Non-isotropy Regularization for Proxy-based Deep Metric Learning

Karsten Roth[1], Oriol Vinyals[2], Zeynep Akata[1,3]
[1]University of Tübingen, [2]DeepMind, [3]MPI for Intelligent Systems

Abstract

Deep Metric Learning (DML) aims to learn representation spaces on which semantic relations can simply be expressed through predefined distance metrics. Best performing approaches commonly leverage class proxies as sample stand-ins for better convergence and generalization. However, these proxy-methods solely optimize for sample-proxy distances. Given the inherent non-bijectiveness of used distance functions, this can induce locally isotropic sample distributions, leading to crucial semantic context being missed due to difficulties resolving local structures and intraclass relations between samples. To alleviate this problem, we propose non-isotropy regularization ($\mathbb{NIR}$) for proxy-based Deep Metric Learning. By leveraging Normalizing Flows, we enforce unique translatability of samples from their respective class proxies. This allows us to explicitly induce a non-isotropic distribution of samples around a proxy to optimize for. In doing so, we equip proxy-based objectives to better learn local structures. Extensive experiments highlight consistent generalization benefits of $\mathbb{NIR}$ while achieving competitive and state-of-the-art performance on the standard benchmarks CUB200-2011, Cars196 and Stanford Online Products. In addition, we find the superior convergence properties of proxy-based methods to still be retained or even improved, making $\mathbb{NIR}$ very attractive for practical usage. Code available at github.com/ExplainableML/NonIsotropicProxyDML.

1. Introduction

Visual similarity plays a crucial role for applications in image & video retrieval and clustering [4, 52, 63], face re-identification [7, 19, 32] or general supervised [22] and unsupervised [5, 18] contrastive representation learning. A majority of approaches used in these fields employ or can be derived from Deep Metric Learning (DML). DML aims to learn highly nonlinear distance metrics parametrized by deep networks. These networks span a representation space in which semantic relations between images are expressed as distances between respective representations. In the field of DML, methods utilizing proxies have shown to provide among the most consistent and highest performances in addition to fast convergence [23, 38, 56]. While other methods introduce ranking tasks over samples for the network to solve, proxy-based methods require the network to contrast samples against a proxy representation, commonly approximating generic class prototypes. Their utilization addresses sampling complexity issues [16, 47, 52, 63] inherent to purely sample-based approaches, resulting in improved convergence and benchmark performance.

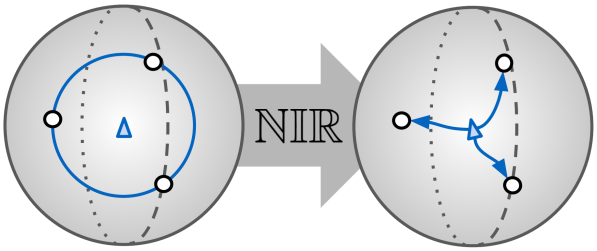

Figure 1. Proxy-based Deep Metric Learning methods optimize for non-bijective similarity measures between proxies (▲) and sample representation (○), which can introduce local isotropy around proxies, impeding local structures and non-discriminative features to be learned. We propose $\mathbb{NIR}$ to explicitly resolve this.

However, there is no free lunch. Relying on sample-proxy relations, relations between samples within a class can not be explicitly captured. This is exacerbated by proxy-based objectives optimizing for distances between samples and proxies using non-bijective distance functions. This means, for a particular proxy, that alignment to a sample is non-unique - as long as the angle between sample and proxy is retained, i.e. samples being aligned isotropically around a proxy (see Fig. 1), their distances and respective loss remain the same. This means that samples lie on a hypersphere centered around a proxy with same distance and thus incurring the same training loss. This incorporates an undesired prior over sample-proxy distributions which doesn't allow local structures to be resolved well. By incorporating multiple classes and proxies (which is automatically done when applying proxy-based losses such as [23, 38, 43, 56] to training data with multiple classes), this is extended to a mixture of sample distributions around proxies. While this offers an implicit workaround to ad-

dress isotropy around modes by incorporating relations of samples to proxies from different classes, relying only on other unrelated proxies potentially far away makes fine-grained resolution of local structures difficult. Furthermore, as training progresses and proxies move further apart. As a consequence, the distribution of samples around proxies, which proxy-based objectives optimize for, comprises modes with high affinity towards local isotropy. This introduces semantic ambiguity, as semantic relations between samples within a class are not resolved well. However, a lot of recent work has shown that understanding and incorporating these non-discriminative relations drives generalization performance [31, 34, 46, 49, 66].

