S2SD: SIMULTANEOUS SIMILARITY-BASED SELF-DISTILLATION FOR DEEP METRIC LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Deep Metric Learning (DML) provides a crucial tool for visual similarity and zero-shot retrieval applications by learning generalizing embedding spaces, although recent work in DML has shown strong performance saturation across training objectives. However, generalization capacity is known to scale with the embedding space dimensionality. Unfortunately, high dimensional embeddings also create higher retrieval cost for downstream applications. To remedy this, we propose *S2SD - Simultaneous Similarity-based Self-distillation*. *S2SD* extends DML with knowledge distillation from auxiliary, high-dimensional embedding and feature spaces to leverage complementary context during training while retaining test-time cost and with negligible changes to the training time. Experiments and ablations across different objectives and standard benchmarks show *S2SD* offering notable improvements of up to 7% in Recall@1, while also setting a new state-of-the-art.

1 INTRODUCTION

Deep Metric Learning (*DML*) aims to learn embedding space ($\mathcal{E}$) models in which a predefined distance metric reflects not only the semantic similarities between training samples, but also transfers to unseen classes. The generalization capabilities of these models are important for applications in image retrieval (Wu et al., 2017), face recognition (Schroff et al., 2015), clustering (Bouchacourt et al., 2018) and representation learning (He et al., 2020). Still, transfer learning into unknown test distributions remains an open problem, with Roth et al. (2020b) and Musgrave et al. (2020) revealing strong performance saturation across DML training objectives. However, Roth et al. (2020b) also show that embedding space dimensionality ($\mathcal{D}$) can be a driver for generalization across objectives due to higher representation capacity. Indeed, this insight can be linked to recent work targeting other objective-independent improvements to DML via artificial samples (Zheng et al., 2019), higher feature distribution moments (Jacob et al., 2019) or orthogonal features (Milbich et al., 2020), which have shown promising relative improvements over selected DML objectives. Unfortunately, these methods come at a cost; be it longer training times or limited applicability. Similarly, drawbacks can be found when naively increasing the operating (*base*) $\mathcal{D}$, incurring increased cost for data retrieval at test time, which is especially problematic on larger datasets. This limits realistically usable $\mathcal{D}$s and leads to benchmarks being evaluated against fixed, predefined $\mathcal{D}$s.

In this work, we propose *Simultaneous Similarity-based Self-Distillation* (*S2SD*) to show that complex higher-dimensional information can actually be effectively leveraged in DML without changing the base $\mathcal{D}$ and test time cost, which we motivate from two key elements. Firstly, in DML, an additional $\mathcal{E}$ can be spanned by a multilayer perceptron (MLP) operating over the feature representation shared with the base $\mathcal{E}$ (see e.g. (Milbich et al., 2020)). With larger $\mathcal{D}$, we can thus cheaply learn a secondary high-dimensional $\mathcal{E}$ simultaneously, also denoted as *target* $\mathcal{E}$. Relative to the large feature backbone, and with the *batchsize* capping the number of additional high dimensional operations, only little additional training cost is introduced. While we can not utilize the high-dim. target $\mathcal{E}$ at test time for aforementioned reasons, we may utilize it to boost the performance of the base $\mathcal{E}$.

Unfortunately, a simple connection of base and target $\mathcal{E}$s through the shared feature backbone is insufficient for the base $\mathcal{E}$ to benefit from the auxiliary, high-dimensional information. Thus, secondly, to efficiently leverage the high-dimensional context, we use insights from knowledge distillation (Hinton et al., 2015), where a small "student" model is trained to approximate a larger "teacher" model. However, while knowledge distillation can be found in DML (Chen et al., 2018), few-shot learning

(Tian et al., 2020) and self-supervised extensions thereof (Rajasegaran et al., 2020), the reliance on additional, commonly larger teacher networks or multiple training runs (Furlanello et al., 2018), introduces much higher training cost. Fortunately, we find that the target $\mathcal{E}$ learned *simultaneously* at higher dimension can sufficiently serve as a "teacher" *during* training - through knowledge distillation of its sample similarities, the performance of the base $\mathcal{E}$ can be improved notably. Such distillation intuitively encourages the lower-dimensional base $\mathcal{E}$ to embed semantic similarities similar to the more expressive target $\mathcal{E}$ and thus incorporate dimensionality-related generalization benefits.

Furthermore, *S2SD* makes use of the low cost to span additional $\mathcal{E}s$ to introduce multiple target $\mathcal{E}$s. Operating each of them at higher, but varying dimensionality, joint distillation can then be used to enforce reusability in the distilled content akin to feature reusability in meta-learning (Raghu et al., 2020) for additional generalization boosts. Finally, in DML, the base $\mathcal{E}$ is spanned over a penultimate feature space of much higher dimensionality, which introduces a dimensionality-based bottleneck (Milbich et al., 2020). By applying the distillation objective between feature and base embedding space in *S2SD*, we further encourage better feature usage in base $\mathcal{E}$. This facilitates the approximation of high-dimensional context through the base $\mathcal{E}$ for additional improvements in generalization.

The benefits to generalization are highlighted in performance boosts across three standard benchmarks, CUB200-2011 (Wah et al., 2011), CARS196 (Krause et al., 2013) and Stanford Online Products (Oh Song et al., 2016), where *S2SD* improves test-set recall@1 of already strong DML objectives by up to 7%, while also setting a new state-of-the-art. Improvements are even more significant in very low dimensional base $\mathcal{E}$s, making *S2SD* attractive for large-scale retrieval problems which can benefit from reduced $\mathcal{D}$s. Importantly, as *S2SD* is applied **during** the same DML training process on the **same** network backbone, no large teacher networks or additional training runs are required. Simple experiments even show that *S2SD* can outperform comparable 2-stage distillation at much lower cost.

In summary, our contributions can be described as:
1) We propose *Simultaneous Similarity-based Self-Distillation* (*S2SD*) for DML, using knowledge distillation of high-dimensional context without large additional teacher networks or training runs.
2) We motivate and evaluate this approach through detailed ablations and experiments, showing that the method is agnostic to choices in objectives, backbones, and datasets.
3) Across benchmarks, we achieve significant improvements over strong baseline objectives and state-of-the-art performance, with especially large boosts for very low-dimensional embedding spaces.

2 Related Work

Deep Metric Learning (DML) has proven useful for zero-shot image/video retrieval & clustering (Schroff et al., 2015; Wu et al., 2017; Brattoli et al., 2020), face verification (Liu et al., 2017; Deng et al., 2019) and contrastive (self-supervised) representation learning (e.g. He et al. (2020); Chen et al. (2020); Misra & van der Maaten (2020)). Approaches can be divided into **1)** improved ranking losses, **2)** tuple sampling methods and **3)** extensions to the standard DML training approach.

