pulled close together, while dissimilar ones are pushed far apart. To this end, various training loss functions, which are broadly categorized into pair-based and proxy-based methods, have been proposed.

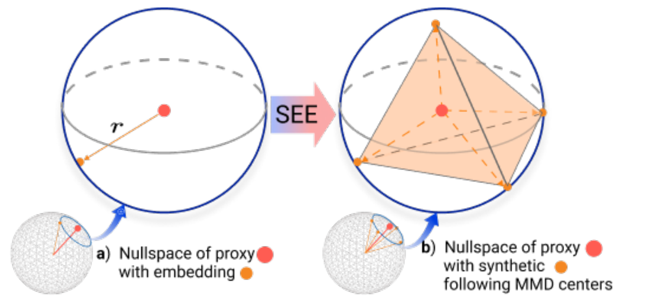

Figure 1: **Motivation of SEE.** SEE aims to discover the latent space of training data by synthesis new embedding vectors ($n_{\text{aug}} = 3$) derived from the nullspace $\boldsymbol{\mathcal{S}}^{d-2}_{||r||}$ of a proxy. The synthesize samples follows the MMD properties that enrich for the representation space, benefiting for optimization procedure.

In addition to refining the loss function, the development of sampling strategies is also pivotal in enhancing performance. Prevailing methods [Wu *et al.*, 2017] emphasize the mining of hard samples. However, this often results in a biased model, as it overlooks the majority of easy samples [Wu *et al.*, 2017; Zheng *et al.*, 2019]. To address this critical issue, recent research [Duan *et al.*, 2018; Zhao *et al.*, 2018; Zheng *et al.*, 2019] has suggested the use of generative adversarial networks or autoencoders to synthesize challenging samples using easy ones. Although promising, these approaches have drawbacks, such as model size and optimization issues. Other studies [Gu and Ko, 2020; Ko and Gu, 2020] have attempted to synthesize these challenging samples directly from the original embedding, yet they are predominantly constrained to paired-based techniques.

In this paper, we introduce a novel proxy-based synthesis technique in the embedding space of deep metric learning, termed as Spherical Embedding Expansion (SEE). As depicted in Figure 1, given an embedding and its corresponding proxy anchor, our approach initially explores the proxy's null space, which is represented as a sphere with a radius of r. This ensures consistent distances of the synthetic samples to the anchor. Subsequently, the synthetic embeddings are generated according to the Max-Mahalanobis distribution (MMD) mean vectors [Pang *et al.*, 2018] (hereinafter referred

*This is an extended abstract of the paper [Le and Woo, 2024] that won the Best Paper Running-Up Award at PAKDD 2024.

to as *MMD centers*), allowing for enabling a comprehensive exploration of the embedding space. Our method is straightforward and seamlessly integrates with existing proxy-based metric learning losses. Notably, implementing our approach neither alters the embedding network architecture nor impacts its training speed. Nonetheless, it enhances overall performance, especially in scenarios with low-dimensional spaces, having a large number of classes. Our contributions in this paper are summarized as follows:

- We propose a novel method that augments the embedding space during training by constructing synthetic feature points aligned with MMD centers.
- Through seamless integration, SEE improves proxy-based metric learning losses across numerous backbones and benchmarks without adding parameters.
- SEE excels at densely navigating embedding space, significantly boosting performance, particularly in low-dimensional spaces with datasets that have a large number of training classes.