To tackle this issue without resorting to sample-based objectives that impede the superior convergence and generalization of proxy-based approaches, this work proposes non-isotropy regularisation ($\mathbb{NIR}$) for proxy-based DML. $\mathbb{NIR}$ extends proxy-based objectives to encourage explicitly learning unique sample-proxy relations and eliminating semantic ambiguity. In detail, we introduce a novel uniqueness constraint, in which samples within a class must be uniquely and sufficiently described by a (non-linear) translation from the respective class proxy. This explicitly induces a distribution for proxy-based objectives to match in which isotropy and ambiguity is heavily penalized. We achieve uniqueness by leveraging a bijective and thus invertible family of translations. As the proxy-sample translations need to adapt to the specific domain at hand, we require both trainability and non-linearity of our translation models. These functional constraints are naturally expressed through Normalizing Flows and Invertible Networks [2, 8, 41]. Using conditional variants, we then formalize $\mathbb{NIR}$ where sample relations are (uniquely) mapped by a Normalizing Flow given some residual conditioned on the respective class proxy.

Extensive experiments show that $\mathbb{NIR}$ indeed introduces higher feature diversity, reduces overclustering, increases uniformity in learned representation spaces and learns more diverse class-distributions than non-regularized counterparts. Evaluating our approach on the standard DML benchmarks CUB200-2011 [57], CARS196 [30] and Stanford Online Products [40] showcases improved generalization capabilities of $\mathbb{NIR}$-equipped proxy DML, achieving competitive or state-of-the-art performance while retaining or even improving convergence speeds.

2. Related Works

Deep Metric Learning (DML) has driven research in image and video retrieval & zero-shot clustering applications [4, 52, 54, 63], with particular applications for example in person re-identification [7, 19, 32, 59] and as an auxiliary tool for improved supervised [22] and unsupervised representation learning [5, 18, 37]. Commonly proposed DML methods introduce ranking tasks for networks to solve as training surrogates. These can involve ranking constituents in tuples (such as pairs [15, 39], triplets [19, 52, 63] or higher-order tuples [40, 54, 61]) to contrasting between sample and prototypical representations [23, 38, 43, 56, 69]. These prototypical- or proxy-based approaches are commonly introduced in order to address sampling complexity issues when sampling tuples for a network to solve, which are otherwise addressed through various sampling heuristics [13, 16, 47, 52, 63]. We propose a natural extension to proxy-based approaches by addressing a major shortcoming introduced when only contrasting between samples and proxies while retaining beneficial properties of these methods. Finally, recent work has focused on generic extensions to DML to improve the quality of learned representation spaces through divide-and-conquer [51], synthetic data [31, 66], adversarial and graph-based training [11, 53, 69], bypassing representation bottlenecks [20, 48], attention [25] and auxiliary or few-shot feature learning [12, 34–36, 46, 49]. These works offer distinct, orthogonal benefits.

3. Non-isotropic Deep Metric Learning

A DML model defines a distance metric $d_\psi(x_i, x_j)$ over images $x_i \in \mathcal{X}$ parametrized by a feature extraction backbone ϕ and a projection f onto the final metric space $\Psi \subset \mathbb{R}^d$, such that $\Psi := f \circ \phi(\mathcal{X})$. Ψ is commonly normalized to the unit hypersphere [60, 63] such that $\Psi = S_\Psi^{d-1}$. This metric space is commonly equipped with a predefined distance metric such as the euclidean distance $d(\bullet, \bullet)$ or cosine similarity $s(\bullet, \bullet)$, which are equivalent on the hypersphere [58, 65]. During training, DML learns $\psi = f \circ \phi$ to connect $d_\psi(x_i, x_j) := d(\psi_i, \psi_j)$ to the semantic similarity of images x_i and x_j. Training methods commonly involve the definition of ranking tasks for the network to solve - given e.g. a triplet of anchor x_a, positive x_p and negative x_n with $y_a = y_p \neq y_n$ where $y \in \mathcal{Y}$ denotes the respective class and triplets $(x_a, x_p, x_n) \in \mathcal{T}_\mathcal{B}$ sampled from a minibatch $\mathcal{B}$. However, tuple sampling is difficult, as the tuple space complexity increases with tuple dimensionality; incurring a lot of redundancy [16, 52, 63]. While sampling heuristics have been introduced to address this [16, 47, 52, 63], recent work [23, 38, 56] has heuristically supported the promise of proxy-based ranking objectives.