1) Ranking losses place constraints on relations in image tuples ranging from pairs (e.g. Hadsell et al. (2006)) to triplets (Schroff et al., 2015) and more complex orderings (Chen et al., 2017; Oh Song et al., 2016; Sohn, 2016; Wang et al., 2019). **2)** The number of possible tuples scales exponentially with dataset size, leading to many tuple sampling approaches to ensure meaningful tuples presented during training. These tuple sampling methods can follow heuristics (Schroff et al. (2015); Wu et al. (2017)), be of hierarchical nature (Ge, 2018) or learned (Roth et al., 2020a). Similarly, learnable proxies to replace tuple members (Movshovitz-Attias et al., 2017; Kim et al., 2020; Qian et al., 2019) can also remedy the sampling issue, which can be extended to tackle DML from a classification viewpoint (Zhai & Wu, 2018; Deng et al., 2019). **3)** Finally, extensions to the basic training scheme can involve synthetic data (Lin et al., 2018; Zheng et al., 2019; Duan et al., 2018), complementary features (Roth et al., 2019; Milbich et al., 2020), a division into subspaces (Sanakoyeu et al., 2019; Xuan et al., 2018; Kim et al., 2018; Opitz et al., 2018), training of multiple networks (Park et al., 2020) using mutual learning Zhang et al. (2018) or higher-order moments (Jacob et al., 2019).
S2SD can similarly be seen as an extension to DML, though we specifically focus on capturing and distilling complex high-dimensional sample relations within lower dimensional embedding spaces to improve generalization.

Knowledge Distillation involves knowledge transfer from teacher to (usually smaller) student models, e.g. by matching network softmax outputs/logits (Buciluǎ et al., 2006; Hinton et al., 2015), (attention-weighted) feature maps (Romero et al., 2015; Zagoruyko & Komodakis, 2016), or latent representations (Ahn et al., 2019; Park et al., 2019; Tian et al., 2019; Laskar & Kannala, 2020). Importantly, Tian et al. (2019) show that under fair comparison, basic matching via Kullback-Leibler (KL) Divergences as used in Hinton et al. (2015) performs very well, which we also find to be the case for *S2SD*. This is further supported in recent few-shot learning literature (Tian et al., 2020), wherein KL-distillation alongside self-distillation (by iteratively reusing the same network as a teacher (Furlanello et al., 2018; Lan et al., 2018)) in additional meta-training stages improves feature representation strength important for generalization (Raghu et al., 2020).

More specifically, our work most closely resembles Zhang et al. (2019) and Liu et al. (2020), which propose to break down a network into a cascading set of subnetworks, wherein each subsequent subnetwork builds on its predecessors. In doing so, each subnetwork is trained independently on a classification task at hand. Knowledge distillation is then applied either from the full network (Zhang et al., 2019) acting as a teacher or via soft targets generated from a meta-learned label generator (Liu et al., 2020), to each smaller student subnetwork during the same training run to improve overall performance. In a related manner, *S2SD* utilizes similar concurrent, but relational self-distillation to instead encode high-dimensional sample relation context from multiple, higher-dimensional teacher embedding spaces; crucial to improve the generalization capabilities of a single student embedding space for zero-shot, out-of-distribution image retrieval tasks. As such, it operates orthogonally to proposals made by Zhang et al. (2019) and Liu et al. (2020). The concurrency of the self-distillation in turn is a consequence of the novel insight that solely the dimensionality of embedding spaces can serve as meaningful teachers, as these can be spanned cheaply over a large, shared feature backbone.

The novel dimensionality-based concurrent distillation also sets S2SD apart from existing knowledge distillation applications to DML, which are done in a generic manner with separate, larger teacher networks or additional training stages (Chen et al., 2018; Yu et al., 2019; Han et al., 2019; Laskar & Kannala, 2020).

3 METHOD

We now introduce key elements for *Simultaneous Similarity-based Self-Distillation (S2SD)* to improve generalization of embedding spaces by utilizing higher dimensional context. We start with preliminary notation and fundamentals to Deep Metric Learning (§3.1). We then define the three key elements to *S2SD*: Firstly, the Dual Self-Distillation (DSD) objective, which uses KL-Distillation on a concurrently learned high-dimensional embedding space (§3.2) to introduce the high-dimensional context into a low-dimensional embedding space during training. We then extend this to Multiscale Self-Distillation (MSD) with distillation from several different high-dimensional embedding spaces to encourage reusability in the distilled context (§3.3). Finally, we shift to self-distillation from normalized feature representations to counter dimensionality bottlenecks (MSDF) (§3.4).

3.1 PRELIMINARIES

DML builds on generic Metric Learning which aims to find a (parametrized) distance metric $d_\theta : \Phi \times \Phi \mapsto \mathbb{R}$ on the *feature space* $\Phi \subset \mathbb{R}^{d^*}$ over images $\mathcal{X}$ that best satisfy ranking constraints usually defined over class labels $\mathcal{Y}$. This holds also for DML. However, while Metric Learning relies on a **fixed** feature extraction method to obtain Φ, DML introduces deep neural networks to concurrently learn a feature representation. Most such DML approaches aim to learn Mahalanobis distance metrics, which cover the parametrized family of inner product metrics (Suárez et al., 2018; Chen et al., 2019). These metrics, with some restrictions (Suárez et al., 2018), can be reformulated as

$$d(\phi_1, \phi_2) = \sqrt{(L(\phi_1 - \phi_2)^T L(\phi_1 - \phi_2)} = \|L\phi_1 - L\phi_2\|_2 = \|\psi_1 - \psi_2\|_2 \tag{1}$$

with learned linear projection $L \in \mathbb{R}^{d \times d^*}$ from d^*-dim. *features* $\phi_i \in \Phi$ to d-dim. *embeddings* $\psi_i := (f \circ \phi)(x_i) \in \Psi_f$ with embedding function $f : \phi_i \mapsto L\phi_i$. Importantly, this redefines the motivation behind DML as learning d-dimensional image embeddings ψ s.t. their euclidean distance $d(\bullet, \bullet) = \|\bullet - \bullet\|_2$ is connected to semantic similarities in $\mathcal{X}$. This embedding-based formulation offers the significant advantage of being compatible with fast approximate similarity search methods

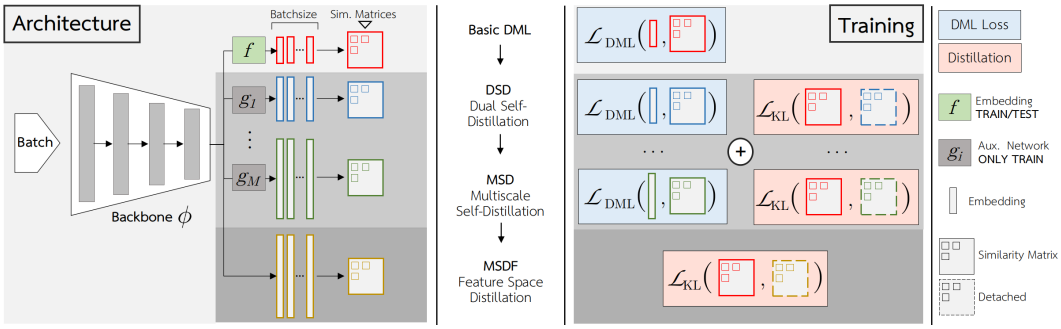

Figure 1: *S2SD*. We use a standard encoder ϕ, embedding f, and multiple auxiliary embedding networks g_i (used only during training) depending on the *S2SD* approach used. During training, for each batch of embeddings produced by the respective embedding network g_i, we compute DML losses while applying embedding distillation on the respective batch-similarity matrices (*DSD/MSD*). We further distill from the feature representation space for additional information gain (*MSDF*).

