Characterizing Generalization under Out-Of-Distribution Shifts in Deep Metric Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

Deep Metric Learning (DML) aims to find representations suitable for zero-shot transfer to *a priori* unknown test distributions. However, common evaluation protocols only test a single, fixed data split in which train and test classes are assigned randomly. More realistic evaluations should consider a broad spectrum of distribution shifts with potentially varying degree and difficulty. In this work, we systematically construct train-test splits of increasing difficulty and present the ***ooDML*** benchmark to characterize generalization under ***o***ut-***o***f-distribution shifts in ***DML***. *ooDML* is designed to probe the generalization performance on much more challenging, diverse train-to-test distribution shifts. Based on our new benchmark, we conduct a thorough empirical analysis of state-of-the-art DML methods. We find that while generalization tends to consistently degrade with difficulty, some methods are better at retaining performance as the distribution shift increases. Finally, we propose few-shot DML as an efficient way to consistently improve generalization in response to unknown test shifts presented in *ooDML*.

1 Introduction

Image representations that generalize well are the foundation of numerous computer vision tasks, such as image and video retrieval [58, 68, 52, 36, 1], face (re-)identification [54, 33, 8] and image classification [62, 4, 19, 38]. In practice, these representations should not only capture data within the training distribution, but also transfer to new, out-of-distribution (OOD) data. However, achieving effective OOD generalization is more challenging than in-distribution [27, 12, 21, 47, 30]. In the case of zero-shot generalization, where train and test classes are completely distinct, Deep Metric Learning (DML) successfully learns metric representation spaces that capture and transfer visual similarity to test distributions of unknown and unspecified shift *a priori*. However, while current DML benchmark sets are split into disjoint train and test classes, these are assigned randomly, with evaluation protocols in DML typically only utilising a *single, predefined and fixed* data split [68, 8, 58, 24, 49, 35, 52, 40]. Thus, *(i)* generalization is only evaluated on a fixed problem difficulty, *(ii)* generalization difficulty is only implicitly defined by the arbitrary data split, *(iii)* the distribution shift is not measured and *(iv)* cannot be not changed. As a result, proposed models can be overly adjusted to these singular evaluation settings, which puts into question the true zero-shot generalization capabilities of proposed DML models.

In this work, we first construct a new benchmark *ooDML* to characterize *g*eneralization under *o*ut-*o*f-distribution shifts in *DML*. We systematically build *ooDML* as a comprehensive benchmark for evaluating OOD generalization in *changing* zero-shot learning settings which covers a much larger variety of zero-shot transfer learning scenarios potentially encountered in practice. We systematically construct training and testing data splits of increasing difficulty as measured by their Frechet-Inception Distance [23] and extensively evaluate the performance of current DML approaches.

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

Our experiments reveal that the standard evaluation splits are often close to i.i.d. evaluation settings. In contrast, our novel benchmark continually evaluates models on significantly harder learning problems, providing a more complete perspective into OOD generalization in DML. Second, we perform a large-scale study of representative DML methods on *ooDML*, and study the actual benefit of underlying regularizations such as self-supervision [36], knowledge distillation [51], adversarial regularization [56] and specialized objective functions [68, 67, 8, 25, 52]. We find that conceptual differences between DML approaches play a more significant role as the distribution shift to the test split becomes harder.

Finally, we present a study on few-shot DML as a simple extension to achieve systematic and consistent OOD generalization. As the transfer learning problem becomes harder, even very little in-domain knowledge effectively helps to adjust learned metric representation spaces to novel test distributions. We publish our code and train-test splits on three established benchmark sets, CUB200-2011 [65], CARS196 [29] and Stanford Online Products (SOP) [41]. Similarly, we provide training and evaluation episodes for further research into few-shot DML.

Overall, our contributions can be summarized as:

- Proposing the *ooDML* benchmark to create a set of more realistic train-test splits that evaluate DML generalization capabilities under increasingly more difficult zero-shot learning tasks.
- Analyzing the current DML method landscape under *ooDML* to characterize benefits and drawbacks of different conceptual approaches to DML.
- Introducing and examining few-shot DML as a potential remedy for systematically improved OOD generalization, especially when moving to larger train-test distribution shifts.

2 Related Work

DML has become essential for many applications, especially in zero-shot image and video retrieval [58, 68, 49, 24, 1]. Proposed approaches most commonly rely on a surrogate ranking task over tuples during training [59], ranging from simple pairs [17] and triplets [54] to higher-order quadruplets [5] and more generic n-tuples [58, 41, 22, 67]. These ranking tasks can also leverage additional context such as geometrical embedding structures [66, 8]. However, due to the exponentially increased complexity of tuple sampling spaces, these methods are usually also combined with tuple sampling objectives, relying on predefined or learned heuristics to avoid training over tuples that are too easy or too hard [54, 69] or reducing tuple redudancy encountered during training [68, 15, 18, 50]. More recent work has tackled sampling complexity through the usage of proxy-representations utilized as sample stand-ins during training, following a NCA [16] objective [39, 25, 61], leveraging softmax-style training through class-proxies [8, 70] or simulating intraclass structures [44].

Unfortunately, the true benefit of these proposed objectives has been put into question recently, with [52] and [40] highlighting high levels of performance saturation of these discriminative DML objectives on default benchmark splits under fair comparison. Instead, orthogonal work extending the standard DML training paradigm through multi-task approaches [53, 49, 37], boosting [42, 43], attention [26], sample generation [11, 32, 71], multi-feature learning [36] or self-distillation [51] have shown more promise with strong relative improvements under fair comparison [52, 36], however still only in single split benchmark settings. It thus remains unclear how well these methods generalize in more realistic settings [27] under potentially much more challenging, different train-to-test distribution shifts, which we investigate in this work.

