DiVA: Diverse Visual Feature Aggregation for Deep Metric Learning

Timo Milbich*[1], Karsten Roth*[1,2], Homanga Bharadhwaj[3,4], Samarth Sinha[2,4], Yoshua Bengio[2,5], Björn Ommer*[1], and Joseph Paul Cohen*[2]

[1] Heidelberg Collaboratory for Image Processing (HCI), Heidelberg University
[2] Mila, Universite de Montreal
[3] Vector Institute, Toronto Robotics Institute
[4] University of Toronto
[5] CIFAR Senior Fellow

Abstract. Visual similarity plays an important role in many computer vision applications. Deep metric learning (DML) is a powerful framework for learning such similarities which not only generalize from training data to identically distributed test distributions, but in particular also translate to *unknown* test classes. However, its prevailing learning paradigm is class-discriminative supervised training, which typically results in representations specialized in separating training classes. For effective generalization, however, such an image representation needs to capture a diverse range of data characteristics. To this end, we propose and study multiple complementary learning tasks, targeting conceptually different data relationships by only resorting to the available training samples and labels of a standard DML setting. Through simultaneous optimization of our tasks we learn a single model to aggregate their training signals, resulting in strong generalization and state-of-the-art performance on multiple established DML benchmark datasets.

Keywords: Deep Metric Learning, Generalization, Self-Supervision

1 Introduction

Many applications in computer vision, such as image retrieval [39, 33, 64] and face verification [53, 54], rely on capturing visual similarity, where approaches are commonly driven by Deep Metric learning (DML) [54, 64, 39]. These models aim to learn an embedding space which meaningfully reflects similarity between training images and, more importantly, generalizes to test classes which are *unknown* during training. Even though models are evaluated on transfer learning, the prevailing training paradigm in DML utilizes discriminative supervised learning. Consequently, the learned embedding space is specialized to features which help only in separating among training classes, and may not correctly translate to unseen test classes. Now, if supervised learning does not result in sufficient generalization, how can we exploit the available training data and class labels to

* Equal first and last authorship.

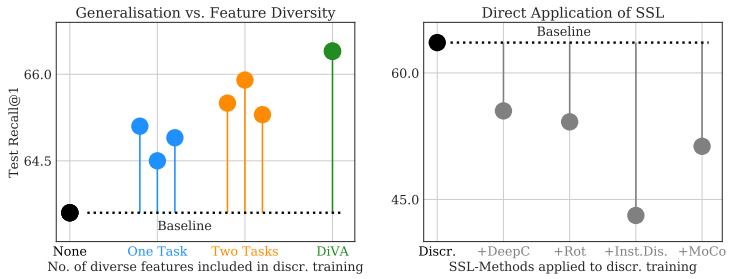

Fig. 1. *DML using diverse learning tasks vs. direct incorporation of self-supervision.* (Left) Generalization performance increases with each task added to training, independent of the exact combination of our proposed tasks (blue: one extra task, orange: two extra tasks, green: all tasks). (Right) Directly combining supervised learning with self-supervised learning techniques such as DeepC(luster) [7], Rot(Net) [18], Inst.Dis(crimination) [65] or Mo(mentum)Co(ntrast) [21] actually hurts DML generalization.

provide additional training signals beyond the standard discriminative task?
Recent breakthroughs in self-supervised learning have shown that contrastive image relations inferred from images themselves yield rich feature representations which even surpass the ability of supervised features to generalize to novel downstream task [43, 21, 9]. However, although DML typically also learns from image relations in the form of pairs [20], triplets [64, 53] or more general image tuples [42, 10], the complementary benefit of self-supervision in DML is largely unstudied. Moreover, the commonly available class assignments give rise to image relations aside from the standard, supervised learning task of 'pulling' samples with identical class labels together while 'pushing' away samples with different labels. As such ranking-based learning is not limited to discriminative training only, other relations can be exploited to learn beneficial data characteristics which so far have seen little coverage in DML literature.
In this paper, we tackle the issue of generalization in DML by designing diverse learning tasks complementing standard supervised training, leveraging only the commonly provided training samples and labels. Each of these tasks aims at learning features representing different relationships between our training classes and samples: *(i)* features discriminating among classes, *(ii)* features shared across different classes, *(iii)* features capturing variations within classes and *(iv)* features contrasting between individual images. Finally, we present how to effectively incorporate them in a unified learning framework. In our experiments we study mutual benefits of these tasks and show that joint optimization of diverse representations greatly improves generalization performance as shown in Fig. 1 (left), outperforming the state-of-the-art in DML. Our contributions can be summarized as follows:

– We design novel triplet learning tasks resulting in a diverse set of features and study their complementary impact on supervised DML.

- We adopt recent contrastive self-supervised learning to the problem of DML and extend it to effectively support supervised learning, as direct incorporation of self-supervised learning does not benefit DML (cf. Fig1) (right).
- We show how to effectively incorporate these learning tasks in a single model, resulting in state-of-the-art performance on standard DML benchmark sets.

2 Related Work

Deep Metric Learning. Deep Metric Learning is one of the primary frameworks for image retrieval [39, 33, 49, 64], zero-shot generalization [53, 49, 52, 2] or face verification [24, 34, 12]. It is also closely related to recent successful unsupervised representation learning approaches employing contrastive learning [21, 38, 9]. Commonly, DML approaches are formulated as ranking tasks on data tuples such as image pairs[20], triplets[24, 64], quadruplets[10] or higher order relations[54, 42, 63]. Effective training of these methods is typically promoted by tuple mining strategies alleviating the high sampling complexity, such as distance-based [53, 64], hierarchical [17] or learned [50]. Methods like ProxyNCA[39], Softtriple[12], Arcface[12] or Normalized Softmax[67] introduce learnable data proxies which represent entire subsets of the data, thus circumventing the tuple selection process. Orthogonally, DML research has started to pay more emphasis on the training process itself. This involves the generation of artificial training data [33, 69] or adversarial objectives [14]. MIC [49] explains away intra-class variance to strengthen the discriminative embedding. [52] propose to separate the input data space to learn subset-specific, yet still only class-discriminative representations similar to other ensemble methods [66, 28, 44, 37]. In contrast, we learn different embeddings on conceptually different tasks to capture diverse image features.