(e.g. Johnson et al. (2017)), allowing for large-scale applications at test time. In this work, we assume Ψ_f to be normalized to the unit hypersphere $\mathcal{S}_{\Psi_f}$, which is commonly done (Wu et al., 2017; Sanakoyeu et al., 2019; Liu et al., 2017; Wang & Isola, 2020) for beneficial regularizing purposes (Wu et al., 2017; Wang & Isola, 2020). For the remainder we hence set Ψ to refer to S_Ψ.
Common approaches to learn such a representation space involve training surrogates on ranking constraints defined by class labels. Such approaches start from pair or triplet-based ranking objectives (Hadsell et al., 2006; Schroff et al., 2015), where the latter is defined as

$$\mathcal{L}_{\text{triplet}} = {}^{1}/_{|\mathcal{T}_\mathcal{B}|} \sum_{(x_i,x_j,x_k)\in\mathcal{T}_\mathcal{B}} [d(\psi_i,\psi_j) - d(\psi_i,\psi_k) + m]_+ \tag{2}$$

with margin m and the set of available triplets $(x_i, x_j, x_k) \in \mathcal{T}_\mathcal{B}$ in a mini-batch $\mathcal{B} \subset \mathcal{X}$, with $y_i = y_j \neq y_k$. This can be extended with more complex ranking constraints or tuple sampling methods. We refer to Supp. B and Roth et al. (2020b) for further insights and detailed studies.

3.2 Embedding Space Self-Distillation

For the aforementioned standard DML setting, generalization performance of a learned embedding space can be linked to the utilized embedding dimensionality. However, high dimensionality results in notably higher retrieval cost on downstream applications, which limits realistically usable dimensions. In *S2SD*, we show that high-dimensional context can be used as a teacher during the training run of the low-dimensional *base* or *reference* embedding space. As the base embedding model is also the one that is evaluated, test time retrieval costs are left unchanged.

To achieve this, we simultaneously train an additional high-dimensional *auxiliary/target* embedding space $\Psi_g := (g \circ \phi)(\mathcal{X})$ spanned by a secondary embedding branch g. g is parametrized by a MLP or a linear projection, similar to the base embedding space Ψ_f spanned by f, see §3.1. Both f and g operate on the same large, shared feature backbone ϕ. For simplicity, we train Ψ_f and Ψ_g using the same DML objective $\mathcal{L}_{\text{DML}}$.

Unfortunately, higher expressivity and improved generalization of high-dimensional embeddings in Ψ_g hardly benefit the base embedding space, even with a shared feature backbone. To explicitly leverage high-dimensional context for our base embedding space, we utilize knowledge distillation from target to base space. However, while common knowledge distillation approaches match single embeddings or features between student and teacher, the different dimensionalities in Ψ_f and Ψ_g inhibit naive matching.

Instead, *S2SD* matches sample relations (see e.g. Tian et al. (2019)) defined over batch-similarity matrices $D \in \mathbb{R}^{\mathcal{B}\times\mathcal{B}}$ in base and target space, D^f and D^g, with batchsize $\mathcal{B}$. We thus encourage the base embedding space to relate different samples in a similar manner to the target space. To compute D, we use a cosine similarity by default, given as $D_{i,j} = \psi_i^T \psi_j$, since ψ_i is normalized to the unit hypersphere. Defining σ_{max} as the softmax operation and $\mathcal{D}_{\text{KL}}(p,q) = \sum \log(p)^{\log(p)}/_{\log(q)}$ as the

Kullback-Leibler-divergence, we thus define the simultaneous self-distillation objective as

$$\mathcal{L}_{\text{dist}}(D^f, D^g) = \sum_i^{|\mathcal{B}|} \mathcal{D}_{\text{KL}}\left(\sigma_{\max}\left({}^{D^f_{i,:}}/_{T}\right), \sigma^{\dagger}_{\max}\left({}^{D^g_{i,:}}/_{T}\right)\right) \tag{3}$$

with temperature T, as visualized in Figure 1. (†) denotes no gradient flow to target branches g as we only want the base space to learn from the target space. By default, we match rows or columns of D, $D_{i,:}$, effectively distilling the relation of an anchor embedding ψ_i to all other batch samples. Embedding all batch samples in base dimension, $\Psi^{\mathcal{B}}_f : \mathcal{B} \mapsto \psi_f(\mathcal{B})$, and higher dimension, $\Psi^{\mathcal{B}}_g : \mathcal{B} \mapsto \psi_g(\mathcal{B})$, the (simultaneous) *Dual Self-Distillation* (DSD) training objective then becomes

$$\mathcal{L}_{\text{DSD}}(\Psi^{\mathcal{B}}_f, \Psi^{\mathcal{B}}_g) = {}^1/_2 \cdot \left[\mathcal{L}_{\text{DML}}(\Psi^{\mathcal{B}}_f) + \mathcal{L}_{\text{DML}}(\Psi^{\mathcal{B}}_g)\right] + \gamma \cdot \mathcal{L}_{\text{dist}}(D^f, D^g) \tag{4}$$

3.3 Reusable Sample Relations by Multiscale Self-distillation

While *DSD* encourages the reference embedding space to recover complex sample relations by distilling from a higher-dimensional target space spanned by g, it is not known *a priori* which distillable sample relations actually benefit generalization of the reference space.

To encourage the usage of sample relations that more likely aid generalization, we follow insights made in Raghu et al. (2020) on the connection between **reusability** of features across multiple tasks and better generalization thereof. We motivate reusability in *S2SD* by extending *DSD* to *Multiscale Self-Distillation* (*MSD*) with distillation instead from m multiple different target spaces spanned by $G = \{g_k\}_{k\in\{1,m\}}$. Importantly, each of these high-dimensional target spaces operate on different dimensionalities, i.e. $dim\ f < dim\ g_1 < ... < dim\ g_{m-1} < dim\ g_m$. As this results in each target embedding space encoding sample relations differently, application of distillation across all spaces spanned by G pushes the base branch towards learning from sample relations that are reusable across all higher dimensional embedding spaces and thereby more likely to generalize (see also Fig. 1).