3 *ooDML*: Constructing a Benchmark for OOD Generalization

A well-generalizing image representation ϕ learned on samples $x \in \mathcal{X}_{\text{train}}$ drawn from some training distribution should also learn features that can transfer to test data $\mathcal{X}_{\text{test}}$ that are not observed during training. In the particular case of OOD generalization, the learned representation ϕ is supposed to transfer to samples $\mathcal{X}_{\text{test}}$ which are not independently and identically distributed (i.i.d.) to $\mathcal{X}_{\text{train}}$. A successful approach to learning such representations is DML which is evaluated for the special case of zero-shot generalization, i.e. the transfer of ϕ to distributions of unknown classes [54, 68, 24, 8, 52, 40]. More specifically, DML models aim to learn an embedding ϕ mapping datapoints x into an embedding space Φ, which allows one to measure the similarity between x_i, x_j

Table 1: FID scores between i.i.d. subsampled training and test sets in comparison to FID scores measured on default splits used in standard DML evaluation protocols. As can be seen, the train-test distribution shift of two out of three benchmarks are actually close i.i.d. settings, in particular when compared to the train-test splits evaluated in Fig. 1 reaching scores over 200.

Dataset →	CUB	CARS	SOP
Default - different classes train/test	52.62	8.59	3.43
i.i.d. - same classes train/test	4.87 ± 0.05	2.33 ± 0.03	0.98 ± 0.01

as $g(\phi(x_i), \phi(x_j))$. Typically, g is a predefined metric, such as the Euclidean or Cosine distance and ϕ is parameterized by a deep neural network.

In realistic zero-shot learning scenarios, test distributions are not specified a priori. Thus, their respective shift to the training set indicating the difficulty of the transfer learning problem is unknown as well. To determine the generalization capabilities of ϕ, we would thus ideally measure its performance on different test distributions covering a large spectrum of difficulties. Unfortunately, standard evaluation protocols test the generalization of ϕ on a *single and fixed* train-test data split of predetermined difficulty. Hence, such protocols only allow for limited general conclusions about actual zero-shot generalization.

To thoroughly assess and compare zero-shot generalization of DML models, we must build an evaluation protocol that resembles the undetermined nature of the transfer learning problem. Hence, we need to be able to *change*, *measure* and *control* the difficulty of train-test data splits. To this end, we present an approach to construct multiple train-test splits of measurably increasing difficulty, while still resorting to the established DML benchmark sets which we subsequently use in Sec. 4 to thoroughly analyze the current state-of-the-art in DML. For future research, this approach is also easily applicable to other datasets and transfer learning problems.

3.1 Measuring the gap between train and test distributions

Measuring the distance between image datasets is a difficult task due to high dimensionality and natural noise in the images. Recently, Frechet Inception Distance (FID) [23] was proposed to measure the distance between two image distributions by using the neural embeddings of an Inception-v2 network trained for classification on the ImageNet dataset. FID assumes that the embeddings of the penultimate layer follow a Gaussian distribution, with a given mean $\mu_{\mathcal{X}}$ and covariance $\Sigma_{\mathcal{X}}$ for a distribution of images $\mathcal{X}$. The FID between two data distributions $\mathcal{X}_1$ and $\mathcal{X}_2$ is defined as:

$$d(\mathcal{X}_1, \mathcal{X}_2) \triangleq \|\mu_{\mathcal{X}_1} - \mu_{\mathcal{X}_2}\|_2^2 + \mathrm{Tr}(\Sigma_{\mathcal{X}_1} + \Sigma_{\mathcal{X}_2} - 2(\Sigma_{\mathcal{X}_1}\Sigma_{\mathcal{X}_2})^{\frac{1}{2}}) \ , \tag{1}$$

In this paper, instead of the Inception network, we use the embeddings of a ResNet-50 classifier (Frechet *ResNet* Distance) for consistency with most DML studies (see e.g. [68, 61, 25, 53, 49, 36, 52, 55]). For simplicity, in the following sections we will still use the abbreviation *FID*.

3.2 On the issue with default train-test splits in DML

To motivate the need for more comprehensive OOD evaluation protocols, we look at the split difficulty as measured by FID of typically used train-test splits and compare to i.i.d. sampling of training and test sets from the same benchmark. Empirical results in Tab. 1 show that commonly utilized DML train-test splits are very close to in-distribution learning problems when compared to more out-of-distribution splits in CARS196 and SOP (see Fig. 1). This indicates that semantic differences due to disjoint train and test classes, do not necessarily relate to actual significant distribution shifts between the train and test set. This also explains the consistently lower zero-shot retrieval performance on CUB200-2011 as compared to both CARS196 and SOP in literature [68, 67, 24, 52, 40, 36], despite SOP containing significantly more classes with fewer examples per class. In addition to the previously discussed issues of DML evaluation protocols, this further questions conclusions drawn from these protocols about the OOD generalization of representations ϕ.