3.1. On the shortcoming of proxy-based DML

Proxy-based objectives use contrastive operations not between samples (e.g. via the cosine similarity $s_\psi(x_i, x_j)$ [7, 23, 61]), but between class-prototypical (class proxies) representations $\rho_j \in \mathcal{P}$ $s(\psi_i, \rho_j)$ for classes y_i and y_j. This removes the need for complex sampling operations in methods reliant on sample-based tuples, which allows proxy-based objectives to benefit from fast convergence and good

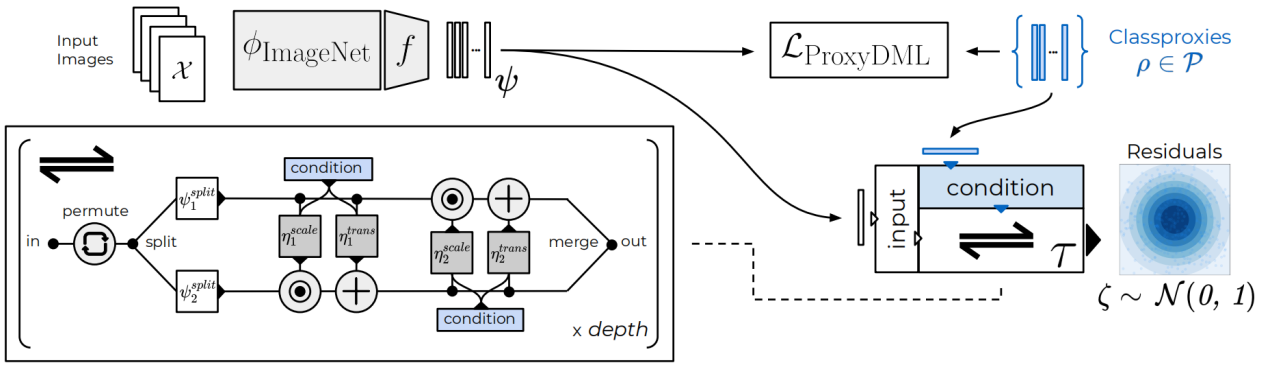

Figure 2. $\mathbb{NIR}$ - *Non-isotropy Regularisation.* We refine the distribution of samples ψ around class proxies ρ, $p(\psi|\rho)$, learned via proxy-based DML by leveraging Normalizing Flows (τ, $\leftrightharpoons$). These allow us to define a bijective translation τ which uses a simple density $q = \mathcal{N}(0, 1)$ of residuals ζ to induce a distribution over unique sample-proxy relations, $p(\tau(\zeta|\rho)|\rho)$. This allows for better resolution of local structures and non-discriminative features to be learned, improving generalisation while retaining fast convergence.

generalization performance.

However, this property also incurs the strongest shortcoming, because relying on sample-proxy pairs and the non-bijective similarity measure $s(\psi, \rho) := s(\psi_i, \rho_{y_{\psi_i}})$ can induce features to locally follow an isotropic distribution around the proxy. This can be seen more explicitly when looking at the sample-proxy distributions various proxy-objectives optimize for. Take for example the foundational ProxyNCA objective [38]. ProxyNCA is heavily connected to various recent, state-of-the-art objectives (such as ProxyAnchor [24] or SoftTriple [43]) and has the form

$$\mathcal{L}_{\text{PNCA}} = -\mathbb{E}_{\substack{x\sim\mathcal{X}_y \\ y\sim\mathcal{Y}}}\left[\log\left(\frac{e^{s(\psi(x),\rho_y)}}{\sum_{\rho^*\in\mathcal{P}^{-y}} e^{s(\psi(x),\rho^*)}}\right)\right] \tag{1}$$

with the complete set of class proxies $\mathcal{P}$ and with class y removed $\mathcal{P}^{-y}$[1] that are trained jointly during training. Minimizing the distance of samples to their respective class proxies while maximizing it for non-related proxies, this objective can be regarded as implicitly maximizing the log-likelihood of samples ψ belonging to proxy ρ (such that $y_\psi = y_\rho$) under a von-Mises-Fisher (vMF[2]) mixture model [14, 17] around directions ρ