2 Methodology

2.1 Preliminary

Consider a deep neural network, denoted as $f : \boldsymbol{\mathcal{D}} \to \boldsymbol{\mathcal{Z}}$, which maps an input data space $\boldsymbol{\mathcal{D}}$ to an embedding space $\boldsymbol{\mathcal{Z}}$ belonging to a unit d-dimensional hypersphere $\boldsymbol{\mathcal{S}}^{d-1}$. Let $y \in \boldsymbol{\mathcal{Y}} = \{1, ..., C\}$ be the label of an embedding feature z. We define a set of normalized proxies as $\boldsymbol{w} = \{w_1, w_2, ...w_C\}$ and formulate a general proxies-based loss function for metric learning as follows:

$$\mathcal{L}_{\text{ML}} = \mathop{\mathbb{E}}_{(z,y)\sim(\boldsymbol{\mathcal{Z}},\boldsymbol{\mathcal{Y}})} \ell(z|y, \boldsymbol{w}). \tag{1}$$

In Eq. 1, the normalized softmax loss [Wang *et al.*, 2017] and its variations [Teh *et al.*, 2020; Deng *et al.*, 2019; Wang *et al.*, 2018] are widely used as classification loss ℓ due to their interpretability and performance.

2.2 Spherical Embedding Expansion

Motivation. Our primary purpose of metric learning is to construct a robust and efficient embedding space for *unseen* samples. A common approach is to apply data augmentation techniques such as Mixup [Zhang *et al.*, 2017]. However, these techniques require forwarding augmented inputs to obtain augmented representations. In contrast, we introduce a plug-and-play module, Spherical Embedding Expansion (SEE), which operates in the embedding space $\boldsymbol{\mathcal{Z}}$. This method facilitates a more efficient augmentation process by allowing for the forwarding of un-augmented inputs and performing augmentations directly on the output representations. The conceptual illustration of SEE is provided in Fig. 2. In fact, the main motivation of our work is to address the following requirements:

Given an embedding vector z and its corresponding proxy w_y, how can we efficiently synthesize n_{aug} additional embedding vectors z_i^* that satisfy the following conditions: (1) The distances between the synthetic vectors and w_y, denoted as $d_{(y,i)}$, remain unchanged. (2) The distances between any two synthetic vectors, denoted as $d_{(i,j)}$, are maximized, resulting in optimal dispersion of synthetic vectors in the space.

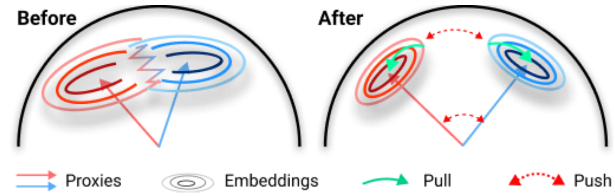

Figure 2: **A schematic representation of our learning objective**. **Left:** Training with a constrained dataset can result in underrepresented regions that fuse the representations of two distinct classes. **Middle:** SEE enhances intra-class samples, leading to denser clustering within each class while ensuring distinct separations between different classes.

The first requirement ensures that the synthetic vectors maintain similar quality to the original input and do not become outliers, or too close to their proxies. The second condition aims to diversify the distribution of the synthetic vectors in the embedding space, enabling the proxies of other classes more challenging and pushing those classes further away from their proxies.