3.3 Creating a train-test splits of increasing difficulty

Let $\mathcal{X}_{\text{train}}$ and $\mathcal{X}_{\text{test}}$ denote the original train and test set of a given benchmark dataset $\mathcal{D} = \mathcal{X}_{\text{train}} \cup \mathcal{X}_{\text{test}}$. To generate train-test splits of increasing difficulty while retaining the available data $\mathcal{D}$

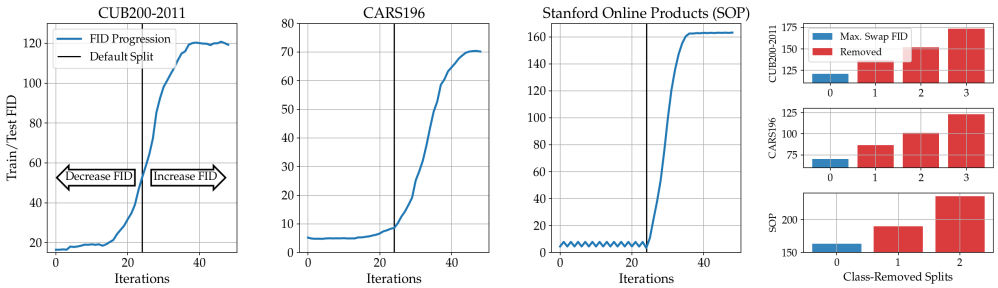

Figure 1: *FID progression with iterative class swapping and removal for train-test split generation.* (Col. 1-3) FID per swapping iteration t on all benchmarks. (Rightmost) FID of data splits obtained by additional k iterations of removing classes. The blue bar denotes the maximal FID after swapping.

and maintaining balance of their sizes, we exchange samples between them. To ensure semantic consistency and unbiased data distributions based on image context unrelated to the target object categories, we swap entire classes instead of individual samples. Using FID to measure distribution similarity, the goal is to first identify classes $C_{\text{train}} \subset \mathcal{X}_{\text{train}}$ and $C_{\text{test}} \subset \mathcal{X}_{\text{test}}$ whose exchange yields higher FID $d(\mathcal{X}_{\text{train}}, \mathcal{X}_{\text{test}})$. To this end, we select C_{train} and C_{test} as

$$C^*_{\text{train}} = \underset{C_{\text{train}} \in \mathcal{X}_{\text{train}}}{\arg\max} \, \|\mu_{C_{\text{train}}} - \mu_{\mathcal{X}_{\text{train}}}\|_2 - \|\mu_{C_{\text{train}}} - \mu_{\mathcal{X}_{\text{test}}}\|_2 \tag{2}$$

$$C^*_{\text{test}} = \underset{C_{\text{test}} \in \mathcal{X}_{\text{test}}}{\arg\max} \, \|\mu_{C_{\text{test}}} - \mu_{\mathcal{X}_{\text{test}}}\|_2 - \|\mu_{C_{\text{test}}} - \mu_{\mathcal{X}_{\text{train}}}\|_2 \tag{3}$$

where we measure distance to mean class-representations $\mu_{\mathcal{X}_C}$. Similar unimodal approximation to intraclass distributions have also been used in recent literature [32, 49, 36]. By iteratively exchanging classes between data splits, i.e. $\mathcal{X}^{t+1}_{\text{train}} = (\mathcal{X}^t_{\text{train}} \setminus C^*_{\text{train}}) \cup C^*_{\text{test}}$ and vice versa, we obtain a more difficult train-test split $(\mathcal{X}^{t+1}_{\text{train}}, \mathcal{X}^{t+1}_{\text{test}})$ at iteration step t. Hence, we obtain a sequence of train-test splits $\boldsymbol{\mathcal{X}}_{\mathcal{D}} = ((\mathcal{X}^0_{\text{train}}, \mathcal{X}^0_{\text{test}}), \ldots, (\mathcal{X}^t_{\text{train}}, \mathcal{X}^t_{\text{test}}), \ldots, (\mathcal{X}^T_{\text{train}}, \mathcal{X}^T_{\text{test}}))$, with $\mathcal{X}^0_{\text{train}} \triangleq \mathcal{X}_{\text{train}}$ and $\mathcal{X}^0_{\text{test}} \triangleq \mathcal{X}_{\text{test}}$. Note that in Eq. 2, we only consider the class means and neglect the covariance term from Eq. 1; we observed this approximation on the Frechet distance to be sufficient to generate data splits of increasing FID. Fig. 1 (columns 1-3) indeed shows that FID gradually increase with each swap until $d(\mathcal{X}^{t+1}_{\text{train}}, \mathcal{X}^{t+1}_{\text{test}})$ cannot be further increased by swapping classes. For CUB200-2011 and CARS196, we swap two classes per iteration, while for Stanford Online Products we swap 1000 classes due to a significantly higher class count. Moreover, to cover the overall spectrum of distribution shifts and ensure comparability between benchmarks we also reverse the iteration procedure on CUB200-2011 to generate splits minimizing FID while still maintaining disjunct train and test classes.

To further increase $d(\mathcal{X}^T_{\text{train}}, \mathcal{X}^T_{\text{test}})$ beyond convergence (see Fig. 1) of the swapping procedure, we subsequently also identify and remove classes from both $\mathcal{X}^T_{\text{train}}$ and $\mathcal{X}^T_{\text{test}}$. More specifically, we remove classes C_{train} from $\mathcal{X}^T_{\text{train}}$ that are closest to the mean of $\mathcal{X}^T_{\text{test}}$ and vice versa. For k steps, we successively repeat class removal as long as 50% of the original data is still maintained in these additional train-test splits. Fig. 1 (rightmost) shows how splits generated through class removal progressively increase the FID beyond what was achieved only by swapping. To analyze if the generated data splits are not inherently biased to the used backbone network for FID computation, we also repeat the train-test generation based on feature representations obtained from different architectures, pretraining methods and datasets. The corresponding results in the supplementary show a consistent, similar increase in (normalized) FID. Overall, using class swapping and removal we select splits that cover the broadest FID range possible, while still maintaining sufficient data. Our splits are significantly more difficult than the default splits, thereby much closer resembling a potential distribution shifts faced in practice.