$$p_{\text{vMFmm}}(\psi|\rho) = \frac{\pi_\rho C_d(\kappa_\rho) e^{\kappa_\rho s(\psi,\rho)}}{\sum_{\rho^*\in\mathcal{P}} \pi_{\rho^*} C_d(\kappa_{\rho^*}) e^{\kappa_{\rho^*} s(\psi,\rho^*)}} \tag{2}$$

$$C_d(\kappa) = \kappa^{d/2-1} \cdot \left[(2\pi)^{d/2} I_{d/2-1}(\kappa)\right]^{-1} \tag{3}$$

assuming a class-independent concentration parameter $\kappa_\rho = \kappa$ and mixture $\pi_\rho = \pi$ such that $C_d(\kappa_\rho) = C_d(\kappa) = const$[3]. Even more, [56] show that performance of $\mathcal{L}_{\text{PNCA}}$ improves when actually optimize the proxy-assignment probability (by replacing $\mathcal{P}^{-y}$ with $\mathcal{P}$ in the denominator, giving $\mathcal{L}_{\text{PNCA++}}$) directly, which equals an explicit negative log-likelihood minimization of p_{vMFmm}:

$$\mathcal{L}_{\text{PNCA++}} = -\mathbb{E}_{x\sim\mathcal{X}_y, y\sim\mathcal{Y}}\left[\log p_{\text{vMFmm}}(\psi(x)|\rho_y)\right] \tag{4}$$

Recent and state-of-the-art proxy objectives that extend upon ProxyNCA, such as the ProxyAnchor [23] objective

$$\begin{aligned}\mathcal{L}_{\text{PA}} = &\frac{1}{|\mathcal{P}^+|}\sum_{\rho\in\mathcal{P}^+}\log\left(1+\sum_{x\in\mathcal{B}, y_x=y_\rho} e^{-\alpha\cdot[s(x,\rho)-\delta]}\right) \\ &+\frac{1}{|\mathcal{P}|}\sum_{\rho\in\mathcal{P}}\log\left(1+\sum_{x\in\mathcal{B}, y_x\neq y_\rho} e^{\alpha\cdot[s(x,\rho)+\delta]}\right)\end{aligned} \tag{5}$$

operate under similar assumptions, suggesting slight, more hyperparameter-heavy variations to the loss terms. While ProxyAnchor specifically suggests pulling samples towards proxies instead of proxies towards samples as done in $\mathcal{L}_{\text{PNCA}}$, it similarly relies on the same sample-proxy contrastive operations to learn a respective metric space. This means that these methods can only learn sample distributions around proxies that are closely related to p_{vMFmm}-like distributions learned via $\mathcal{L}_{\text{PNCA}}$. Indeed, our experiments (see §4.2 and Tab. 1) show that when adapting the ProxyNCA objective to the ProxyAnchor formulation

$$\begin{aligned}\mathcal{L}^*_{\text{PNCA}} = &\frac{1}{|\mathbb{B}^+|}\sum_{x\in\mathcal{B}}\log\left(1+e^{-\alpha\cdot[s(x,\rho_{y_x})-\delta]}\right) \\ &+\frac{1}{|\mathcal{B}|}\sum_{x\in\mathcal{X}}\log\left(1+\sum_{\rho\in\mathcal{P}, y_x\neq y_\rho} e^{\alpha\cdot[s(x,\rho)+\delta]}\right)\end{aligned} \tag{6}$$

performance becomes much more similar than indicated in [23]. And while ProxyAnchor may not optimize for the

[1] We use cosine similarity instead of the euclidean distance as done in [38, 56], as both are equivalent on the hypersphere.

[2] An assumption found also e.g. in self-supervised learning [70].

[3] C_d incorporates the modified Bessel function I_p of the first kind and order p, which can be neglected here as C_d cancels out

exact $p(\psi|\rho)$ formulation, the results support a very strong distributional relation between these proxy-objectives.

However, mixture distributions such as p_{vMFmm} suffer from several issues. Firstly, each mode on its own is isotropic as $s(\psi, \rho)$ returns the same value as long as the angle $\theta(\psi, \rho)$ is retained. This means that class-specific structures can only be resolved implicitly through relations with proxies of different classes. Secondly, this intraclass resolution becomes worse as training progress, since proxies from different classes continue to contrast further and sample-proxy relations for same-class pairs are overshadowed (see e.g. Eq. 1, 5, 6). Similarly, for samples closer towards each respective class proxy, resolving local structures becomes harder. Effectively, this results in learned sample distributions to have a strong affinity towards local isotropy.