Method. To ensure the first requirement, we define a $||r||$-radius $(d-1)$-dimensional hypersphere as: $\boldsymbol{\mathcal{S}}_{||r||}^{d-2} = \{\mu|\mu \perp w_y \wedge ||\mu|| = ||r||\}$, where $r = z - \langle w_y, z\rangle \cdot w_y$. This space $\boldsymbol{\mathcal{S}}_{||r||}^{d-2}$ represents the null space of w_y, and r is the projection of z onto this defined null space. As a result, for any $\mu \in \boldsymbol{\mathcal{S}}_{||r||}^{d-2}$, a synthetic vector formed by $z^* = \langle w_y, z\rangle \cdot w_y + \mu$ will satisfy $d(y, i) = d_y$. In practice, basis of this space can be constructed using Gram–Schmidt process. To generate a set of synthetic vectors z^*, one approach is to randomly sample n_{aug} vectors μ from the hyper-spherical space $\boldsymbol{\mathcal{S}}_{||r||}^{d-2}$ and translate them to z^*. However, randomly sampling n_{aug} vectors when $n_{\text{aug}} \ll d$ may not efficiently utilize the space. Conversely, if we choose a large value of n_{aug}, it will scale up the mini-batch size and affect computational efficiency. Hence, to fully utilize the space $\boldsymbol{\mathcal{S}}_{||r||}^{d-2}$ while maintaining efficiency, we need to satisfy the second requirement. This requirement aims to maximize the distance between any two synthetic vectors and achieve optimal dispersion in the embedding space. Inspired by the above analysis, we propose the Max-Mahalanobis center sampling method to induce high-density regions in the hyper-spherical space $\boldsymbol{\mathcal{S}}_{||r||}^{d-2}$, where the MMD [Pang *et al.*, 2018] is a mixture of Gaussian distributions with an identity covariance matrix and K preset centers denoted as $\boldsymbol{\mu}^* = \{\mu_j^*\}_{[K]}$. The MMD centers are created based on the criterion $\boldsymbol{\mu}^* = \arg\min_\mu \max_{i\neq j}\langle \mu_i, \mu_j\rangle$. This criterion aims to maximize the smallest angle between any two centers, resulting in the most dispersion of the centers across the entire hyper-spherical space [Pang *et al.*, 2018]. Previous work [Pang *et al.*, 2018] introduced a fixed set of $\boldsymbol{\mu}^*$. However, in our case, the centers vary depending on $r = \mu_1^*$, which is the image of z in the defined null space as illustrated in Figure 1. Additionally, the set of centers must satisfy the constraint $\mu_i^* \perp w_y$. To overcome this challenge,

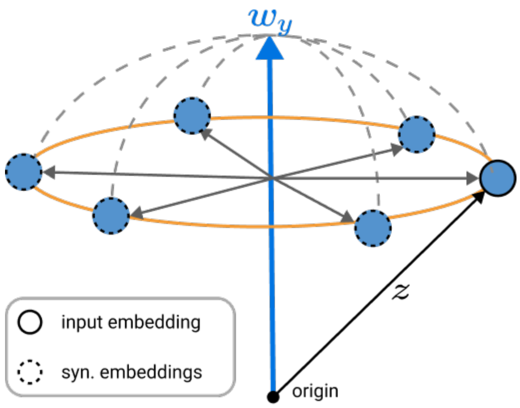

Figure 3: **Illustration of feature point generation.** For illustration, we depict hypersphere $\boldsymbol{\mathcal{S}}^{d-2}_{||r||}$ as a orange circle.

we propose a novel algorithm outlined in Alg. 1 to generate optimal expanded vectors following the centers of MMD within the constrained space $\boldsymbol{\mathcal{S}}^{d-2}_{||r||}$. The main difference between Alg. 1 and the *GenerateOptMeans* algorithm in [Pang *et al.*, 2018] are the initialization of $\mu_1^* = r/||r||$ *vs.* $\mu_1^* = e_1$ (one-hot vector), and subsequent MMD centers formalized in lines $2^{nd} - 5^{th}$. By using Alg. 1, one can easily prove that the set $\{\mu_i^*\}_{n_{\text{aug}}+1}$ are MMD centers, *i.e.*,

$$\mu_i^{*T}\mu_j^* = \begin{cases} 1, & i = j \\ -1/n_{\text{aug}}, & i \neq j \end{cases}, \tag{2}$$

but they are more flexible than [Pang *et al.*, 2018] in terms of initialization of μ_1^*. Hence, the *GenerateOptMeans* algorithm in [Pang *et al.*, 2018] is solely used as a regularization and inapplicable to synthesize new embedding vectors in our case. Consequently, the optimal sampled vectors, as shown in Figure 3, are produced as (line 6^{th} of Alg. 1): $z_i^* = \langle w_y, z\rangle \cdot w_y + ||r|| \cdot \mu_i^*$.