Self-supervised Representation Learning. Commonly, self-supervised representation learning aims to learn transferable feature representations from unlabelled data, and is typically applied as pre-training for downstream tasks[23, 36]. Early methods on representation learning are based on sample reconstructions [58, 30] which have been further extended by interpolation constraints [4] and generative adversarial networks [11, 15, 3, 13]. Further, introducing manually designed surrogate objectives encourage self-supervised models to learn about data-related properties. Such tasks range from predicting image rotations [18], solving a Jigsaw puzzle [40, 41, 6] to iteratively refining the initial network bias using clustering algorithms [7]. Recently, self-supervision approaches based on contrastive learning result in strong features performing close to or even stronger than supervised pretraining [21, 38, 9, 55] by leveraging invariance to realistic image augmentations. As the these approaches are essentially defined on pairwise image relations, they share common ground with ranking-based DML. In our work, we extend such a contrastive objective to effectively complement supervised DML training.

Multi-task Learning. Concurrently solving different tasks is also employed by classical multi-task learning which are often based on a divide-and-conquer principle with multiple learner optimizing a given subtask. [5] utilizes additional

training data and annotations to capture extra information, while our tasks are defined on standard training data only. [46] learn different classifiers for groups of entire categories, thus following a similar motivation as some DML approaches [44, 52]. The latter aims at learning more fine-grained, yet only discriminative features by reducing the data variance for each learner, thus being related to standard hard-negative mining [53]. In contrast, our work formulates various specific learning tasks to target *different* data characteristics of the training data.

3 Method

Let $f_i := f(I_i, \theta) \in \mathbb{R}^N$ be a N-dimensional encoding of an image $I_i \in \mathcal{I}$ represented by a deep neural network with parameters θ. Based on f_i, deep metric learning (DML) aims to learn image embeddings $\phi_i := \phi(f_i, \zeta) : \mathbb{R}^N \mapsto \mathbb{R}^D$ which allow to measure the similarity between images I_i, I_j as $d^{\phi}_{i,j} := d(\phi_i, \phi_j)$ under a predefined distance metric $d(\cdot, \cdot)$. Typically, $\phi(\cdot, \zeta)$ is a linear layer on the features f representation, parameterized by ζ and normalized to the real hypersphere $\mathbb{S}^{D-1}$ for regularization [64]. $d(\cdot, \cdot)$ is usually chosen to be the Euclidean distance. In standard supervised DML, ϕ is then optimized to reflect semantic similarity between images I_i defined by the corresponding class labels $y_i \in \mathcal{Y}$.
While there are many ways to define training objectives on ϕ, ranking losses, such as variants of the popular triplet loss [64, 54, 42], are a natural surrogate for the DML problem. Based on image triplets $t = \{I_a, I_p, I_n\}$ with I_a defined as anchor, I_p as a similar, positive and I_n as a negative image, we minimize

$$\mathcal{L}_{\text{tri}}(t) = [d^{\phi}_{a,p} - d^{\phi}_{a,n} + \gamma]_+ \,, \tag{1}$$

where $[\cdot]_+$ defines the hinge function which clips any negative value to zero. Hence, we maximize the gap between $d^{\phi}_{a,p}$ and $d^{\phi}_{a,n}$ as long as a margin γ is violated.
In supervised DML, $\mathcal{L}_{\text{tri}}(t)$ is typically optimized to discriminate between classes. Thus, f is trained to predominantly capture highly discriminative features while being invariant to image characteristics which do not facilitate training class separation. However, as we are interested in generalizing to unknown test distributions, we should rather aim at maximizing the amount of features captured from the training set $\mathcal{I}$, i.e. additionally formulate learning tasks which yield features beyond mere class-discrimination.
In order to formulate such tasks, we make use of the fact that triplet losses are instance-based. Thus, conceptually they are not restricted to class discrimination, but allow to learn *commonalities between the provided anchor I_a and positive I_p* compared to a negative sample I_n. Following, we will use this observation to also learn those commonalities which are neglected by discriminative training, such as commonalities shared between and within classes.

3.1 Diverse learning tasks for DML

We now introduce several tasks for learning a diverse set of features, resorting only to the standard training information provided in a DML problem. Each

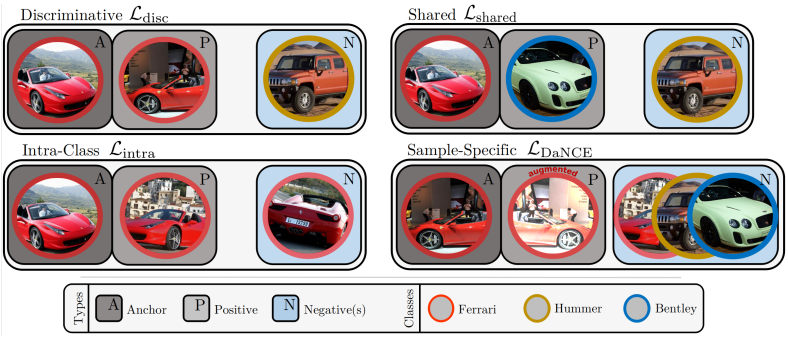

Fig. 2. *Schematic description of each task.* We learn four complementary tasks to capture features focusing on different data characteristics. The standard *class-discriminative* task which learning features separating between samples of different classes, the *shared* task which captures features relating samples across different classes, a *sample-specific* task to enforce image representations invariant to transformations and finally the *intra-class* task modelling data variations within classes.

of these tasks is designed to learn features which are conceptually neglected by the others to be mutually complementary. First, we introduce the intuition behind each feature type, before describing how to learn it based on pairwise or triplet-based image relations.

Class-discriminative features. These features are learned by standard class-discriminative optimization of ϕ and focus on data characteristics which allow to accurately separate one class from all others. It is the prevailing training signal of common classification-based [61, 12, 67], proxy-based [39, 47] or ranking-based [49, 44] approaches. For the latter, we can formulate the training task using Eq. 1 by means of triplets $\{I_a, I_p, I_n\} \in \mathcal{T}_{\text{disc}}$ with $y_a = y_p$ and $y_a \neq y_n$, as

$$\mathcal{L}_{\text{disc}} = \frac{1}{Z} \sum_{t \sim \mathcal{T}_{\text{disc}}} \mathcal{L}_{\text{tri}}(t) \ , \tag{2}$$

thus minimizing embedding distances between samples of the same class while maximizing it for samples of different classes. Moreover, the discriminative signal is important to learn how to aggregate features into classes, following the intuition of "the whole is more than the sum of its parts" analyzed in Gestalt theory [59].
Class-shared features. In contrast to discriminative features which look for characteristics *separating classes*, class-shared features capture commonalities, i.e variations, *shared across classes*. For instance, cars have a certain invariance towards color changes, thus being of little help when separating between them. However, to learn about this characteristic is actually beneficial when generalizing to other colorful object classes like flowers or fishes. Given suitable label information, learning such features would naturally follow the standard discriminative training setup. However, having only class labels available, we must resort to