4 Assessing the State of Generalization in Deep Metric Learning

This section assesses the state of zero-shot generalization in DML via a large experimental study of representative DML methods on our *ooDML* benchmark, offering a much more complete and thorough perspective on zero-shot generalization in DML as compared to previous DML studies [13, 52, 40, 37].

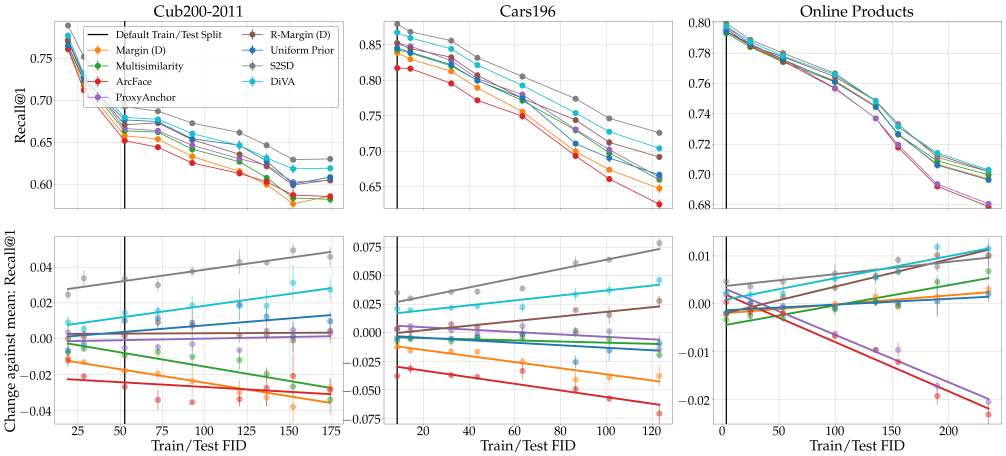

Figure 2: *Zero-Shot Generalization performance under varying distribution shifts.* (top row) Absolute Recall@1 performance for each increasingly more difficult train-test split in the *goodDML* benchmark (cf. Sec. 3) on CUB200-2011, CARS196 and SOP. We report mean Recall@1 performances and standard deviations over 5 runs. For results based on mAP@1000 see supplementary. (bottom) Differences of performances against the mean over all methods for each train-test split.

4.1 Experimental Details

For our experiments we use the three most widely used benchmarks in DML, CUB200-2011[65], CARS196[29] and Stanford Online Products[41]. For a complete list of implementation and training details see the supplementary if not explicitly stated in the respective sections. Moreover, to measure generalization performance, we resort to the most widely used metric for image retrieval in DML, Recall@k. Additionally, we also evaluate results over mean average precision (mAP@1000) [52, 40], but provide respective tables and visualizations in the supplementary when the interpretation of results is not impacted.

The exact training and test splits ultimately utilized throughout this work are selected based on Fig. 1 to ensure approximately uniform coverage of the spectrum of distribution shifts within intervals ranging from the lowest (near i.i.d. splits) to the highest generated shift achieved with class removal. For experiments on CARS196 and Stanford Online Products, eight total splits were investigated, included the original default benchmark split. For CUB200-2011, we select nine splits to also account for benchmark additions with reduced distributional shifts. The exact FID ranges are provided in the supplementary. Training on CARS196 and CUB200-2011 was done for a maximum of 200 epochs following standard training protocols utilized in [52], while 150 epochs were used for the much larger SOP dataset. Additional training details if not directly stated in the respective sections can be found in the supplementary. For further usage, we will release our source code and train-test splits.

4.2 Zero-shot generalization under varying distribution shifts

Many different concepts have been proposed in DML to learn embedding functions ϕ that generalize from the training distribution to differently distributed test data. To analyze the zero-shot transfer capabilities of DML models, we consider representative approaches making use of the following concepts: *(i)* surrogate ranking tasks and tuple mining heuristics (Margin loss with distance-based sampling [68] and Multisimilarity loss [67]), *(ii)* geometric constraints or class proxies (ArcFace [8], ProxyAnchor [25]), *(iii)* learning of semantically diverse features (R-Margin [52]) and self-supervised training (DiVA [36]), adversarial regularization (Uniform Prior [56]) and *(iv)* knowledge self-distillation (S2SD [51]). Fig. 2 (top) analyzes these methods for their generalization to distribution shifts the varying degrees represented in *ooDML*. The top row shows absolute zero-shot retrieval performance measured on Recall@1 (results for mAP@1000 can be found in the supplementary) with respect to the FID between train and test sets. Additionally, Fig. 2 (bottom) examines the relative differences of performance to the performance mean over all methods for each train-test split. Based on these experiments, we make the following main observations:

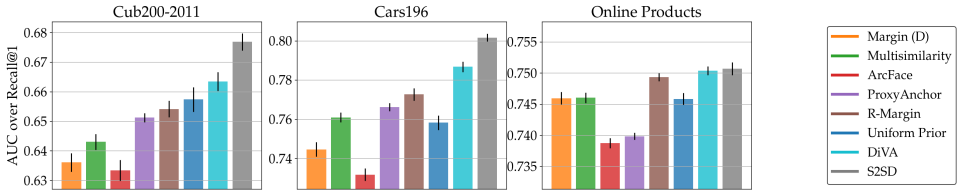

Figure 3: *Comparison of DML methods via AGS based on Recall@1 across benchmarks.* To compute AGS (cf. Sec. 4.2), we aggregate the performances from Fig. 2 across all train-test distribution shifts of our proposed *ooDML* benchmark using the Area-Under-the-Curve metric.