Although the synthetic embedding vectors can diversity its metric space, early applying the expansion can hinder model's optimization. Inspired by curriculum training scheme [Bengio *et al.*, 2009; Huang *et al.*, 2020], we selectively apply our method on top-k embedding vector z's such that its d_y in top-k smallest in one mini-batch, denoted as $\mathcal{M}_k$, where k is monotonously increasing after epochs. Therefore, at epoch t^{th}, we have the loss function for synthetic vectors as follows:

$$\mathcal{L}_{\text{SEE}} = \underset{(z,y)\sim(\boldsymbol{\mathcal{Z}},\boldsymbol{\mathcal{Y}}),d_y\in\mathcal{M}_k}{\mathbb{E}}\left[\sum\nolimits_{n_{\text{aug}}} \ell(z_i^*|y,\boldsymbol{w})\right]. \tag{3}$$

Overall Objective. Although our approach can help learning model to be more robust by diversely and optimally exploring the embedding space, it is important to note that a metric learning loss still play crucial roles as it utilizes the ground-truth labels for supervised training. The overall training loss for our proposed approach is formulated as follows:

$$\mathcal{L} = \mathcal{L}_{\text{ML}} + \lambda\mathcal{L}_{\text{SEE}}, \tag{4}$$

where λ is a hyper-parameter that balances the contribution of the original embedding and the synthetic vectors. It is important to note that our proposed approach does not require any

Algorithm 1 Generate optimal synthetic samples following MMD centers.

Require: Embedding vector z and its corresponding proxy vector w_y in $\boldsymbol{\mathcal{S}}^{d-1}$; number of expansion samples n_{aug}.
1: Initialization: Let $r = z - \langle w_y, z\rangle \cdot w_y$, $\boldsymbol{V} = \{v_0, v_1, ..., v_{n_{\text{aug}}+1}\}$, in which $v_0 = w_y$, $v_1 = r/||r||$, and $v_{i>1}$ are normalized vectors generated by Gram–Schmidt process sequentially. Let $\mu_1^* = v_1$.
2: **for** $k = 2$ to $n_{\text{aug}} + 1$ **do**
3: $\mu_k^* = \sum_{i=1}^{k} \alpha_{ki} v_i$, where
4: $\begin{cases} \alpha_{k1} = -1/n_{\text{aug}} \\ \alpha_{kj} = -\left(1 + n_{\text{aug}} \cdot \sum_{i=1}^{j-1} \alpha_{ki}\alpha_{ji}\right) / \left(n_{\text{aug}} \cdot \alpha_{jj}\right) \\ \alpha_{kk} = \sqrt{1 - \sum_{i=1}^{k-1} \alpha_{ki}^2} \end{cases}$
5: **end for**
6: **Return** $\{z_k^* = \langle w_y, z\rangle \cdot w_y + ||r|| \cdot \mu_k^*\}_{k\in\overline{2,\ldots,n_{\text{aug}}+1}}$.

modification to the loss function. It can be used as a plug-and-play module in the training process, introducing negligible computational cost.

Discussion Incorporating SEE into a deep metric model yields two pronounced effects. Fostering a more generalized model by comprehensively exploring of under-represented regions; and pushing negative anchors by creating hard negative samples. Specifically, taking the normalized softmax loss in Eq. 1 as a simple example, we rewrite it as:

$$\ell(z|y,\boldsymbol{w}) = \tau\text{Softplus}\left[\text{LSE}_{j\neq y}(d_y - d_j)/\tau\right], \tag{5}$$

where $\epsilon \geq 0$, and d_j represents the distance between z and the proxy w_j, such as $d_j = -\langle w_j, z\rangle = -\cos\theta_j$, and $\text{Softplus}(x) = \log(\epsilon + e^x)$. As illustrated in Figure 3, the synthetic feature points z_i^* maintain consistent distances to their respective proxy anchors; that is, all d_ys are identical. Furthermore, the Log-Sum-Exp (LSE) function serves as a smooth approximation to the maximum function [Nielsen and Sun, 2016]. Thus, our SEE is adept at effectively pushing the most challenging negative anchors (represented by the smallest d_j) with every synthetic feature point.