approximations. To this end, we exploit the hypothesis that for most arbitrarily sampled triplets $\{I_a, I_p, I_n\} \in \mathcal{T}_{\text{shared}}$ with each constituent coming from mutually different classes, i.e. $y_a \neq y_p \neq y_n$, the anchor I_a and positive I_p share some common pattern when compared the negative image I_n. Commonalities which are frequently observed between classes y_a, y_p, will occur more often than noisy patterns which are unique to few t_{shared}, which is commonly observed when learning on imbalanced data [8, 16, 31]. Learning is then performed by optimizing

$$\mathcal{L}_{\text{shared}} = \frac{1}{Z} \sum_{t \sim \mathcal{T}_{\text{shared}}} \mathcal{L}_{\text{tri}}(t) \; . \tag{3}$$

As deep networks learn from frequent backpropagation of similar learning signals resulting in informative gradients, only prominent shared features are captured. Further, since shared features can be learned between any classes, we need to warrant diverse combinations of classes in our triplets $\mathcal{T}_{\text{shared}}$. Thus, enabling triplet constituents to be sampled from the whole embedding space ϕ using distance-based sampling [64] is crucial to avoid any bias towards samples which are mostly far (random sampling) or close (hard-negative sampling) to a given anchor I_a.

Intra-class features. The tasks defined so far model image relations across classes. In contrast, intra-class features describe variations within a given class. While these variations may also apply to other classes (thus exhibiting a certain overlap with class-shared features), more class-specific details are targeted. Hence, to capture such data characteristics by means of triplet constraints, we can follow a similar intuition as for learning class-shared features. Thus, to learn intra-class features, we define triplets by means of triplets $\{I_a, I_p, I_n\} \in \mathcal{T}_{\text{intra}}$ with $y_a = y_p = y_n$, i.e. each constituent coming from the same class, and minimize

$$\mathcal{L}_{\text{intra}} = \frac{1}{Z} \sum_{t \sim \mathcal{T}_{\text{intra}}} \mathcal{L}_{\text{tri}}(t) \; . \tag{4}$$

Sample-specific features. Recent approaches for self-supervised learning [43, 21, 1] based on noise contrastive estimation (NCE) [19] show that features exhibiting strong generalization for transfer learning can be learned only from training images themselves. As NCE learns to increase the correlation between embeddings of an anchor sample and a similar positive sample by constrasting against a set of negative samples, it naturally translates to DML. He et al. [21] proposed an efficient self-supervised framework which first applies data augmentation to generate positive surrogates $\tilde{I}_a$ for a given anchor I_a. Next, using NCE we contrast their embeddings $\phi_a := \phi(I_a, \zeta), \tilde{\phi}_a := \phi(\tilde{I}_a, \zeta)$ against randomly sampled negatives $I_n \in \mathcal{N} \subset \mathcal{I}$ by minimizing

$$\mathcal{L}_{\text{NCE}} = \frac{1}{Z} \sum_{I_a \sim \mathcal{I}} -\log \frac{\exp(\phi(I_a, \zeta)^\top \phi(\tilde{I}_a, \zeta)/\tau)}{\sum_{I_n \in \mathcal{N}} \exp(\phi(I_a, \zeta)^\top \phi(I_n, \zeta)/\tau)} \tag{5}$$

where the temperature parameter τ is adjusted during optimization to control the training signal, especially during earlier stages of training. By contrasting

each sample against many negatives, i.e. large sets $\mathcal{N}$, this task effectively yields a general, class-agnostic features description of our data. Moreover, as the contrastive objective explicitly increases the similarity of an anchor image with its augmentations, invariance against data transformations and scaling are learned. Fig. 2 summarizes and visually explains the different training objectives of each task.

3.2 Improved generalization by multi feature learning

Following we show how to efficiently incorporate the learning tasks introduced in the previous section into a single DML model. We first extend the objective Eq. 5 using established triplet sampling strategies for improved adjustment to DML, before we jointly train our learning tasks for maximal feature diversity.

Adapting noise contrastive estimation to DML. Efficient strategies for mining informative negatives I_n are a key factor [53] for successful training of ranking-based DML models. Since NCE essentially translates to a ranking between images $I_a, \tilde{I}_a, I_n$, its learning signal is also impaired if $I_n \in \mathcal{N}$ are uninformative, i.e. $d^{\phi}(I_a, I_n)$ being large. To this end, we control the contribution of each negative $I_n \in \mathcal{N}$ to $\mathcal{L}_{\text{NCE}}$ by a weight factor $w(d) = \min(\lambda, q^{-1}(d))$. Here, $q(d) = d^{D-2}\left[1 - \frac{1}{4}d^2\right]^{\frac{D-3}{2}}$ is the distribution of pairwise distances[1] on the D-dimensional unit hypersphere $\mathbb{S}^{D-1}$ and λ a cut-off parameter. Similar to [64], $w(d)$ helps to equally weigh negatives from the whole range of possible distances in ϕ and, in particular, increases the impact of harder negatives. Thus, our distance-adapted NCE loss becomes

$$\mathcal{L}_{\text{DaNCE}} = \frac{1}{Z} \sum_{I_a \sim \mathcal{I}} -\log \frac{\exp(\phi(I_a)^\top \phi(\tilde{I}_a)/\tau)}{\sum_{I_n \in \mathcal{N}} \exp(w(d^{\phi}_{a,n}) \cdot \phi^*(I_n)^\top \phi(I_a)/\tau)} \,. \tag{6}$$

NCE-based objectives learn best using large sets of negatives [21]. However, naively utilizing only negatives from the current mini-batch constrains $\mathcal{N}$ to the available GPU memory. To alleviate this limitation, we follow [21] and realize $\mathcal{N}$ as a large memory queue, which is constantly updated with embeddings $\phi^*(\tilde{I}_a)$ from training iteration t by utilising the running-average network $\phi^{*,t+1} = \mu\phi^{*,t} + (1-\mu)\phi^t$.