Specifically, given the set of target similarity matrices $\{D^k\}_{k\in\{f,g_1,...,g_m\}}$ and target batch embeddings $\Gamma^m := \{\Psi^{\mathcal{B}}_k\}_{k\in\{f,g_1,...,g_m\}}$, we then define the *MSD* training objective as

$$\mathcal{L}_{\text{MSD}}(\Gamma^m) = {}^1/_2 \cdot \left[\mathcal{L}_{\text{DML}}(\Psi^{\mathcal{B}}_f) + {}^1/_m \sum_{i=1}^m \mathcal{L}_{\text{DML}}(\Psi^{\mathcal{B}}_{g_i})\right] + {}^\gamma/_m \sum_{i=1}^m \mathcal{L}_{\text{dist}}(D^f, D^{g_i}) \tag{5}$$

3.4 Tackling the Dimensionality Bottleneck by Feature Space Self-Distillation

As noted in §3.1, the base embedding space Ψ utilizes a linear projection f from the (penultimate) feature space Φ where $dim\ \Phi$ is commonly much larger than $dim\ \Psi$. While compressed semantic spaces encourage stronger representations (Alemi et al., 2016; Dai & Wipf, 2019) to be learned, Milbich et al. (2020) show that the actual test performance of the lower-dimensional embedding space Φ is inferior to that of the non-adapted, but higher-dimensional feature space Ψ.

This supports a dimensionality-based loss of information beneficial to generalization, which can hinder the base embedding space to optimally approximate the high-dimensional context introduced in §3.2 and 3.3.
To rectify this, we apply self-distillation following eq. 3 on the normalized feature representations Φ^n generated by normalizing the backbone output ϕ. With the batch of normalized feature representations $\Psi^{\mathcal{B}}_{\phi^n}$ we get *multiscale self-distillation with feature distillation* (*MSDF*) (see also Fig. 1)

$$\mathcal{L}_{\text{MSDF}}(\Gamma^m, \Psi^{\mathcal{B}}_{\phi^n}) = \mathcal{L}_{\text{MSD}}(\Gamma^m) + \gamma\mathcal{L}_{\text{dist}}(D^f, D^{\phi^n}) \tag{6}$$

In the same manner, one can also address other architectural information bottlenecks such as through the generation of feature representations from a single global pooling operation. While not noted in the original publication, Kim et al. (2020) address this in the official code release by using both global max- and average pooling to create their base embedding space. While this naive usage changes the architecture at test time, in *S2SD* we can *fairly* leverage potential benefits by *only* spanning the auxiliary spaces (and distilling) from such feature representations (denoted as *DSDA*/*MSDA*/*MSDFA*).

4 Experimental Setup

We study *S2SD* in four experiments to establish 1) method ablation performance & relative improvements, 2) state-of-the-art, 3) comparisons to standard 2-stage distillation, benefits to low-dimensional embedding spaces & generalization properties and 4) motivation for architectural choices.

Method Notation. We abbreviate ablations of *S2SD* (see §3) in our experiments as: *DSD* & *MSD* for **D***ual* (3.2) & **M***ultiscale* **S***elf*-**D***istillation* (3.3), *MSDF* the addition of **F***eature distillation* (3.4) and *DSDA*/*MSD(F)A* the inclusion of multiple pooling operations in the auxiliary branches (also §3.4).

4.1 Experiments

Fair Evaluation of Ablations. §5.1 specifically applies *S2SD* and its ablations to three DML baselines. To show realistic benefit, *S2SD* is applied to best-performing objectives evaluated in Roth et al. (2020b), namely *(i)* Margin loss with Distance-based Sampling (Wu et al., 2017), *(ii)* their proposed Regularized Margin loss and *(iii)* Multisimilarity loss (Wang et al., 2019), following their experimental training pipeline. This setup utilizes no learning rate scheduling and fixes common implementational factors of variation in DML pipelines such as batchsize, base embedding dimension, weight decay or feature backbone architectures to ensure comparability in DML (more details in Supp. A.2). As such, our results are directly comparable to their large set of examined methods and guaranteed that relative improvements solely stem from the application of *S2SD*.

Evaluation Across Architectures and Embedding Dimensions. §5.2 further highlights the benefits of *S2SD* by comparing *S2SD*'s boosting properties across literature standards, with different backbone architectures and base embedding dimensions: *(1)* ResNet50 with $d = 128$ (Wu et al., 2017; Roth et al., 2019) and *(2)* $d = 512$ (Zhai & Wu, 2018) as well as *(3)* variants to Inception-V1 with Batch-Normalization at $d = 512$ (Wang et al., 2019; Qian et al., 2019; Milbich et al., 2020). Only here do we conservatively apply learning rate scheduling, since all references noted in Table 2 employ scheduling as well. We categorize published work based on backbone architecture and embedding dimension for fairer comparison. Note that this is a less robust comparison than done in §5.1, due to potential implementation differences between our pipeline and reported literature results.

Comparison to 2-Stage Distillation and Generalization Study. §5.3 compares *S2SD* to 2-stage distillation, investigates benefits to very low dimensional reference spaces and examines the connection between improvements and increased embedding space feature richness, measured by density and spectral decay (see Supp. D), which are linked to improved generalization in Roth et al. (2020b).

Investigation of Method Choices. §5.4 finally ablates and motivates specific architectural choices in *S2SD* used throughout §4. Pseudo code and detailed results are available in Supp. F, G, and I.

4.2 Implementation

Datasets & Evaluation. In all experiments, we evaluate on standard DML benchmarks: *CUB200-2011* (Wah et al., 2011), *CARS196* (Krause et al., 2013) and *Stanford Online Products (SOP)* (Oh Song et al., 2016). Performance is measured in *recall at 1* (R@1) and *at 2* (R@2) (Jegou et al., 2011) as well as *Normalized Mutual Information* (NMI) (Manning et al., 2010). More details in Supp. A & C.

Experimental Details. Our implementation follows Roth et al. (2020b), with more details in Supp. (A). For §5.1-5.4, we only adjust the respective pipeline elements in questions. For *S2SD*, unless noted otherwise (s.a. in §5.4), we set $\gamma = 50, T = 1$ for all objectives on CUB200 and CARS196, and $\gamma = 5, T = 1$ on SOP. *DSD* uses target-dim. $d = 2048$ and MSD target-dims. $d \in [512, 1024, 1536, 2048]$. We found it beneficial to activate the feature distillation after $n = 1000$ iterations for CUB200, CARS196 and SOP, respectively, to ensure that meaningful features are learned first before feature distillation is applied. The additional embedding spaces are generated by two layer MLPs with row-wise KL-distillation of similarities (eq. 3), applied as in $\mathcal{L}_{\text{multi}}$ (eq. 5). By default, we use Multisimilarity Loss as stand-in for $\mathcal{L}_{\text{DML}}$.