As such, proxy-based objectives are inherently handicapped in resolving local intraclass clusters and structures. Consequently, semantic relations between samples within a class are not well encoded in the representational structure of the learned deep metric spaces. However, the ability to explain and represent intraclass sample relations has been consistently shown to be a key driver for downstream generalization performance in DML [24, 31, 34, 35, 46, 49].

While sample-based objectives similarly suffer from the non-bijectivity of $s(\bullet, \bullet)$, the usage of sample-to-sample operations explicitly introduces intersample relational context [49] that allows for better structuring of samples within a class. However, just incorporating a sample-based contrastive operation into the training process is not a sufficient remedy as it re-introduces the sampling complexity issue; whose removal was what made proxy-based methods and their fast convergence attractive in the first place.

3.2. Non-isotropy Regularization

Motivation. To address the non-learnability of intraclass context in proxy-based DML, we therefore have to address the inherent issue of local isotropy in the learned sample-proxy distribution $p(\psi|\rho)$. However, to retain the convergence (and generalization) benefits of proxy-based methods, this has to be achieved without resorting to additional augmentations that move the overall objective away from a purely proxy-based one. As such, we aim to find $p(\psi|\rho)$ whose optimization better resolves the distribution of sample representations ψ around our proxy ρ. This can be achieved by breaking down the fundamental issue of non-bijectivity in the used similarity measure $s(\psi, \rho)$, which (on its own) introduces non-unique sample-proxy relations. To do so, we look for some regularization that specifically encourages unique sample-proxy relations to exists. For such **unique** sample-proxy relations to exist, we must have access to some **bijective** and thus **invertible** (deterministic) translation $\psi = \tau(\zeta|\rho)$ which, given some residual ζ from some prior distribution $q(\zeta)$, allows to uniquely translate from the respective proxy ρ to ψ. Given such a unique translation of samples and proxies within a class, the local alignment of samples would then no longer rely on relations to proxies and samples from different classes which, as noted, do not scale well locally and as training progresses. Assuming proxies and sample representation to have the same dimensionality, this can be achieved through some affine transformation. However, to capture non-linear relations and proxy-to-sample translations, it is much more beneficial for τ to be non-linear.

Normalizing Flows. Such invertible, non-linear functions are naturally expressed through Normalizing Flows (NF) or more generally invertible neural networks [1, 2, 9, 27, 44, 45, 50]. A Normalizing Flow can be generally seen as a transformation between two probability distributions, most commonly between simple, well-defined ones and complex multimodal ones [8, 9, 27, 41]. More specifically, we leverage flows similar to the one proposed in [9] and [27] (and as used e.g. in [1, 2, 10, 45, 50]), which introduces a sequence of non-linear, but still invertible coupling operations as showcased in Fig. 2 ($\leftrightharpoons$). Given some input representation ψ, a coupling block splits ψ into ψ_1 and ψ_2, which are scaled and translated in succession with non-linear scaling and translation networks η_1 and η_2, respectively. Note that following [27], each network η_i provides both scaling η_i^s and translation values η_i^t, such that

$$\begin{aligned} \psi_2^* &= \psi_2 \odot \exp\left(\eta_1^s(\psi_1)\right) + \eta_1^t(\psi_1) \\ \psi_1^* &= \psi_1 \odot \exp\left(\eta_2^s(\psi_2^*)\right) + \eta_2^t(\psi_2^*) \\ \psi^* &= [\psi_1^*, \psi_2^*] \end{aligned} \tag{7}$$

where ψ^* denotes ψ after passing through a respective coupling block. Successive application of different η_i then gives our non-linear invertible transformation τ from some prior distribution over residuals $q(\zeta)$ with explicit density and CDF (for sampling) to our target distribution.

Enforcing non-isotropy. Consequently, our bijective τ (conditioned on proxies ρ) induces a new sample representation distribution $p(\tau(\zeta|\rho)|\rho)$ as *pushforward* from our prior distribution of residuals $q(\zeta)$ which accounts for unique sample-to-proxy relations, and which we wish to impose over our learned sample distribution $p(\psi|\rho)$. This introduces our $\mathbb{N}$on-$\mathbb{I}$sotropy $\mathbb{R}$egularization ($\mathbb{NIR}$). $\mathbb{NIR}$ can be naturally approached through maximization of the expected log-likelihood $\mathbb{E}_{x,\rho_{y_x}}\left[\log p\left(\psi(x)|\rho_{y_x}\right)\right]$ over sample-proxy pairs (x, ρ_{y_x}) similar to Eq. 4, but under the constraint that each distribution of samples around a respective proxy, $p(\psi|\rho)$, is a *pushforward* of τ from our residual distribution $q(\zeta)$. This gives (see e.g. [29])