Joint optimization for maximal feature diversity. The tasks presented in Sec. 3.1 are formulated to extract mutually complementary information from our training data. In order to capture their learned features in a single model to obtain a rich image representation, we now discuss how to jointly optimize these tasks.
While each task targets a semantically different concept of features, their driving learning signals are based on potentially contradicting ranking constraints on

[1] To compute d, we use the euclidean distance between samples. Since ϕ is regularized to the unit hypersphere $\mathbb{S}^{D-1}$, the euclidean distance correlates with cosine distance.

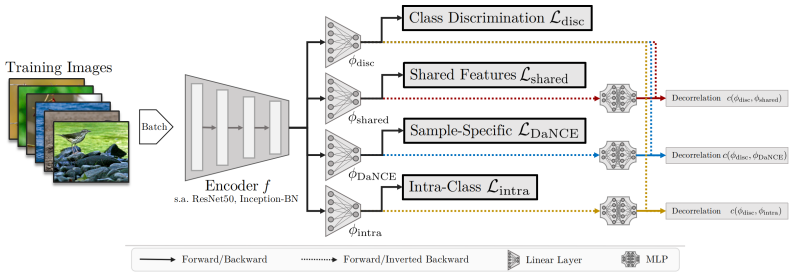

Fig. 3. *Architecture of our propose model.* Each task $\mathcal{L}_\bullet$ optimizes an individual embedding $\phi_\bullet$ implemented as a linear layer with a shared underlying feature encoder f. Pairwise decorrelation $c(\cdot,\cdot)$ of the embeddings utilizing the mapping ψ based on a two-layer MLP encourages each task to further emphasize on its targeted data characteristics. Gradient inversion R is applied during the backward pass to each embedding head.

the learned embedding space. Thus, aggregating these signals to optimizing a joint, single embedding function ϕ may entail detrimental interference between them. In order to circumvent this issue, we learn a dedicated embedding space for each task, as often conducted in multi-task optimization [49, 48], i.e. $\phi_{\text{disc}}(f), \phi_{\text{shared}}(f), \phi_{\text{intra}}(f)$ and $\phi_{\text{nce}}(f)$ with $\phi_\bullet(f) : \mathbb{R}^N \mapsto \mathbb{R}^D$ (cf. Sec. 3). As all embeddings share the same feature extractor f, each task still benefits from the aggregated learning signals. Additionally, as there may still be redundant overlap in the information captured by each task, we mutually decorrelate these representations, thus maximizing the diversity of the overall training signal. Similar to [44, 49] we minimize the mutual information of two embedding functions ϕ^a, ϕ^b by maximizing their correlation c in the embedding space of ϕ^b, followed by a gradient reversal. For that, we learn a mapping $\psi : \mathbb{R}^D \mapsto \mathbb{R}^D$ from ϕ_i^a to ϕ_i^b given an image I_i and compute the correlation $c(\phi_i^a, \phi_i^b) = \|(R(\phi_i^a) \odot \psi(R(\phi_i^b)))\|_2^2$ with $\odot$ being the point-wise product. R denotes a gradient reversal operation, which inverts the resulting gradients during backpropagation. Maximizing c results in ψ aiming to make ϕ^a and ϕ^b comparable. However, through the subsequent gradients reversal, we actually decorrelate the embedding functions. Joint training of all tasks is finally performed by minimizing

$$\mathcal{L} = \mathcal{L}_{\text{disc}} + \alpha_1 \mathcal{L}_{\text{shared}} + \alpha_2 \mathcal{L}_{\text{intra}} + \alpha_3 \mathcal{L}_{\text{DaNCE}} - \sum_{(\phi_a, \phi_b) \in \mathcal{P}} \rho_{a,b} \cdot c(\phi_a, \phi_b) \quad (7)$$

where $\mathcal{P}$ denotes the pairs of embeddings to be decorrelated. We found

$$\mathcal{P} = \{(\phi_{\text{disc}}, \phi_{\text{DaNCE}}), (\phi_{\text{disc}}, \phi_{\text{shared}}), (\phi_{\text{disc}}, \phi_{\text{intra}})\} \,, \quad (8)$$

to work best, which decorrelates the auxiliary tasks with the class-discriminative task. Initial experiments showed that further decorrelation $c(\bullet, \bullet)$ among the auxiliary tasks does not result in further benefit and is therefore disregarded. The weighting parameters $\rho_{a,b}$ adjusting the degree of decorrelation between the

embeddings are set to the same, constant value in our implementation. Fig. 3 provides an overview of our model. Finally, we combine our learned embedding to form an ensemble representation to fully make use of all information.

Computational costs. We train all tasks using the same mini-batch to avoid computational overhead. While optimizing each learner on an individual batch can further alleviate training signal interference [52, 66], training time increases significantly. Using a single batch per iteration, we minimize the required extra computations to the extra forward pass through ϕ^* (however without computing gradients) for contrasting against negatives sampled from the memory queue as well as the small mapping networks ψ. Across datasets, we measure an increase in training time by $10-15\%$ per epoch compared to training a standard supervised DML task. This is comparable to or lower than other methods, which perform a full clustering on the dataset [49, 52] after each epoch, compute extensive embedding statistics [26] or simultaneously train generative models [33].

4 Experiments

Following we first present our implementation details and the benchmark datasets. Next, we evaluate our proposed model and study how our learning tasks complement each other and improve over baseline performances. Finally, we discuss our results in the context of the current state-of-the-art and conduct analysis and ablation experiments.

Implementation details. We follow the common training protocol of [64, 49, 52] for implementations utilizing a ResNet50-backbone. The shorter image axis is resized to 256, followed by a random crop to 224×224 and a random horizontal flip with $p = 0.5$. During evaluation, only a center crop is taken after resizing. The embedding dimension is set to $D = 128$ for each task embedding. For model variants using the Inception-V1 with Batch-Normalization[25], we follow [63, 26] and use $D = 512$. Resizing, cropping and flipping is done in the same way as for ResNet50 versions. The implementation is done using the PyTorch framework[45], and experiments are performed on compute clusters containing NVIDIA Titan X, Tesla V4, P100 and V100, always limited to 12GB VRAM following the standard training protocol [64]. For DiVA, we utilise the triplet-based margin loss [64] with fixed margin $\gamma = 0.2$ and $\beta = 1.2$ and fixed temperature $\tau = 0.1$.