5 Results

5.1 S2SD Improves Performance under Fair Evaluation

In Tab. 1 (full table in Supp. Tab. 4), we show that under the fair experimental protocol used in Roth et al. (2020b), utilizing *S2SD* and its ablations gives an objective and benchmark independent, significant boost in performance by up to 7% opposing the exisiting DML objective performance plateau. This holds even for regularized objectives s.a. R-Margin loss, highlighting the effectiveness of *S2SD* for DML. Across objectives, *S2SD*-based changes in wall-time do not exceed negligible 5%.

Table 1: *S2SD comparison against strong baseline objectives.* All results computed over multiple seeds. **Bold** denotes best results per loss & benchmark, **bluebold** marks best results per benchmark. Evaluations using the mAP@R metric as proposed in Roth et al. (2020b) and Musgrave et al. (2020) can be found in the Supplementary (Table 5), similarly showing the notable benefits of S2SD.

BENCHMARKS→	CUB200-2011		CARS196		SOP	
APPROACHES ↓	R@1	NMI	R@1	NMI	R@1	NMI
Margin, $\beta = 1.2$, (Wu et al., 2017)	63.09 ± 0.46	68.21 ± 0.33	79.86 ± 0.33	67.36 ± 0.34	78.43 ± 0.07	90.40 ± 0.03
+ DSD	65.11 ± 0.18	69.65 ± 0.44	83.19 ± 0.18	69.28 ± 0.56	79.05 ± 0.12	90.52 ± 0.18
+ MSD	66.13 ± 0.34	70.83 ± 0.27	83.63 ± 0.31	69.80 ± 0.36	79.26 ± 0.15	90.60 ± 0.10
+ MSDF	$\mathbf{67.58 \pm 0.32}$	$\mathbf{71.47 \pm 0.19}$	85.55 ± 0.23	$\mathbf{71.68 \pm 0.54}$	$\mathbf{79.63 \pm 0.15}$	$\mathbf{90.70 \pm 0.09}$
+ MSDFA	67.21 ± 0.23	71.43 ± 0.25	$\mathbf{86.45 \pm 0.35}$	71.46 ± 0.24	78.82 ± 0.09	90.49 ± 0.06
R-Margin, $\beta = 0.6$, (Roth et al., 2020b)	64.93 ± 0.42	68.36 ± 0.32	82.37 ± 0.13	68.66 ± 0.47	77.58 ± 0.11	90.42 ± 0.03
+ DSD	66.58 ± 0.08	70.03 ± 0.41	84.64 ± 0.16	70.87 ± 0.18	77.86 ± 0.10	90.50 ± 0.03
+ MSD	66.81 ± 0.27	70.47 ± 0.16	85.01 ± 0.10	71.67 ± 0.40	78.00 ± 0.06	90.47 ± 0.04
+ MSDF	68.12 ± 0.30	$\mathbf{71.80 \pm 0.33}$	85.78 ± 0.22	$\mathbf{72.24 \pm 0.31}$	$\mathbf{78.57 \pm 0.09}$	$\mathbf{90.58 \pm 0.02}$
+ MSDFA	$\mathbf{68.58 \pm 0.26}$	71.64 ± 0.40	$\mathbf{86.81 \pm 0.35}$	71.48 ± 0.29	78.00 ± 0.11	90.41 ± 0.02
Multisimilarity (Wang et al., 2019)	62.80 ± 0.70	68.55 ± 0.38	81.68 ± 0.19	69.43 ± 0.38	77.99 ± 0.09	90.00 ± 0.02
+ DSD	65.57 ± 0.26	70.08 ± 0.33	83.51 ± 0.20	70.30 ± 0.05	78.23 ± 0.04	90.08 ± 0.04
+ MSD	65.80 ± 0.16	70.66 ± 0.01	83.98 ± 0.10	71.34 ± 0.09	78.42 ± 0.09	90.09 ± 0.03
+ MSDF	67.04 ± 0.29	$\mathbf{71.87 \pm 0.19}$	85.69 ± 0.19	$\mathbf{72.77 \pm 0.13}$	$\mathbf{78.59 \pm 0.08}$	$\mathbf{90.09 \pm 0.06}$
+ MSDFA	$\mathbf{67.68 \pm 0.29}$	71.40 ± 0.21	$\mathbf{85.89 \pm 0.15}$	71.45 ± 0.26	78.07 ± 0.06	89.88 ± 0.10

Table 2: *State-of-the-art comparison.* We show that *S2SD*, represented by its variants *MSDF(A)*, boosts baseline objectives to state-of-the-art across literature. (*) stands for Inception-V1 with frozen Batch-Norm. **Bold**: best results per literature setup. **Bluebold**: best results per overall benchmark.