***(i)* Performance deteriorates monotonically with the distribution shifts.** Independent of dataset, approach or evaluation metric, performance drops steadily as the distribution shift increases.
***(ii)* Relative performance differences are affected by train-test split difficulty.** We see that the overall ranking between approaches oftentimes remains stable on the CARS196 and CUB200-2011 datasets. However, we also see that particularly on a large-scale dataset (SOP), established proxy-based approaches ArcFace [8] (which incorporates additional geometric constraints) and ProxyAnchor [25] are surprisingly susceptible to more difficult distribution shifts. Both methods perform poorly compared to the more consistent general trend of the other approaches. Hence, conclusions on the generality of methods solely based on the default benchmarks need to be handled with care, as at least for SOP, performance comparisons reported on single (e.g. the standard) data splits *do not* translate to more general train-test scenarios.
***(iii)* Conceptual differences matter at larger distribution shifts** While the ranking between most methods is largely consistent on CUB200-2011 and CARS196, their differences in performance becomes more prominent with increasing distribution shifts. The relative changes (deviation from the mean of all methods at the stage) depicted in Fig. 2 (bottom) clearly indicates that particular methods based on machine learning techniques such as self-supervision, feature diversity (DiVA, R-Margin) and self-distillation (S2SD) are among the best at generalizing in DML on more challenging splits while retaining strong performance on more i.i.d. splits as well.

While directly stating performance in dependence to the individual distribution shifts offers a detailed overview, the overall comparison of approaches is typically based on single benchmark scores. To further provide a single metric of comparison, we utilize the well-known Area-under-Curve (AUC) score to condense performance (either based on Recall@1 or mAP@1000) over all available distribution shifts into a single aggregated score indicating general zero-shot capabilities. This **Aggregated Generalization Score (AGS)** is computed based on the normalized FID scores of our splits to the interval $[0, 1]$. As Recall@k or mAP@k scores are naturally bounded to $[0, 1]$, AGS is similarly bound to $[0, 1]$ with higher being the better model. Our corresponding results are visualized in Fig. 3. Indeed, we see that AGS reflects our observations from Fig. 2, with self-supervision (DiVA) and self-distillation (S2SD) generally performing best when facing unknown train-test shifts. Exact scores are provided in the supplementary.

4.3 Consistency of structural representation properties on *ooDML*

The work of Roth et al. [52] attempts to identify potential drivers of generalization in DML by measuring the following structural properties of a representation ϕ: *(i)* the mean distance π_{inter} between the centers of the embedded samples of each class, *(ii)* the mean distance π_{intra} between the embedded samples within a class π_{intra}, *(iii)* the 'embedding space density' measured as the ratio $\pi_{\text{ratio}} = \frac{\pi_{\text{intra}}}{\pi_{\text{inter}}}$ and *(iv)* 'spectral decay' $\rho(\Phi)$ measuring the degree of uniformity of singular values obtained by singular value decomposition on the training sample representations, which indicates the number of significant directions of variance. For a more detailed description, we refer to [52]. These metrics indeed are empirically shown to exhibit a certain correlation to generalization performance on the default benchmark splits. In contrast, we are now interested if these observations hold when measuring generalization performance on the *ooDML* train-test splits of varying difficulty.
We visualize our results in Fig. 4 for CUB200-2011 and SOP, with CARS196 provided in the supplementary. For better visualization we normalize all measured values obtained for both metrics *(i)*-*(iv)* and the recall performances (Recall@1) to the interval $[0, 1]$ for each train-test split. Thus, the relation between structural properties and generalization performance becomes comparable

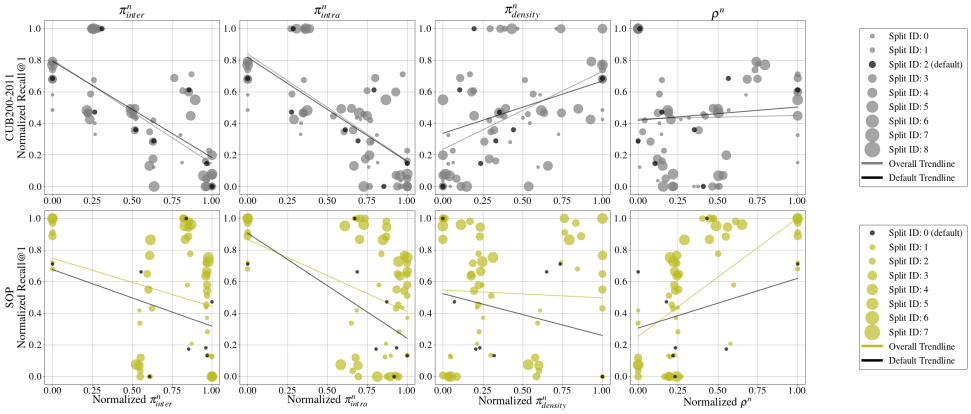

Figure 4: *Generalization metrics computed on ooDML benchmark for CUB200-2011 and SOP.* Each column plots one of the (normalized) measured structural representation property (cf. 4.3) over the corresponding Recall@1 performance for all examined DML methods and distribution shifts. Trendlines are computed as least squares fit over all datapoints (overall), respectively only those corresponding to default splits (default).

across all train-test splits, allowing us to examine if superior generalization is still correlated to the structural properties of the learned representation ϕ, i.e. if the correlation is independent of the underlying distribution shifts. For a perfectly descriptive metric, one should expect to see heavy correlation between normalized metric and normalized generalization performance jointly across shifts. Unfortunately, our results show only a small indication of any structural metric being consistently correlated with generalization performance over varying distribution shifts. This is also observed when evaluating only against basic, purely discriminative DML objectives as was done in [52] for the default split, as well as when incorporating methods that extend and change the base DML training setup (such as DiVA [36] or adversarial regularization [56]).