Hyperparameters. For training, we use Adam[29] with learning rate 10^{-5} and a weight decay of $5 \cdot 10^{-4}$. For ablations, we use no learning rate scheduling, while our final model is trained using scheduling values determined by cross-validation. We train for 150 epochs. Our joint training framework can be adjusted by setting the de-correlation weights $\rho_{a,b}$ and weight parameters α_i. In both cases we utilize the same values for all learning task, thus we effectively only adjust two parameters $[\rho, \alpha]$ to each benchmark sets: CUB200-2011 (IBN: $[300, 0.15]$, R50 $[1500, 0.3]$), CARS196 (IBN: $[100, 0.15]$, R50 $[100, 0.1]$), SOP (IBN: $[150, 0.2]$, R50 $[150, 0.2]$). This is comparable to other approaches, e.g. MS [63], SoftTriple [47], D&C [52]. As our auxiliary embeddings generalize better, they are more emphasized for computing $d_{i,j}$ during testing (e.g. double on CUB200 and CARS196).

Dataset →		CUB200-2011[60]			CARS196[32]			SOP[42]		
Approach ↓	Dim	R@1	R@2	NMI	R@1	R@2	NMI	R@1	R@10	NMI
Margin[64] (orig, R50)	128	63.6	74.4	69.0	79.6	86.5	69.1	72.7	86.2	90.7
Margin[64] (ours, IBN)	512	63.6	74.7	68.3	79.4	86.6	66.2	76.6	89.2	89.8
DiVA (IBN, D & Da)	512	64.5	76.0	68.8	80.4	87.7	67.2	77.0	89.4	**90.1**
DiVA (IBN, D & S)	512	65.1	76.4	69.0	81.5	88.3	66.8	77.2	89.6	90.0
DiVA (IBN, D & I)	512	64.9	75.8	68.4	80.6	87.9	67.4	76.9	89.4	89.9
DiVA (IBN, D & Da & I)	510	65.3	76.5	68.3	82.2	89.1	67.8	75.8	89.0	89.8
DiVA (IBN, D & S & I)	510	65.5	76.4	68.4	82.1	89.4	67.2	77.0	89.3	89.7
DiVA (IBN, D & Da & S)	510	65.9	76.7	68.9	82.6	89.6	68.0	77.4	89.6	90.1
DiVA (IBN, D & Da & S & I)	512	**66.4**	**77.2**	**69.6**	**83.1**	**90.0**	**68.1**	**77.5**	**90.3**	**90.1**

Table 1. *Comparison of different combinations of learning tasks.* I(nception-V1) B(atch-)N(ormalization), and R(esNet)50 denote the backbone architecture. No learning rate scheduling is used. Our tasks are denoted by D(*iscriminative*), S(*hared*), I(*ntra-Class*) & and Da(*NCE*). The dimensionality per task embedding depends on the number of tasks used, totalling in $D = 512$ (two tasks use 256 each, three 170, four 128.

Datasets. We evaluate the performance on three common benchmark datasets with standard training/test splits (see e.g. [64, 49, 52, 63]): *CARS196*[32], which contains 16,185 images from 196 car classes. The first 98 classes containing 8054 images are used for training, while the remaining 98 classes with 8131 images are used for testing. *CUB200-2011*[60] with 11,788 bird images from 200 classes. Training/test sets contain the first/last 100 classes with 5864/5924 images respectively. *Stanford Online Products (SOP)*[42] provides 120,053 images divided in 22,634 product classes. 11318 classes with 59551 images are used for training, while the remaining 11316 classes with 60502 images are used for testing.

4.1 Performance study of multi-feature DML

We now compare our model and the complementary benefit of our proposed feature learning tasks for supervised DML. Tab. 1 evaluates the performance of our model based on margin loss [64], a triplet based objective with an additionally learnable margin, and distance-weighted triplet sampling [64]. We use Inception-V1 with Batchnorm and a maximal aggregated embedding dimensionality of 512. Thus, if two tasks are utilized, each embedding has $D = 256$, in case of three tasks 170 and four tasks result in $D = 128$. No learning rate scheduling is used. Evaluation is conducted on CUB200-2011 [60], CARS196 [32] and SOP [42]. Retrieval performance is measured through Recall@k[27] and clustering quality via Normalized Mutual Information (NMI) [35]. While our results vary between possible task combinations, we observe that the generalization of our model consistently increases with each task added to the joint optimization. Our strongest model including all proposed tasks improves the generalization performance by 2.8% on CUB200-2011, 3.7% on CARS196 and 0.9% on SOP. This highlights that *(i)* purely discriminative supervised learning disregards valuable training

Dataset →		CUB200-2011[60]			CARS196[32]			SOP[42]		
Approach ↓	Dim	R@1	R@2	NMI	R@1	R@2	NMI	R@1	R@2	NMI
HTG[68]	512	59.5	71.8	-	76.5	84.7	-	-	-	-
HDML[69]	512	53.7	65.7	62.6	79.1	87.1	69.7	68.7	83.2	89.3
Margin[64]	128	63.6	74.4	69.0	79.6	86.5	69.1	72.7	86.2	90.8
HTL[17]	512	57.1	68.8	-	81.4	88.0	-	74.8	88.3	-
DVML[33]	512	52.7	65.1	61.4	82.0	88.4	67.6	70.2	85.2	90.8
MultiSim[63]	512	65.7	77.0	-	84.1	90.4	-	78.2	90.5	-
D&C[52]	128	65.9	76.6	69.6	84.6	90.7	70.3	75.9	88.4	90.2
MIC[49]	128	66.1	76.8	69.7	82.6	89.1	68.4	77.2	89.4	90.0
Significant increase in network parameter:										
HORDE[26]+Contr.[20]	512	66.3	76.7	-	83.9	90.3	-	-	-	-
Softtriple[47]	512	65.4	76.4	-	84.5	90.7	70.1	78.3	90.3	**92.0**
Ensemble Methods:										
A-BIER[44]	512	57.5	68.7	-	82.0	89.0	-	74.2	86.9	-
Rank[62]	1536	61.3	72.7	66.1	82.1	89.3	71.8	**79.8**	**91.3**	90.4
DREML[66]	9216	63.9	75.0	67.8	86.0	91.7	**76.4**	-	-	-
ABE[28]	512	60.6	71.5	-	85.2	90.5	-	76.3	88.4	-
Inception-BN										
Ours (Trip-DiVA-IBN-512)	512	66.7	77.1	69.3	83.1	90.3	68.8	76.9	88.9	89.4
Ours (DiVA-IBN-512)	512	66.8	77.7	70.0	84.1	90.7	68.7	78.1	90.6	90.4
ResNet50										
Ours (Margin[64]-R50-512)	512	64.4	75.4	68.4	82.2	89.0	68.1	78.3	90.0	90.1
Ours (Trip-DiVA-R50-512)	512	68.5	78.5	**71.1**	**87.3**	**92.8**	72.1	79.4	90.8	90.3
Ours (DiVA-R50-512)	512	**69.2**	**79.3**	**71.4**	**87.6**	**92.9**	72.2	**79.6**	**91.2**	90.6

Table 2. *Comparison to the state-of-the-art methods on CUB200-2011*[60], *CARS196*[32] *and SOP*[42]. DiVA-Arch-Dim describes the backbone used with DiVA (IBN: Inception-V1 with Batchnorm, R50: ResNet50) and the total training and testing embedding dimensionality. For fair comparison, we also ran a standard ResNet50 with embedding dimensionality of 512. Trip-DiVA-Arch-Dim indicates standard triplet loss as base objective.

information and *(ii)* carefully designed learning tasks are able to capture this information for improved generalization to unknown test classes. We further analyze our observations in the ablation experiments.