3 Experiments

3.1 Settings

We use the following four popular benchmark datasets for evaluateing our method: 1) CUB-200-2011 (CUB)[Wah *et al.*, 2011], 2) Cars-196 (Cars) [Krause *et al.*, 2013], 3) Stanford Online Product (SOP) [Oh Song *et al.*, 2016] , 4) In-shop Clothes Retrieval (In-Shop) [Liu *et al.*, 2016]. We utilize the Recall@k as experimental evalution metric. Regarding backbones, we adopt ResNet50 [He *et al.*, 2016] (R) with an embedding size of $d = 512$ and three versions of vision transformer architecture: DeiT-S [Touvron *et al.*, 2021a] (D), DINO [Caron *et al.*, 2021] (DN), and ViT-S [Dosovitskiy *et al.*, 2020] (V), each with embedding sizes $d = 128$ and $d = 384$. These models are optimized with AdamW optimizer [Loshchilov and Hutter, 2017] and a learning rate of 10^{-5} for ViT-S and DeiT-S, and 5×10^{-6} for DINO models.

Methods	Arch.	CUB			Cars			SOP			In-Shop		
		R@1	R@2	R@4	R@1	R@2	R@4	R@1	R@10	R@100	R@1	R@10	R@20
Backbone architecture: CNN													
NSoftmax [Zhai and Wu, 2018]	R^{128}	56.5	69.6	79.9	81.6	88.7	93.4	75.2	88.7	95.2	86.6	96.8	97.8
MIC [Roth *et al.*, 2019]	R^{128}	66.1	76.8	85.6	82.6	89.1	93.2	77.2	89.4	94.6	88.2	97.0	-
XBM [Wang *et al.*, 2020]	R^{128}	-	-	-	-	-	-	80.6	91.6	96.2	91.3	97.8	98.4
XBM [Wang *et al.*, 2020]	B^{512}	65.8	75.9	84.0	82.0	88.7	93.1	79.5	90.8	96.1	89.9	97.6	98.4
HTL [Ge, 2018]	B^{512}	57.1	68.8	78.7	81.4	88.0	92.7	74.8	88.3	94.8	80.9	94.3	95.8
MS [Wang *et al.*, 2019]	B^{512}	65.7	77.0	86.3	84.1	90.4	94.0	78.2	90.5	96.0	89.7	97.9	98.5
SoftTriple [Qian *et al.*, 2019]	B^{512}	65.4	76.4	84.5	84.5	90.7	94.5	78.6	86.6	91.8	-	-	-
PA [Kim *et al.*, 2020]	B^{512}	68.4	79.2	86.8	86.1	91.7	95.0	79.1	90.8	96.2	91.5	98.1	98.8
NSoftmax [Zhai and Wu, 2018]	R^{512}	61.3	73.9	83.5	84.2	90.4	94.4	78.2	90.6	96.2	86.6	97.5	98.4
†ProxyNCA++ [Teh *et al.*, 2020]	R^{512}	69.0	79.8	87.3	86.5	92.5	95.7	80.7	92.0	96.7	90.4	98.1	98.8
Hyp [Ermolov *et al.*, 2022]	R^{512}	65.5	76.2	84.9	81.9	88.8	93.1	79.9	91.5	96.5	90.1	98.0	98.7
SEE (*ours*)	R^{512}	69.3	79.0	87.3	88.5	93.4	95.9	80.3	91.5	96.5	92.8	98.3	98.8
Backbone architecture: ViT													
IRT$_R$ [El-Nouby *et al.*, 2021]	De^{128}	72.6	81.9	88.7	-	-	-	83.4	93.0	97.0	91.1	98.1	98.6
Hyp [Ermolov *et al.*, 2022]	De^{128}	74.7	84.5	90.1	82.1	89.1	93.4	83.0	93.4	97.5	90.9	97.9	98.6
SEE (*ours*)	De^{128}	75.1	84.1	90.1	85.2	91.5	94.8	83.0	93.1	97.2	91.2	98.0	98.6
Hyp [Ermolov *et al.*, 2022]	DN^{128}	78.3	86.0	91.2	86.0	91.9	95.2	84.6	94.1	97.7	92.6	98.4	99.0
SEE (*ours*)	DN^{128}	78.8	86.5	91.6	89.0	93.6	96.3	84.8	94.1	97.5	92.6	98.6	99.0
Hyp [Ermolov *et al.*, 2022]	V^{128}	84.0	90.2	94.2	82.7	89.7	93.9	85.5	94.9	98.1	92.7	98.4	98.9
SEE (*ours*)	V^{128}	84.1	90.2	93.5	86.8	91.7	95.1	85.9	94.7	97.9	92.8	98.6	99.1
IRT$_R$ [El-Nouby *et al.*, 2021]	De^{384}	76.6	85.0	91.1	-	-	-	84.2	93.7	97.3	91.9	98.1	98.9
DeiT-S [Touvron *et al.*, 2021b]	De^{384}	70.6	81.3	88.7	52.8	65.1	76.2	58.3	73.9	85.9	37.9	64.7	72.1
Hyp [Ermolov *et al.*, 2022]	De^{384}	77.8	86.6	91.9	86.4	92.2	95.5	83.3	93.5	97.4	90.5	97.8	98.5
SEE (*ours*)	De^{384}	78.3	86.5	91.9	88.8	93.7	96.3	83.6	93.4	97.4	91.7	98.1	98.7
DNO [Caron *et al.*, 2021]	DN^{384}	70.8	81.1	88.8	42.9	53.9	64.2	63.4	78.1	88.3	46.1	71.1	77.5
Hyp [Ermolov *et al.*, 2022]	DN^{384}	80.9	87.6	92.4	89.2	94.1	96.7	85.1	94.4	97.8	92.4	98.4	98.9
SEE (*ours*)	DN^{384}	81.9	88.8	92.9	91.5	95.2	97.3	85.5	94.6	97.9	93.0	98.5	99.1
ViT-S [Krause *et al.*, 2013]	V^{384}	83.1	90.4	94.4	47.8	60.2	72.2	62.1	77.7	89.0	43.2	70.2	76.7
Hyp [Ermolov *et al.*, 2022]	V^{384}	85.6	91.4	94.8	86.5	92.1	95.3	85.9	94.9	98.1	92.5	98.3	98.8
SEE (*ours*)	V^{384}	85.8	91.4	94.6	88.8	93.8	96.4	86.3	95.0	98.2	93.2	98.6	99.1