BENCHMARKS →	CUB200 (Wah et al., 2011)			CARS196 (Krause et al., 2013)			SOP (Oh Song et al., 2016)		
METHODS ↓	R@1	R@2	NMI	R@1	R@2	NMI	R@1	R@10	NMI
ResNet50-128									
Div&Conq (Sanakoyeu et al., 2019)	65.9	76.6	69.6	84.6	90.7	70.3	75.9	88.4	90.2
MIC (Roth et al., 2019)	66.1	76.8	69.7	82.6	89.1	68.4	77.2	89.4	90.0
PADS (Roth et al., 2020a)	67.3	78.0	69.9	83.5	89.7	68.8	76.5	89.0	89.9
Multisimilarity+*S2SD*	68.0 ± 0.2	78.7 ± 0.1	71.7 ± 0.4	86.3 ± 0.1	91.8 ± 0.3	72.0 ± 0.3	79.0 ± 0.2	90.2 ± 0.1	90.6 ± 0.1
Margin+*S2SD*	67.6 ± 0.3	78.2 ± 0.2	70.8 ± 0.3	86.0 ± 0.2	91.8 ± 0.2	72.2 ± 0.2	$\mathbf{80.2 \pm 0.2}$	$\mathbf{91.5 \pm 0.1}$	$\mathbf{90.9 \pm 0.1}$
R-Margin+*S2SD*	$\mathbf{68.9 \pm 0.3}$	$\mathbf{79.0 \pm 0.3}$	$\mathbf{72.1 \pm 0.4}$	$\mathbf{87.6 \pm 0.2}$	$\mathbf{92.7 \pm 0.2}$	$\mathbf{72.3 \pm 0.2}$	79.2 ± 0.2	90.3 ± 0.1	90.8 ± 0.1
ResNet50-512									
EPSHN (Xuan et al., 2020)	64.9	75.3	-	82.7	89.3	-	78.3	90.7	-
NormSoft (Zhai & Wu, 2018)	61.3	73.9	-	84.2	90.4	-	78.2	90.6	-
DiVA (Milbich et al., 2020)	69.2	79.3	71.4	87.6	92.9	72.2	79.6	91.2	90.6
Multisimilarity+*S2SD*	69.2 ± 0.1	79.1 ± 0.2	71.4 ± 0.2	89.2 ± 0.2	93.8 ± 0.2	$\mathbf{74.0 \pm 0.2}$	80.8 ± 0.2	$\mathbf{92.2 \pm 0.2}$	90.5 ± 0.3
Margin+*S2SD*	68.8 ± 0.2	78.5 ± 0.2	$\mathbf{72.3 \pm 0.1}$	89.3 ± 0.2	93.8 ± 0.2	73.7 ± 0.3	$\mathbf{81.0 \pm 0.2}$	92.1 ± 0.2	$\mathbf{91.1 \pm 0.3}$
R-Margin+*S2SD*	$\mathbf{70.1 \pm 0.2}$	$\mathbf{79.7 \pm 0.2}$	71.6 ± 0.2	$\mathbf{89.5 \pm 0.2}$	$\mathbf{93.9 \pm 0.3}$	72.9 ± 0.3	80.0 ± 0.2	91.4 ± 0.2	90.8 ± 0.1
Inception-BN-512									
DiVA (Milbich et al., 2020)	66.8	77.7	70.0	84.1	**90.7**	68.7	78.1	**90.6**	90.4
Multisimilarity+*S2SD*	66.7 ± 0.3	77.5 ± 0.3	$\mathbf{70.5 \pm 0.2}$	83.8 ± 0.3	90.3 ± 0.2	$\mathbf{69.8 \pm 0.3}$	$\mathbf{78.5 \pm 0.2}$	$\mathbf{90.6 \pm 0.2}$	$\mathbf{90.6 \pm 0.1}$
Margin+*S2SD*	66.8 ± 0.2	77.9 ± 0.2	69.9 ± 0.3	$\mathbf{84.3 \pm 0.2}$	$\mathbf{90.7 \pm 0.2}$	$\mathbf{69.8 \pm 0.2}$	78.4 ± 0.2	90.5 ± 0.2	90.4 ± 0.1
R-Margin+*S2SD*	$\mathbf{67.4 \pm 0.3}$	$\mathbf{78.0 \pm 0.4}$	70.3 ± 0.2	83.9 ± 0.3	90.3 ± 0.2	69.4 ± 0.2	78.1 ± 0.2	90.4 ± 0.3	90.3 ± 0.2
Softtriple* (Qian et al., 2019)	65.4	76.4	69.3	84.5	90.7	70.1	78.3	90.3	**92.0**
Multisimilarity* (Wang et al., 2019)	65.7	77.0	-	84.1	90.4	-	78.2	90.5	-
Multisimilarity*+*S2SD*	68.2 ± 0.3	79.1 ± 0.2	$\mathbf{71.6 \pm 0.2}$	86.3 ± 0.2	92.2 ± 0.2	72.0 ± 0.3	78.9 ± 0.2	90.8 ± 0.2	90.6 ± 0.1
Margin*+*S2SD*	68.3 ± 0.2	78.8 ± 0.2	71.2 ± 0.2	$\mathbf{87.1 \pm 0.2}$	$\mathbf{92.4 \pm 0.1}$	$\mathbf{72.2 \pm 0.2}$	$\mathbf{79.1 \pm 0.2}$	$\mathbf{91.0 \pm 0.3}$	90.4 ± 0.1
R-Margin*+*S2SD*	$\mathbf{69.6 \pm 0.3}$	$\mathbf{79.6 \pm 0.3}$	71.2 ± 0.1	86.6 ± 0.3	92.1 ± 0.3	70.9 ± 0.2	78.5 ± 0.1	90.5 ± 0.2	90.0 ± 0.2

5.2 S2SD ACHIEVES SOTA ACROSS ARCHITECTURE AND EMBEDDING DIMENSION

Motivated by Tab. 1, we use *MSDFA* for CUB200/CARS196 and *MSDF* for SOP. Table 2 shows that *S2SD* can boost baseline objectives to reach and even surpass SOTA, in parts with a notable margin, even when reported with confidence intervals, which is commonly neglected in DML. *S2SD* outperforms much more complex methods with feature mining or RL-policies s.a. MIC (Roth et al., 2019), DiVA (Milbich et al., 2020) or PADS (Roth et al., 2020a).

5.3 *S2SD* IS A STRONG SUBSTITUTE FOR NORMAL DISTILLATION & LEARNS GENERALIZING EMBEDDING SPACES ACROSS DIMENSIONALITIES.

Comparison to standard distillation. With student S (same objective/embed. dim. as the reference branch in *DSD*) and a teacher T at highest optimal dim. $d = 2048$, we find separating *DSD* into

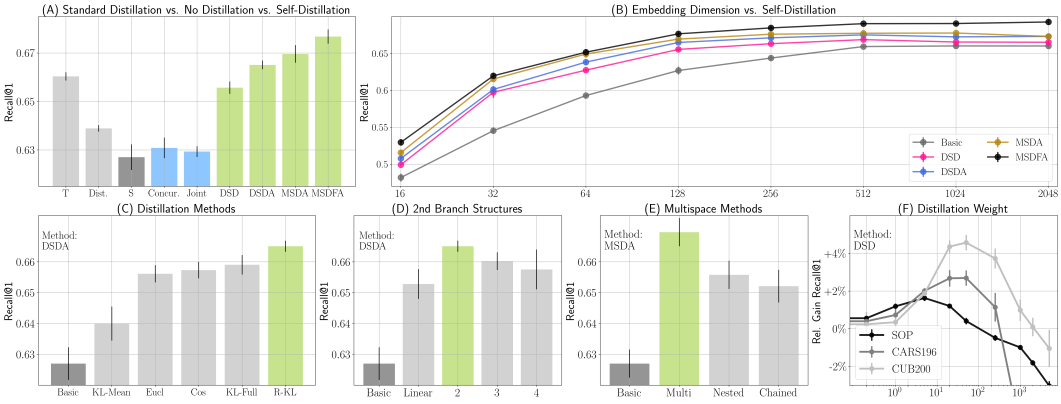

Figure 3: *S2SD study and ablations.* **(A)** *DSD* outperforms comparable two-stage distillation on student *S* (*Dist.*) using teacher (*T*), with *MSD(FA)* even outperforming the teacher. We further see that distillation is essential for improvement - training multiple spaces in parallel (*Joint.*) or a detached lower-dimensional base embedding (*Concur.*) gives little benefit. **(B)** We see benefits across base dimensionalities, especially in the low-dimensional regime. **(C)** We find KL-distillation between similarity vectors (*R-KL*) to work best. **(D)** An additional non-linearity in aux. branches g gives a boost, but going deeper degenerates generalization. **(E)** Distilling each aux. embed. space (*Multi*) to the reference space compares favourable against other distillation setups s.a. *Nested* and *Chained* distillation. **(F)** We find performance to be robust to changes in weight values.

standard 2-stage distillation degenerates performance (see Fig. 3A, compare to *Dist.*). *S2SD* also allows for easy integration of teacher ensembles, realized by *MSD(F,A)*, to even outperform the teacher notably *while* operating on the embedding dimensionality of the student.