4.2 Comparison to state-of-the-art approaches

Next, we compare our model using fixed learning rate schedules per benchmark to the current state-of-the-art approaches in DML. For fair comparison to the different methods, we report result both using Inception-BN (IBN) and ResNet50 (R50) as backbone architecture. As Inception-BN is typically trained with embedding dimensionality of 512, we restrict each embedding to $D = 128$ for direct comparison with non-ensemble methods. Thus we deliberately impair the potential of our model due to a significantly lower capacity per task, compared to the standard $D = 512$. For comparison with ensemble approaches and maximal performance, we use a ResNet50 [64, 49, 52] architecture and the corresponding standard dimensionality $D = 128$ per task. Fig. 2 summarizes

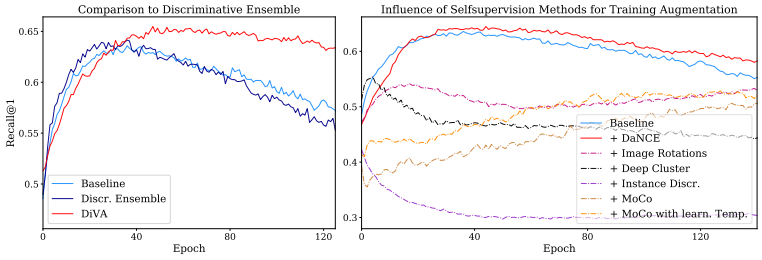

Fig. 4. *Analysis of complementary tasks for supervised learning.* (left): Performance comparison between class-dicsriminative training only (Baseline), ensemble of class-discriminative learners (Discr. Ensemble) and our proposed DiVA, which exhibits a large boost in performance. (right): Evaluation of self-supervised learning approaches combined with standard discriminative DML.

our results using both standard triplet loss [53] and margin loss [64] as our base training objective. In both cases, we significantly improve over methods with comparable backbone architectures and achieve new state-of-the-art results with our ResNet50-ensemble. In particular we outperform the strongest ensemble methods, including DREML [66] which utilize a much higher total embedding dimensionality. The large improvement is explained by the diverse and mutually complementary learning signals contributed by each task in our ensemble. In contrast, previous ensemble methods rely on the same, purely class-discriminative training signal for each learner. Note that some approaches strongly differ from the standard training protocols and architectures, resulting in more parameters and much higher GPU memory consumption, such as Rank [62] (32GB), ABE [28] (24GB), Softtriple [47] and HORDE [26]. Additionally, Rank [62] employs much larger batch-sizes to increase the number of classes per batch. This is especially crucial on the SOP dataset, which greatly benefits from higher class coverage due to its vast amount of classes [51]. Nevertheless, our model outperforms these methods - in some cases even in its constrained version (IBN-512).

4.3 Ablation Studies

In this section we conduct ablation experiments for various parts of our model. For every ablation we again use the Inception-BN network. The dimensionality setting follows the performance study in sec. 4.1. Again, we train each model with a fixed learning rate for fair comparison among ablations.

Influence of distance-adaption in DaNCE. To evaluate the benefit of our extension from $\mathcal{L}_{\text{nce}}$ [21, 19] to $\mathcal{L}_{\text{DaNCE}}$, we compare both versions in combination with standard supervised DML (i.e. class-discriminative features) in Fig. 4 (right). Our experiment indicates two positive effects: *(i)* The training convergence with our extended objective is much faster and *(ii)* the performance differs greatly between employing $\mathcal{L}_{\text{nce}}$ and $\mathcal{L}_{\text{DaNCE}}$. In fact, using the standard NCE objective is even detrimental to learning, while our extended version improves over the only discriminatively trained baseline. We attribute this to both the slow convergence

Methods →	Baseline	DiVA	No De-correlation	Separated models
Recall@1 →	63.6	66.4	65.6	48.7

Table 3. *Ablation studies.* We compare standard margin loss as baseline and DiVA performance against ablations of our model: no decorrelation between embeddings (No-Decorrelation.) and training an independent model for each task (Separated models). Total embedding dimensionality is 512.

of $\mathcal{L}_{\text{nce}}$ which is not able to support the faster discriminative learning and to emphasizing harder negatives in $\mathcal{L}_{\text{DaNCE}}$. In particular the latter is an important factor in ranking based DML [53], as during training more and more negatives become uninformative. To tackle this issue, we also experimented with learning the temperature parameter τ. While convergence speed increases slightly, we find no significant benefit in final generalization performance.

Evaluation of self-supervision methods. Fig. 4 (right) compares DaNCE to other methods from self-supervised representation learning. For that purpose we train the discriminative task with either DeepCluster [7], RotNet [18] or Instance Discrimination [65]. We observe that neither of these tasks is able to provide complementary information to improve generalization. DeepCluster, trained with 300 pseudo classes for classification, actually aims at approximating the class-discriminative learning signal while RotNet is strongly dependent on the variance of the training classes and converges very slowly. Instance discrimination seems to provide a contradictory training signal to the supervised task. These results are in line with previous works [22] which report difficulties to directly combine both supervised and self-supervised learning for improved test performance. In contrast, we explicitly adapt NCE to DML in our proposed objective DaNCE.

Comparison to purely class-discriminative ensemble. We now compare DiVA to an ensemble of class-discriminative learner (Discr. Ensemble) based on the same model architecture using embedding decorrelation in Fig. 4 (left). While the discriminative ensemble improves over the baseline, the amount of captured data information eventually saturates and, thus, performs significantly worse compared to our multi-feature DiVA ensemble. Further, our ablation reveals that joint optimization of diverse learning tasks regularizes training and reduces overfitting effects which eventually occur during later stages of DML training.