Table 1: **Performance of metric learning methods on the four datasets**.

3.2 Results

Our experimental findings, as presented in Table 1, underscore the effectiveness of our proposed methodology. When compared with other CNN-based techniques, our approach, utilizing ResNet50 as the backbone, consistently outperforms competitors across multiple datasets, with the exception of SOP. Comparing to methods like ProxyNCA++ [Teh *et al.*, 2020] which employs a more expansive input size, or XBM [Wang *et al.*, 2020] which leverages an extensive memory bank to augment their training process, our method still surpasses them in most datasets. In the context of ViT-based experiments, our proposed methodology exhibits a discernible advantage over competing baselines, particularly with respect to R@1 scores, spanning various embedding dimensions. Furthermore, even on challenging datasets such as SOP or In-shop, our technique continues to demonstrate marked improvements, even at small dimensions like 128 and 384.

For the flexibility of our proposed synthesis method, we demonstrate the enhancements achieved when applying it to various proxy-based metric learning losses, specifically when integrated with CNN-based architectures. The detailed results are presented in Table 2. As evident from the table, our synthesis approach consistently enhances performance across different proxy-based losses, achieving up to a 1.8% improvement in Recall@1 accuracy. Furthermore, this enhancement persists even as we increase the number of neighbors k.

For more details on the empirical analyses, please see our Supplementary Material and [Le and Woo, 2024].