Benefits to lower base dimensions. We show that our module is able to vastly boost networks limited to very low embedding dimensions (c.f. 3B). For example, $d = 32$ & 64 networks trained with *S2SD* can match the performance of embed. dimensions *four or eight times* the size. For $d = 128$, *S2SD* even outperforms the highest dimensional baseline at $d = 2048$ notably.

Embedding space metrics. We now look at relative changes in embedding space density and spectral decay as in Roth et al. (2020b), although we investigate changes within the same objectives. Fig. 2 shows *S2SD* increasing embedding space density and lowering the spectral decay (hence providing a more feature-diverse embedding space) across criteria.

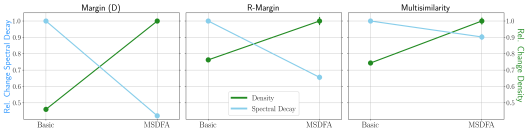

Figure 2: *Generalization metrics. S2SD* increases embed. space density and lowers spectral decay.

5.4 Motivating S2SD Architecture Choices

Is distillation in *S2SD* important? Fig. 3A (*Joint*) and Fig. 3F ($\gamma = 0$) highlight how crucial self-distillation is, as using a secondary embedding space without hardly improves performance. Fig. 3A (*Concur.*) shows that joint training of a detached reference embedding f while otherwise training in high dimension also doesn't offer notable improvement. Finally, Figure 3F shows robustness to changes in γ, with peaks around $\gamma = 50$ and $\gamma = 5$ for CUB200/CARS196 and SOP. We also found best performance for temperatures $T \in [0.2, 2]$ and hence set $T = 1$ by default.

Best way to enforce reusability. To motivate our many-to-one self-distillation $\mathcal{L}_{\text{MSD}}$ (eq. 5, here also dubbed $\mathcal{L}_{\text{Multi}}$), we evaluate against other distillation setups that could support reusability of distilled sample relations: *(1) Nested* distillation, where instead of distilling all target spaces only to the reference space, we distill from a target space to *all* lower-dimensional embedding spaces:

$$\mathcal{L}_{\text{Nested}}(\Gamma^m) = \frac{1}{2}\left[\mathcal{L}_{\text{DML}}(\Psi_f^{\mathcal{B}}) + \frac{1}{m}\sum_{i=1}^{m}\mathcal{L}_{\text{DML}}(\Psi_{g_i}^{\mathcal{B}})\right] + \frac{\gamma}{\binom{m}{m-1}} \sum_{\substack{i=0, j=1, j\neq i \\ dim\ g_j \geq dim\ g_i}}^{m} \mathcal{L}_{\text{dist}}(\Psi_{g_i}^{\mathcal{B}}, \Psi_{g_j}^{\mathcal{B}}) \quad (7)$$

In the second term, g_0 denotes the base embedding f. *(2) Chained* distillation, which distills from a target space only to the lower-dim. embedding space closest in dimensionality:

$$\mathcal{L}_{\text{Chained}}(\Gamma^m) = \frac{1}{2}\left[\mathcal{L}_{\text{DML}}(\Psi_f^{\mathcal{B}}) + \frac{1}{m}\sum_{i=1}^{m}\mathcal{L}_{\text{DML}}(\Psi_{g_i}^{\mathcal{B}})\right] + \frac{\gamma}{m}\sum_{i=0}^{m-1}\mathcal{L}_{\text{dist}}(\Psi_{g_i}^{\mathcal{B}}, \Psi_{g_j}^{\mathcal{B}}) \tag{8}$$

Figure 3E shows that while either distillation method provides strong benefits, a many-to-one distillation performs notably better, supporting the reusability aspect and $\mathcal{L}_{\text{multi}}$ as our default method.

Choice of distillation method & branch structures. Fig. 3C evaluates various distillation objectives, finding KL-divergence between vectors of similarities to perform better than KL-divergence applied over full similarity matrices or row-wise means thereof, as well as cosine/euclidean distance-based distillation (see e.g. (Yu et al., 2019)). Figure 3D shows insights into optimal auxiliary branch structures, with two-layer MLPs giving the largest benefit, although even a linear target mapping reliably boosts performance. This coincides with insights made by Chen et al. (2020). Further network depth only deteriorates performance.

6 Conclusion

In this paper, we propose a novel knowledge-distillation based DML training paradigm, *Simultaneous Similarity-based Self-Distillation* (*S2SD*), to utilize high-dimensional context for improved generalization. *S2SD* solves the standard DML objective simultaneously in higher-dimensional embedding spaces while applying knowledge distillation concurrently between these high-dimensional teacher spaces and a lower-dimensional reference space. *S2SD* introduces little additional computational overhead, with no extra cost at test time. Thorough ablations and experiments show *S2SD* significantly improving the generalization performance of existing DML objectives regardless of embedding dimensionality, while also setting a new state-of-the-art on standard benchmarks.

REFERENCES

Sungsoo Ahn, Shell Xu Hu, Andreas Damianou, Neil D. Lawrence, and Zhenwen Dai. Variational information distillation for knowledge transfer. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2019. doi: 10.1109/cvpr.2019.00938. URL http://dx.doi.org/10.1109/CVPR.2019.00938.

Alexander A. Alemi, Ian Fischer, Joshua V. Dillon, and Kevin Murphy. Deep variational information bottleneck. *CoRR*, abs/1612.00410, 2016. URL http://arxiv.org/abs/1612.00410.

Diane Bouchacourt, Ryota Tomioka, and Sebastian Nowozin. Multi-level variational autoencoder: Learning disentangled representations from grouped observations. In *AAAI 2018*, 2018.

Biagio Brattoli, Joseph Tighe, Fedor Zhdanov, Pietro Perona, and Krzysztof Chalupka. Rethinking zero-shot video classification: End-to-end training for realistic applications. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 535–541, 2006.

Shuo Chen, Lei Luo, Jian Yang, Chen Gong, Jun Li, and Heng Huang. Curvilinear distance metric learning. In *Advances in Neural Information Processing Systems 32*, pp. 4223–4232. Curran Associates, Inc., 2019. URL http://papers.nips.cc/paper/8675-curvilinear-distance-metric-learning.pdf.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Everest Hinton. A simple framework for contrastive learning of visual representations. 2020. URL https://arxiv.org/abs/2002.05709.

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*, 2017.

Yuntao Chen, Naiyan Wang, and Zhaoxiang Zhang. Darkrank: Accelerating deep metric learning via cross sample similarities transfer, 2018. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17147.

Bin Dai and David P. Wipf. Diagnosing and enhancing VAE models. *CoRR*, abs/1903.05789, 2019. URL http://arxiv.org/abs/1903.05789.

J. Deng, J. Guo, N. Xue, and S. Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4685–4694, 2019. doi: 10.1109/CVPR.2019.00482.