Benefit of task decorrelation. The role of decorrelating the embedding representations of each task during learning is analyzed by comparison to a model trained without this constraint. Firstly, Tab. 3 demonstrates that omitting the decorrelation still outperforms the standard margin loss ('Baseline') by 2.1% while operating on the same total embedding dimensionality. This proves that learning diverse features significantly improves generalization. Adding the de-corralation constraint then additionally boosts performance by 1.2%, as now each task is further encouraged to capture distinct data characteristics.

Learning without feature sharing. To highlight the importance of feature sharing among our learning tasks, we train an individual, independent model for the class-discriminative, class-shared, sample-specific and intra-class task. At

testing time, we combine their embeddings similar to our proposed model. Tab. 3 shows a dramatic drop in performance to 48% for the disconnected ensemble ('Separately Trained'), proving that sharing the complementary information captured from different data characteristics is crucial and mutually benefits learning. Without the class-discriminative signal, the other tasks lack the concept of an object class, which hurts the aggregation of embeddings (cf. Sec. 3).

Generalization and embedding space compression. Recent work [51] links DML generalization to a decreased compression [56] of the embedding space. Their findings report that the number of directions with significant variance [57, 51] of a representation correlates with the generalization ability in DML. To this end, we analyze our model using their proposed spectral decay ρ (lower is better) which is computed as the KL-divergence between the normalized singular value spectrum and a uniform distribution. Fig. 5 compares the spectral decays of our model and a standard supervised baseline model. As expected, due to the diverse information captured, our model learns a more complex representation which results in a significantly lower value of ρ and better generalization.

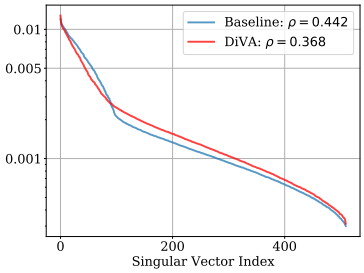

Fig. 5. *Singular Value Spectrum.* We analyze the singular value spectrum of DiVA embeddings and that of a network trained with the standard discriminative task. We find that our gains in generalization performance (Tab. 2, 1) are reflected by a reduced spectral decay [51] for our learned embedding space.

5 Conclusion

In this paper we propose several learning tasks which complement the class-discriminative training signal of standard, supervised Deep Metric Learning (DML) for improved generalization to unknown test distributions. Each of our tasks is designed to capture different characteristics of the training data: class-discriminative, class-shared, intra-class and sample-specific features. For the latter, we adapt contrastive self-supervised learning to the needs of supervised DML. Jointly optimizing all tasks results in a diverse overall training signal which is further amplified by mutual decorrelation between the individual tasks. Unifying these distinct representations greatly boosts generalization over purely discriminatively trained models.

Acknowledgements This work has been supported by hardware donations from NVIDIA (DGX-1), resources from Compute Canada, in part by Bayer AG and the German federal ministry BMWi within the project "KI Absicherung".

References

1. Bachman, P., Hjelm, R.D., Buchwalter, W.: Learning representations by maximizing mutual information across views. arXiv preprint arXiv:1906.00910 (2019)
2. Bautista, M.A., Sanakoyeu, A., Tikhoncheva, E., Ommer, B.: Cliquecnn: Deep unsupervised exemplar learning. In: Advances in Neural Information Processing Systems. pp. 3846–3854 (2016)
3. Belghazi, M.I., Rajeswar, S., Mastropietro, O., Rostamzadeh, N., Mitrovic, J., Courville, A.: Hierarchical adversarially learned inference (2018)
4. Berthelot, D., Raffel, C., Roy, A., Goodfellow, I.: Understanding and improving interpolation in autoencoders via an adversarial regularizer (2018)
5. Bhattarai, B., Sharma, G., Jurie, F.: Cp-mtml: Coupled projection multi-task metric learning for large scale face retrieval. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4226–4235 (2016)
6. Büchler, U., Brattoli, B., Ommer, B.: Improving spatiotemporal self-supervision by deep reinforcement learning. In: Proceedings of the European Conference on Computer Vision (ECCV) (2018)
7. Caron, M., Bojanowski, P., Joulin, A., Douze, M.: Deep clustering for unsupervised learning of visual features. European Conference on Computer Vision (2018)
8. Chawla, N.V., Japkowicz, N., Kotcz, A.: Editorial: Special issue on learning from imbalanced data sets. SIGKDD Explor. Newsl. **6** (Jun 2004)
9. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. arXiv preprint arXiv:2002.05709 (2020)
10. Chen, W., Chen, X., Zhang, J., Huang, K.: Beyond triplet loss: a deep quadruplet network for person re-identification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017)
11. Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P.: Infogan: Interpretable representation learning by information maximizing generative adversarial nets (2016)
12. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: Arcface: Additive angular margin loss for deep face recognition (2018)
13. Donahue, J., Krähenbühl, P., Darrell, T.: Adversarial feature learning (2016)
14. Duan, Y., Zheng, W., Lin, X., Lu, J., Zhou, J.: Deep adversarial metric learning. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)
15. Dumoulin, V., Belghazi, I., Poole, B., Mastropietro, O., Lamb, A., Arjovsky, M., Courville, A.: Adversarially learned inference (2016)
16. Gautheron, L., Morvant, E., Habrard, A., Sebban, M.: Metric learning from imbalanced data (2019)
17. Ge, W.: Deep metric learning with hierarchical triplet loss. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 269–285 (2018)
18. Gidaris, S., Singh, P., Komodakis, N.: Unsupervised representation learning by predicting image rotations. In: International Conference on Learning Representations (2018)
19. Gutmann, M., Hyvärinen, A.: Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (2010)
20. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2006)

21. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning (2019)
22. Hendrycks, D., Mazeika, M., Kadavath, S., Song, D.: Using self-supervised learning can improve model robustness and uncertainty. CoRR **abs/1906.12340** (2019), http://arxiv.org/abs/1906.12340
23. Hsu, K., Levine, S., Finn, C.: Unsupervised learning via meta-learning (2018)
24. Hu, J., Lu, J., Tan, Y.: Discriminative deep metric learning for face verification in the wild. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition (2014)
25. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. International Conference on Machine Learning (2015)
26. Jacob, P., Picard, D., Histace, A., Klein, E.: Metric learning with horde: High-order regularizer for deep embeddings. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
27. Jegou, H., Douze, M., Schmid, C.: Product quantization for nearest neighbor search. IEEE transactions on pattern analysis and machine intelligence **33**(1), 117–128 (2011)
28. Kim, W., Goyal, B., Chawla, K., Lee, J., Kwon, K.: Attention-based ensemble for deep metric learning. In: Proceedings of the European Conference on Computer Vision (ECCV) (2018)
29. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization (2015)
30. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: Proceedings of the International Conference on Learning Representations (ICLR) (2013)
31. Konno, T., Iwazume, M.: Cavity filling: Pseudo-feature generation for multi-class imbalanced data problems in deep learning (2018)
32. Krause, J., Stark, M., Deng, J., Fei-Fei, L.: 3d object representations for fine-grained categorization. In: Proceedings of the IEEE International Conference on Computer Vision Workshops. pp. 554–561 (2013)
33. Lin, X., Duan, Y., Dong, Q., Lu, J., Zhou, J.: Deep variational metric learning. In: The European Conference on Computer Vision (ECCV) (September 2018)
34. Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B., Song, L.: Sphereface: Deep hypersphere embedding for face recognition. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
35. Manning, C., Raghavan, P., Schütze, H.: Introduction to information retrieval. Natural Language Engineering **16**(1), 100–103 (2010)
36. Milbich, T., Ghori, O., Diego, F., Ommer, B.: Unsupervised representation learning by discovering reliable image relations. Pattern Recognition (PR) (2020)
37. Milbich, T., Roth, K., Brattoli, B., Ommer, B.: Sharing matters for generalization indeep metric learning. IEEE Transactions on Pattern Analysis and Machine Intelligence (2020)
38. Misra, I., van der Maaten, L.: Self-supervised learning of pretext-invariant representations (2019)
39. Movshovitz-Attias, Y., Toshev, A., Leung, T.K., Ioffe, S., Singh, S.: No fuss distance metric learning using proxies. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 360–368 (2017)
40. Noroozi, M., Favaro, P.: Unsupervised learning of visual representations by solving jigsaw puzzles. In: European Conference on Computer Vision. pp. 69–84. Springer (2016)

41. Noroozi, M., Vinjimoor, A., Favaro, P., Pirsiavash, H.: Boosting self-supervised learning via knowledge transfer. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 9359–9367 (2018)
42. Oh Song, H., Xiang, Y., Jegelka, S., Savarese, S.: Deep metric learning via lifted structured feature embedding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4004–4012 (2016)
43. Oord, A., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding (2018), `arXivpreprintarXiv:1807.03748`
44. Opitz, M., Waltner, G., Possegger, H., Bischof, H.: Deep metric learning with bier: Boosting independent embeddings robustly. IEEE Transactions on Pattern Analysis and Machine Intelligence (2018)
45. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch. In: NIPS-W (2017)
46. Pu, J., Jiang, Y.G., Wang, J., Xue, X.: Which looks like which: Exploring inter-class relationships in fine-grained visual categorization. In: Proceedings of the IEEE European Conference on Computer Vision (ECCV). pp. 425–440 (2014)
47. Qian, Q., Shang, L., Sun, B., Hu, J., Li, H., Jin, R.: Softtriple loss: Deep metric learning without triplet sampling (2019)
48. Ren, S., He, K., Girshick, R.B., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: NeuRips
49. Roth, K., Brattoli, B., Ommer, B.: Mic: Mining interclass characteristics for improved metric learning. In: The IEEE International Conference on Computer Vision (ICCV) (October 2019)
50. Roth, K., Milbich, T., Ommer, B.: Pads: Policy-adapted sampling for visual similarity learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
51. Roth, K., Milbich, T., Sinha, S., Gupta, P., Ommer, B., Cohen, J.P.: Revisiting training strategies and generalization performance in deep metric learning. In: Proceedings of the International Conference on Machine Learning (ICML) (2020)
52. Sanakoyeu, A., Tschernezki, V., Buchler, U., Ommer, B.: Divide and conquer the embedding space for metric learning. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
53. Schroff, F., Kalenichenko, D., Philbin, J.: 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)
54. Sohn, K.: Improved deep metric learning with multi-class n-pair loss objective. In: Advances in Neural Information Processing Systems. pp. 1857–1865 (2016)
55. Tian, Y., Krishnan, D., Isola, P.: Contrastive multiview coding. arXiv preprint arXiv:1906.05849 (2019)
56. Tishby, N., Zaslavsky, N.: Deep learning and the information bottleneck principle (2015)
57. Verma, V., Lamb, A., Beckham, C., Najafi, A., Mitliagkas, I., Courville, A., Lopez-Paz, D., Bengio, Y.: Manifold mixup: Better representations by interpolating hidden states (2018)
58. Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.A.: Extracting and composing robust features with denoising autoencoders (2008)
59. Wagemans, J., Elder, J.H., Kubovy, M., Palmer, S.E., Peterson, M.A., Singh, M., von der Heydt, R.: A century of gestalt psychology in visual perception: I. perceptual grouping and figure–ground organization. Psychological bulletin **138**(6), 1172 (2012)

60. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The caltech-ucsd birds-200-2011 dataset (2011)
61. Wang, J., Zhou, F., Wen, S., Liu, X., Lin, Y.: Deep metric learning with angular loss. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2593–2601 (2017)
62. Wang, X., Hua, Y., Kodirov, E., Hu, G., Garnier, R., Robertson, N.M.: Ranked list loss for deep metric learning. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
63. Wang, X., Han, X., Huang, W., Dong, D., Scott, M.R.: Multi-similarity loss with general pair weighting for deep metric learning. CoRR **abs/1904.06627** (2019), http://arxiv.org/abs/1904.06627
64. Wu, C.Y., Manmatha, R., Smola, A.J., Krahenbuhl, P.: Sampling matters in deep embedding learning. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2840–2848 (2017)
65. Wu, Z., Xiong, Y., Yu, S., Lin, D.: Unsupervised feature learning via non-parametric instance-level discrimination (2018)
66. Xuan, H., Souvenir, R., Pless, R.: Deep randomized ensembles for metric learning. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 723–734 (2018)
67. Zhai, A., Wu, H.Y.: Classification is a strong baseline for deep metric learning (2018)
68. Zhao, Y., Jin, Z., Qi, G.j., Lu, H., Hua, X.s.: An adversarial approach to hard triplet generation. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 501–517 (2018)
69. Zheng, W., Chen, Z., Lu, J., Zhou, J.: Hardness-aware deep metric learning. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)