This not only demonstrates that experimental conclusions derived from the analysis of only single benchmark split may not hold for overall zero-shot generalization, but also that future research should consider more general learning problems and difficulty to better understand the conceptual impact various regulatory approaches. To this end, our benchmark protocol offers more comprehensive experimental ground for future studies to find potential drivers of zero-shot generalization in DML.

4.4 Network capacity and pretrained representations

A common way to improve generalization, as also highlighted in [52] and [40], is to select a stronger backbone architecture for feature extraction. In this section, we look at how changes in network capacity can influence OOD generalization across distribution shifts. Moreover, we also analyze the zero-shot performance of a diverse set of state-of-the-art pretraining approaches.

Influence of network capacity. In Fig. 5, we compare different members of the ResNet architecture family [20] with increasing capacity, each of which achieve increasingly higher performance on i.i.d. test benchmarks such as ImageNet [7], going from a small ResNet18 (R18) over ResNet50 (R50) to ResNet101 (R101) variants. As can be seen, while larger network capacity helps to some extent, we observe that performance actually saturates in zero-shot transfer settings, regardless of the DML approach and dataset (in particular also the large scale SOP dataset). Interestingly, we also observe that the performance drops with increasing distribution shifts are consistent across network capacity, suggesting that zero-shot generalization is less driven by network capacity but rather conceptual choices of the learning formulation (compare Fig. 2).

Generic representations versus Deep Metric Learning. Recently, self-supervised representation learning has taken great leaps with ever stronger models trained on huge amounts of image [28, 45] and language data [9, 34, 2]. These approaches are designed to learn expressive, well-transferring features and methods like CLIP [45] even prove surprisingly useful for zero-shot classification. We now evaluate and compare such representations against state-of-the-art DML models to understand if generic representations that are readily available nowadays actually pose an alternative to explicit application of DML. We select state-of-the-art self-supervision models *SwAV* [3] (ResNet50 back-

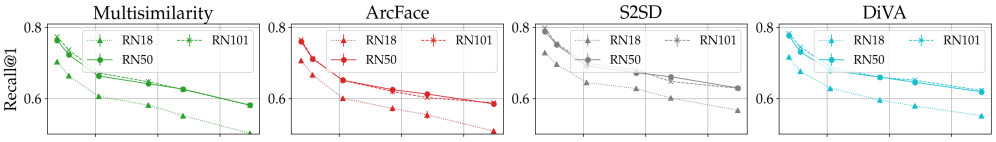

Figure 5: *Generalization performance for different backbone architectures for varying distribution shifts on CUB200-2011.* We show absolute Recall@1 performances averaged over 5 runs for each train-test split. Other datasets show similar results and are provided in the supplementary.

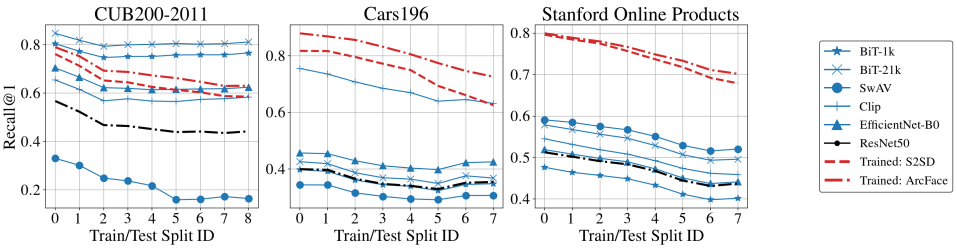

Figure 6: *Comparison of DML to various non-adapted generic representations pretrained on large amounts unlabelled data and state-of-the-art architectures.* For DML, we show best and worst DML objectives based on results in Fig. 2. Performance of generic representations are heavily dependent on target dataset, architecture, amount of training data and learning objective.

bone), CLIP [45] trained via natural language supervision on a large dataset of 400 million image and sentence pairs (VisionTransformer [10] backbone), BiT(-M) [28], which trains a ResNet50-V2 [28] on both the standard ImageNet [7] (1 million training samples) and the ImageNet-21k dataset [7, 48] with 14 million training samples and over 21 thousand classes, an EfficientNet-B0 [60] trained on ImageNet, and a standard baseline ResNet50 network trained on ImageNet. We note that none of these representations have been additionally adapted to the benchmark sets and only the pretrained representations are being evaluated, in contrast to the DML approaches which have been trained on the respective train splits.

The results presented in Fig. 6 show large performance differences of the pretrained representations, which are largely dependent on the test dataset. While BiT outperforms the DML state-of-the-art on CUB200-2011 without any finetuning, it significantly trails behind the DML models on the other two datasets. On CARS196, only CLIP comes close to the DML approaches when the distribution shift is sufficiently large. Finally, on SOP, none of these models comes even close to the adapted DML methods. This shows how although representations learned by extensive pretraining can offer strong zero-shot generalization, their performance heavily depends on the target dataset and specific model. Furthermore, the generalization abilities notably depend on the size of the pretraining dataset (compare e.g. BiT-1k vs BiT-21k or CLIP), which is significantly larger than the number of training images seen by the DML methods. We see that only actual training on these datasets provides sufficiently reliable performance.