Method	Arch.	R@1	R@2	R@4	R@8	R@16
NSoftmax [Zhai and Wu, 2018]	R^{512}	84.2	90.4	94.4	96.9	-
NSoftmax+SEE	R^{512}	86.5	92.0	95.4	97.4	98.7
CosFace [Wang *et al.*, 2018]	R^{512}	86.9	92.3	95.3	97.4	98.6
CosFace+SEE	R^{512}	87.1	92.5	95.4	97.5	98.7
ArcFace [Deng *et al.*, 2019]	R^{512}	86.8	92.1	95.3	97.3	98.7
ArcFace+SEE	R^{512}	87.6	92.8	95.9	97.6	98.7
†ProxyNCA++ [Teh *et al.*, 2020]	R^{512}	86.5	92.5	95.7	97.7	-
†ProxyNCA++ +SEE	R^{512}	88.3	93.4	96.4	98.0	99.0
PA [Kim *et al.*, 2020]	B^{512}	86.1	91.7	95.0	97.0	98.3
PA+SEE	B^{512}	86.2	91.9	95.2	97.2	98.4
PA [Kim *et al.*, 2020]	R^{512}	87.7	92.7	95.5	97.3	98.4
PA+SEE	R^{512}	88.5	93.4	95.9	97.5	98.8

Table 2: **Recall@k for proxy-based losses integrated with our SEE.**

4 Conclusions

In this work, we have introduced a spherical embedding expansion technique for augmentation within the embedding space, designed to complement existing proxy-based metric learning losses. Within this space, we augment a sample around its anchor by adhering to the MMD centers situated within the anchor's nullspace, thereby ensuring a thorough exploration. Our proposed method is streamlined and straightforward, obviating the need to modify model architecture or incur computational overhead. Empirical results reveal that our approach considerably enhances the efficacy of established proxy-based losses across a range of model architectures and benchmark datasets.

Acknowledgements

This work was partly supported by Institute for Information & communication Technology Planning & evaluation (IITP) grants funded by the Korean government MSIT: (RS-2022-II221199, RS-2022-II220688, RS-2019-II190421, RS-2023-00230337, RS-2024-00356293, RS-2021-II212068, RS-2024-00437849, RS-2025-02263841, RS-2025-02304983, and RS-2024-00436936).

References

[Bengio *et al.*, 2009] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48, 2009.

[Caron *et al.*, 2021] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[Chen *et al.*, 2017] Weihua Chen, Xiaotang Chen, Jianguo Zhang, and Kaiqi Huang. Beyond triplet loss: a deep quadruplet network for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 403–412, 2017.

[Deng *et al.*, 2019] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2019.

[Dosovitskiy *et al.*, 2020] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[Duan *et al.*, 2018] Yueqi Duan, Wenzhao Zheng, Xudong Lin, Jiwen Lu, and Jie Zhou. Deep adversarial metric learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2780–2789, 2018.

[El-Nouby *et al.*, 2021] Alaaeldin El-Nouby, Natalia Neverova, Ivan Laptev, and Hervé Jégou. Training vision transformers for image retrieval. *arXiv preprint arXiv:2102.05644*, 2021.

[Ermolov *et al.*, 2022] Aleksandr Ermolov, Leyla Mirvakhabova, Valentin Khrulkov, Nicu Sebe, and Ivan Oseledets. Hyperbolic vision transformers: Combining improvements in metric learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7409–7419, 2022.

[Ge, 2018] Weifeng Ge. Deep metric learning with hierarchical triplet loss. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 269–285, 2018.

[Grill *et al.*, 2020] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, 33:21271–21284, 2020.

[Gu and Ko, 2020] Geonmo Gu and Byungsoo Ko. Symmetrical synthesis for deep metric learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10853–10860, 2020.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[Huang *et al.*, 2020] Yuge Huang, Yuhan Wang, Ying Tai, Xiaoming Liu, Pengcheng Shen, Shaoxin Li, Jilin Li, and Feiyue Huang. Curricularface: adaptive curriculum learning loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5901–5910, 2020.

[Kim *et al.*, 2019] Sungyeon Kim, Minkyo Seo, Ivan Laptev, Minsu Cho, and Suha Kwak. Deep metric learning beyond binary supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2288–2297, 2019.