Yueqi Duan, Wenzhao Zheng, Xudong Lin, Jiwen Lu, and Jie Zhou. Deep adversarial metric learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

Tommaso Furlanello, Zachary Chase Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born-again neural networks. In Jennifer G. Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1602–1611. PMLR, 2018. URL http://proceedings.mlr.press/v80/furlanello18a.html.

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

Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

Jiaxu Han, Tianyu Zhao, and Changqing Zhang. Deep distillation metric learning. *Proceedings of the ACM Multimedia Asia*, 2019.

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*, pp. 770–778, 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. pp. 448–456, 2015. URL http://jmlr.org/proceedings/papers/v37/ioffe15.pdf.

Pierre Jacob, David Picard, Aymeric Histace, and Edouard Klein. Metric learning with horde: High-order regularizer for deep embeddings. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Herve Jegou, Matthijs Douze, and Cordelia Schmid. Product quantization for nearest neighbor search. *IEEE transactions on pattern analysis and machine intelligence*, 33(1):117–128, 2011.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *arXiv preprint arXiv:1702.08734*, 2017.

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 (CVPR)*, June 2020.

Wonsik Kim, Bhavya Goyal, Kunal Chawla, Jungmin Lee, and Keunjoo Kwon. Attention-based ensemble for deep metric learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

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*, pp. 554–561, 2013.

Xu Lan, Xiatian Zhu, and Shaogang Gong. Self-referenced deep learning. *CoRR*, abs/1811.07598, 2018. URL http://arxiv.org/abs/1811.07598.

Zakaria Laskar and Juho Kannala. Data-efficient ranking distillation for image retrieval. *CoRR*, abs/2007.05299, 2020. URL https://arxiv.org/abs/2007.05299.

Xudong Lin, Yueqi Duan, Qiyuan Dong, Jiwen Lu, and Jie Zhou. Deep variational metric learning. In *The European Conference on Computer Vision (ECCV)*, September 2018.

Benlin Liu, Yongming Rao, Jiwen Lu, Jie Zhou, and Cho-jui Hsieh. Metadistiller: Network self-boosting via meta-learned top-down distillation. *CoRR*, abs/2008.12094, 2020. URL https://arxiv.org/abs/2008.12094.

Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Stuart P. Lloyd. Least squares quantization in pcm. *IEEE Trans. Information Theory*, 28:129–136, 1982.

Christopher Manning, Prabhakar Raghavan, and Hinrich Schütze. Introduction to information retrieval. *Natural Language Engineering*, 16(1):100–103, 2010.

T. Milbich, K. Roth, B. Brattoli, and B. Ommer. Sharing matters for generalization in deep metric learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2020. doi: 10.1109/TPAMI.2020.3009620.

Timo Milbich, Karsten Roth, Homanga Bharadhwaj, Samarth Sinha, Yoshua Bengio, Björn Ommer, and Joseph Paul Cohen. Diva: Diverse visual feature aggregation for deep metric learning. *CoRR*, abs/2004.13458, 2020. URL https://arxiv.org/abs/2004.13458.

Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pp. 6706–6716. IEEE, 2020. doi: 10.1109/CVPR42600.2020.00674. URL https://doi.org/10.1109/CVPR42600.2020.00674.

Yair Movshovitz-Attias, Alexander Toshev, Thomas K Leung, Sergey Ioffe, and Saurabh Singh. No fuss distance metric learning using proxies. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 360–368, 2017.

Kevin Musgrave, Serge J. Belongie, and Ser-Nam Lim. A metric learning reality check. *CoRR*, abs/2003.08505, 2020. URL https://arxiv.org/abs/2003.08505.

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*, pp. 4004–4012, 2016.

Michael Opitz, Georg Waltner, Horst Possegger, and Horst Bischof. Deep metric learning with bier: Boosting independent embeddings robustly. *IEEE transactions on pattern analysis and machine intelligence*, 2018.

Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2019. doi: 10.1109/cvpr.2019.00409. URL http://dx.doi.org/10.1109/CVPR.2019.00409.

Wonpyo Park, Wonjae Kim, Kihyun You, and Minsu Cho. Diversified mutual learning for deep metric learning. 2020.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

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 (ICCV)*, October 2019.

Aniruddh Raghu, Maithra Raghu, Samy Bengio, and Oriol Vinyals. Rapid learning or feature reuse? towards understanding the effectiveness of MAML. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL https://openreview.net/forum?id=rkgMkCEtPB.

Jathushan Rajasegaran, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Mubarak Shah. Self-supervised knowledge distillation for few-shot learning. *CoRR*, abs/2006.09785, 2020. URL https://arxiv.org/abs/2006.09785.

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6550.

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

Karsten Roth, Timo Milbich, and Bjorn Ommer. Pads: Policy-adapted sampling for visual similarity learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020a.

Karsten Roth, Timo Milbich, Samarth Sinha, Prateek Gupta, Björn Ommer, and Joseph Paul Cohen. Revisiting training strategies and generalization performance in deep metric learning, 2020b.

Artsiom Sanakoyeu, Vadim Tschernezki, Uta Buchler, and Bjorn Ommer. Divide and conquer the embedding space for metric learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823, 2015.

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

Juan-Luis Suárez, Salvador García, and Francisco Herrera. A tutorial on distance metric learning: Mathematical foundations, algorithms and software. *CoRR*, abs/1812.05944, 2018. URL http://arxiv.org/abs/1812.05944.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. *CoRR*, abs/1910.10699, 2019. URL http://arxiv.org/abs/1910.10699.

Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B Tenenbaum, and Phillip Isola. Rethinking few-shot image classification: a good embedding is all you need? *arXiv preprint arXiv:2003.11539*, 2020.

C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The caltech-ucsd birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. *arXiv preprint arXiv:2005.10242*, 2020.

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 (CVPR)*, June 2019.

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*, pp. 2840–2848, 2017.

Hong Xuan, Richard Souvenir, and Robert Pless. Deep randomized ensembles for metric learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 723–734, 2018.

Hong Xuan, Abby Stylianou, and Robert Pless. Improved embeddings with easy positive triplet mining. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

Lu Yu, Vacit Oguz Yazici, Xialei Liu, Joost van de Weijer, Yongmei Cheng, and Arnau Ramisa. Learning metrics from teachers: Compact networks for image embedding. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2019. doi: 10.1109/cvpr.2019.00302. URL http://dx.doi.org/10.1109/CVPR.2019.00302.

Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *CoRR*, abs/1612.03928, 2016. URL http://arxiv.org/abs/1612.03928.

Andrew Zhai and Hao-Yu Wu. Making classification competitive for deep metric learning. *CoRR*, abs/1811.12649, 2018. URL http://arxiv.org/abs/1811.12649.

Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

Ying Zhang, Tao Xiang, Timothy M. Hospedales, and Huchuan Lu. Deep mutual learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

Wenzhao Zheng, Zhaodong Chen, Jiwen Lu, and Jie Zhou. Hardness-aware deep metric learning. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.