4.5 Few-shot adaption boosts generalization performance in DML

Since distribution shifts can be arbitrarily large, the zero-shot transfer of ϕ can be ill-posed. Features learned on a training set $\mathcal{X}_{\text{train}}$ will not meaningfully transfer to test samples $\mathcal{X}_{\text{test}}$ once they are sufficiently far from $\mathcal{X}_{\text{train}}$, as also already indicated by Fig. 2. As a remedy *few-shot learning* [57, 64, 14, 46, 31, 6, 63] assumes few samples of the test distribution to to be available during training, i.e. adjusting a previously learned representation. While these approaches are typically explicitly trained for fast adaption to novel classes, we are now interested if similar adaptation of DML representations ϕ helps to bridge increasingly large distribution shifts.

To investigate this hypothesis, we follow the evaluation protocol of few-shot learning and use k representatives (also referred to as *shots*) of each class from a test set $\mathcal{X}_{\text{test}}$ as a support set for finetuning the penultimate embedding network layer. The remaining test samples then represent the new test set to evaluate retrieval performance, also referred to as query set. For evaluation we perform

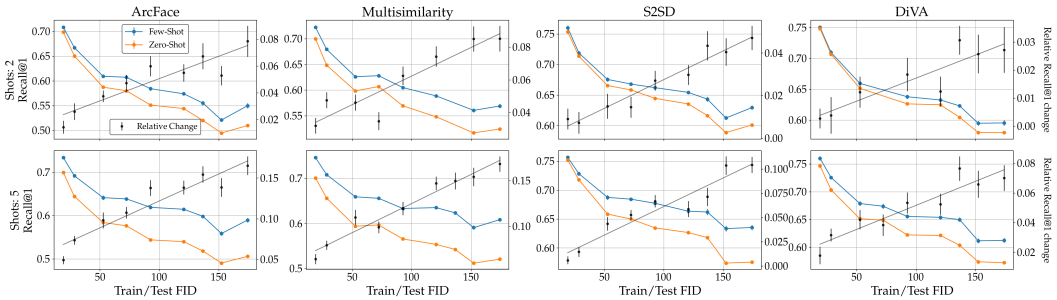

Figure 7: *Few-Shot adaptation of DML representations on CUB200-2011.* Columns show average Recall@1 performance over 10 episodes of 2- and 5-shot adaption for various DML approaches (fewshot and zeroshot), highlighting a substantial benefit of few-shot adaptation for *a priori* unknown distribution shifts (see black line highlighting relative improvements).

10 episodes, i.e. we repeat and average the adaptation of ϕ over 10 different, randomly sampled support and corresponding query sets. Independent of the DML model used for learning the original representation ϕ on $\mathcal{X}_{\text{train}}$, adaptation to the support data is conducted using the Marginloss [68] objective with distance-based sampling [68] due to its faster convergence. This also ensures fair comparisons to the adaptation benefit to ϕ and also allows to adapt complex approaches like self-supervision (DiVA [36]) to the small number of samples in the support sets.

Fig. 7 shows 2 and 5 shot results on CUB200-2011, with CARS196 available in the supplementary. SOP is not considered since each class is already composed of small number of samples. As we see, even very limited in-domain data can significantly improve generalization performance, with the benefit becoming stronger for larger distribution shifts. Moreover, we observe that weaker approaches like ArcFace [8] seem to benefit more than state-of-the-art methods like S2SD [51] or DiVA [36]. We presume this to be caused by their underlying concepts already encouraging learning of more robust and general features. To conclude, few-shot learning appears to provides a substantial benefit when facing difficult OOD learning settings, providing an especially strong starting point for reliable improvement in settings where the shift is not known *a priori*.

5 Conclusion

In this work we analyzed zero-shot transfer of image representations learned by Deep Metric Learning (DML) models. We proposed a systematic construction of train-test data splits of increasing difficulty, as opposed to standard evaluation protocols that test out-of-distribution generalization only on single data splits of fixed difficulty. Based on this, we presented the novel benchmark *ooDML* and thoroughly assessed current DML methods. Our study reveals the following main findings:

Standard evaluation protocols are insufficient to probe general out-of-distribution transfer: Prevailing train-test splits in DML are often close to i.i.d. evaluation settings. Hence, they only provide limited insights into the impact of train-test distribution shift on generalization performance. Our benchmark *ooDML* alleviates this issue by evaluating a large, controllable and measurable spectrum of problem difficulty to facilitate future research.

Larger distribution shifts show the impact of conceptual differences in DML approaches: Our study reveals that generalization performance degrades consistently with increasing problem difficulty for all DML methods. However, certain concepts underlying the approaches are shown to be more robust to shifts than others, such as semantic feature diversity and knowledge-distillation.

Generic, self-supervised representations without finetuning can surpass dedicated data adaptation: When facing large distribution shifts, representations learned only by self-supervision on large amounts of of unlabelled data are competitive to explicit DML training without any finetuning. However, their performance is heavily dependent on the data distribution and the models themselves.

Few-shot adaptation consistently improves out-of-distribution generalization in DML: Even very few examples from a target data distribution effectively help to adapt DML representations. The benefit becomes even more prominent with increasing train-test distribution shifts, and encourages further research into few-shot adaptation in DML.

References

[1] 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.

[2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020.

[3] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 9912–9924. Curran Associates, Inc., 2020.

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 13–18 Jul 2020.

[5] 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.

[6] Yinbo Chen, Xiaolong Wang, Zhuang Liu, Huijuan Xu, and Trevor Darrell. A new meta-baseline for few-shot learning. *CoRR*, abs/2003.04390, 2020.

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[8] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[11] 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.