[Kim *et al.*, 2020] Sungyeon Kim, Dongwon Kim, Minsu Cho, and Suha Kwak. Proxy anchor loss for deep metric learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3238–3247, 2020.

[Ko and Gu, 2020] Byungsoo Ko and Geonmo Gu. Embedding expansion: Augmentation in embedding space for deep metric learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7255–7264, 2020.

[Krause *et al.*, 2013] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561, 2013.

[Le and Woo, 2024] Binh Minh Le and Simon S Woo. See: Spherical embedding expansion for improving deep metric learning. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 131–143. Springer, 2024.

[Liu *et al.*, 2016] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1096–1104, 2016.

[Loshchilov and Hutter, 2017] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[Nielsen and Sun, 2016] Frank Nielsen and Ke Sun. Guaranteed bounds on the kullback–leibler divergence of univariate mixtures. *IEEE Signal Processing Letters*, 23(11):1543–1546, 2016.

[Oh Song *et al.*, 2016] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4004–4012, 2016.

[Pang *et al.*, 2018] Tianyu Pang, Chao Du, and Jun Zhu. Max-mahalanobis linear discriminant analysis networks. In *International Conference on Machine Learning*, pages 4016–4025. PMLR, 2018.

[Qian *et al.*, 2019] Qi Qian, Lei Shang, Baigui Sun, Juhua Hu, Hao Li, and Rong Jin. Softtriple loss: Deep metric learning without triplet sampling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6450–6458, 2019.

[Qiao *et al.*, 2019] Limeng Qiao, Yemin Shi, Jia Li, Yaowei Wang, Tiejun Huang, and Yonghong Tian. Transductive episodic-wise adaptive metric for few-shot learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3603–3612, 2019.

[Roth *et al.*, 2019] Karsten Roth, Biagio Brattoli, and Bjorn Ommer. Mic: Mining interclass characteristics for improved metric learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8000–8009, 2019.

[Teh *et al.*, 2020] Eu Wern Teh, Terrance DeVries, and Graham W Taylor. Proxynca++: Revisiting and revitalizing proxy neighborhood component analysis. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16*, pages 448–464. Springer, 2020.

[Touvron *et al.*, 2021a] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021.

[Touvron *et al.*, 2021b] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021.

[Wah *et al.*, 2011] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[Wang *et al.*, 2017] Feng Wang, Xiang Xiang, Jian Cheng, and Alan Loddon Yuille. Normface: L2 hypersphere embedding for face verification. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1041–1049, 2017.

[Wang *et al.*, 2018] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5265–5274, 2018.

[Wang *et al.*, 2019] Xun Wang, Xintong Han, Weilin Huang, Dengke Dong, and Matthew R Scott. Multi-similarity loss with general pair weighting for deep metric learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5022–5030, 2019.

[Wang *et al.*, 2020] Xun Wang, Haozhi Zhang, Weilin Huang, and Matthew R Scott. Cross-batch memory for embedding learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6388–6397, 2020.

[Wu *et al.*, 2017] Chao-Yuan Wu, R Manmatha, Alexander J Smola, and Philipp Krahenbuhl. Sampling matters in deep embedding learning. In *Proceedings of the IEEE international conference on computer vision*, pages 2840–2848, 2017.

[Zhai and Wu, 2018] Andrew Zhai and Hao-Yu Wu. Classification is a strong baseline for deep metric learning. *arXiv preprint arXiv:1811.12649*, 2018.

[Zhang *et al.*, 2017] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

[Zhao *et al.*, 2018] Yiru Zhao, Zhongming Jin, Guo-jun Qi, Hongtao Lu, and Xian-sheng Hua. An adversarial approach to hard triplet generation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 501–517, 2018.

[Zheng *et al.*, 2019] Wenzhao Zheng, Zhaodong Chen, Jiwen Lu, and Jie Zhou. Hardness-aware deep metric learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 72–81, 2019.