[12] Logan Engstrom, Brandon Tran, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. Exploring the landscape of spatial robustness. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1802–1811. PMLR, 09–15 Jun 2019.

[13] Istvan Fehervari, Avinash Ravichandran, and Srikar Appalaraju. Unbiased evaluation of deep metric learning algorithms, 2019.

[14] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135. PMLR, 06–11 Aug 2017.

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

[16] Jacob Goldberger, Geoffrey E Hinton, Sam Roweis, and Russ R Salakhutdinov. Neighbourhood components analysis. In L. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems*, volume 17. MIT Press, 2005.

[17] 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.

[18] Ben Harwood, BG Kumar, Gustavo Carneiro, Ian Reid, Tom Drummond, et al. Smart mining for deep metric learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2821–2829, 2017.

[19] 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.

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

[21] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019.

[22] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In Defense of the Triplet Loss for Person Re-Identification. *arXiv e-prints*, page arXiv:1703.07737, Mar. 2017.

[23] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium, 2017.

[24] 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.

[25] 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.

[26] 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.

[27] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton A. Earnshaw, Imran S. Haque, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. Wilds: A benchmark of in-the-wild distribution shifts, 2021.

[28] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (bit): General visual representation learning. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 491–507, Cham, 2020. Springer International Publishing.

[29] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 554–561, 2013.

[30] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex), 2021.

[31] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[32] 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.

[33] 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.

[34] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019. cite arxiv:1907.11692.

[35] Timo Milbich, Miguel Bautista, Ekaterina Sutter, and Björn Ommer. Unsupervised video understanding by reconciliation of posture similarities. In *Proc. ICCV*, 2017.

[36] 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. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 590–607, Cham, 2020. Springer International Publishing.

[37] Timo Milbich, Karsten Roth, Biagio Brattoli, and Björn Ommer. Sharing matters for generalization in deep metric learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[38] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[39] 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*, pages 360–368, 2017.

[40] Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. A metric learning reality check, 2020.

[41] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4004–4012, 2016.

[42] Michael Opitz, Georg Waltner, Horst Possegger, and Horst Bischof. Bier-boosting independent embeddings robustly. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5189–5198, 2017.

[43] 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.

[44] 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.

[45] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. *CoRR*, abs/2103.00020, 2021.

[46] Aniruddh Raghu, Maithra Raghu, Samy Bengio, and Oriol Vinyals. Rapid learning or feature reuse? towards understanding the effectiveness of maml. In *International Conference on Learning Representations*, 2020.

[47] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do ImageNet classifiers generalize to ImageNet? In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5389–5400. PMLR, 09–15 Jun 2019.

[48] Tal Ridnik, Emanuel Ben Baruch, Asaf Noy, and Lihi Zelnik-Manor. Imagenet-21k pretraining for the masses. *CoRR*, abs/2104.10972, 2021.

[49] 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*, pages 8000–8009, 2019.

[50] 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 2020.

[51] Karsten Roth, Timo Milbich, Björn Ommer, Joseph Paul Cohen, and Marzyeh Ghassemi. S2SD: simultaneous similarity-based self-distillation for deep metric learning. *CoRR*, abs/2009.08348, 2020.

[52] Karsten Roth, Timo Milbich, Samarth Sinha, Prateek Gupta, Bjorn Ommer, and Joseph Paul Cohen. Revisiting training strategies and generalization performance in deep metric learning. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 8242–8252. PMLR, 13–18 Jul 2020.

[53] 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.

[54] 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*, pages 815–823, 2015.

[55] Jenny Seidenschwarz, Ismail Elezi, and Laura Leal-Taixé. Learning intra-batch connections for deep metric learning, 2021.

[56] Samarth Sinha, Karsten Roth, Anirudh Goyal, Marzyeh Ghassemi, Hugo Larochelle, and Animesh Garg. Uniform priors for data-efficient transfer, 2020.

[57] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

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

[59] Juan Luis Suárez, Salvador García, and Francisco Herrera. A tutorial on distance metric learning: Mathematical foundations, algorithms and experiments, 2018.

[60] Mingxing Tan and Quoc Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6105–6114. PMLR, 09–15 Jun 2019.

[61] Eu Wern Teh, Terrance DeVries, and Graham W. Taylor. Proxynca++: Revisiting and revitalizing proxy neighborhood component analysis. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 448–464, Cham, 2020. Springer International Publishing.

[62] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. In *International Conference on Learning Representations*, 2020.

[63] Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B. Tenenbaum, and Phillip Isola. Rethinking few-shot image classification: A good embedding is all you need? In Andrea Vedaldi, Horst Bischof, Thomas Brox,

and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 266–282, Cham, 2020. Springer International Publishing.

[64] Eleni Triantafillou, Richard Zemel, and Raquel Urtasun. Few-shot learning through an information retrieval lens. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[65] 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.

[66] Jian Wang, Feng Zhou, Shilei Wen, Xiao Liu, and Yuanqing Lin. Deep metric learning with angular loss. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2593–2601, 2017.

[67] 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.

[68] Chao-Yuan Wu, R Manmatha, Alexander J Smola, and Philipp Krahenbuhl. Sampling matters in deep embedding learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2840–2848, 2017.

[69] 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.

[70] Andrew Zhai and Hao-Yu Wu. Classification is a strong baseline for deep metric learning, 2018.

[71] 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.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [No]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No] We will make code, data and instructions available after review.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See also the supplementary.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] We included the type of GPUs in Sec. 4.1.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No] All relevant assets, namely both code to run everything as well as to generate the respective benchmark splits will be made available after the review process.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]