$$\begin{aligned} \mathcal{L}_{\mathbb{NIR}} = -\mathbb{E}_{x,\rho_{y_x}}\big[&\log q\left(\tau^{-1}(\psi(x)|\rho_{y_x})\right) \\ &+ \log\left|\det J_{\tau^{-1}}(\tau^{-1}(\psi(x)|\rho_{y_x})|\rho_{y_x})\right|\big] \end{aligned} \tag{8}$$

with Jacobian J for translation τ^{-1} and proxies ρ_{y_x}, where

y_x denotes the class of sample x. To arrive at above equation, we simply leveraged the change of variables formula

$$p(\psi|\rho) = q(\tau^{-1}(\psi|\rho))|\det J_{\tau^{-1}}(\tau^{-1}(\psi|\rho)|\rho)|. \quad (9)$$

In practice, by setting our prior $q(\zeta)$ to be a standard zero-mean unit-variance normal distribution $\mathcal{N}(0,1)$, we get

$$\begin{aligned}\mathcal{L}_{\text{NIR}} = \frac{1}{|\mathcal{B}|}\sum_{(x,\rho_{y_x})\sim\mathcal{B}} \left\|\tau^{-1}(\psi(x)|\rho_{y_x})\right\|_2^2 \\ - \log|\det J_{\tau^{-1}}(\tau^{-1}(\psi(x)|\rho_{y_x})|\rho_{y_x})|\end{aligned} \quad (10)$$

i.e. given sample representations $\psi(x)$, we project them onto our residual space ζ via τ^{-1} and compute Eq. 10. By selecting suitable normalizing flows such as GLOW [27], we make sure that the Jacobian is cheap to compute.

NIR for proxy-based DML. As NIR targets the alignment of samples around proxies, we still need to learn the global alignment of proxies through proxy-based DML $\mathcal{L}_{\text{PDML}}(\psi, \mathcal{P})$. This gives the full training objective

$$\mathcal{L} = f(\mathcal{L}_{\text{NIR}}) + \omega \cdot \mathcal{L}_{\text{PDML}}(\psi, \mathcal{P}) \quad (11)$$

where $f(\cdot)$ is a monotonous function of $\mathcal{L}_{\text{NIR}}$ to match the scaling and training dynamics of $\mathcal{L}_{\text{PDML}}$ without changing the invertibility constraint. As most proxy-based $\mathcal{L}_{\text{PDML}}$ utilize exponential components, we simply use $f(\cdot) = \exp(\cdot)$. Full optimization over $\mathcal{L}$ then learns proxies while uniquely resolving sample placement around them. More specifically, backpropagating through $\mathcal{L}_{\text{NIR}}$ optimizes for sample alignment around proxies ρ, the translation τ and provides updates to the proxies, though we found the latter to not be a necessity, as proxies primarily serve to resolve global alignment of sample clusters.

NIR-proxy-DML has several advantages. Firstly, the final sample-proxy distribution optimized for directly addresses issues of local isotropy to better resolve local intra-class structure, as the retention of a unique sample distribution around each proxies requires implicit knowledge about the intraclass alignment of each class sample around their respective proxies. Secondly, unlike ProxyNCA-like objectives (see the previous section), we do not assume the same concentration of samples per class as assumed in e.g. Eq. 1. Instead, the non-linear translation conditioned on the class proxy can introduce class-dependent concentrations when needed. Finally, being able to directly resolve local structures can potentially benefit convergence of these methods.

4. Experiments

This section lists experimental details (§4.1), showcases significant benefits of NIR for proxy-based DML (§4.2) and highlights the impact on convergence and training times in §4.3. We also study quantitative impacts on learned representation spaces (§4.4), provide method ablations in §4.5 and investigate self-regulatory properties of NIR (§4.6).

4.1. Experimental Details

Implementations use `PyTorch` [42]. ImageNet [6]-pretrainings are taken from `torchvision` [33] and `timm` [62]. Our experiments were run on compute servers with NVIDIA 2080Ti. Our Normalizing Flow utilizes 8 coupling blocks and subnets η comprising linear layers with 128 nodes. Optimization is done using Adam [26] (learning rate 10^{-5}, weight decay $4 \cdot 10^{-3}$, [49]). We set $\omega \in [0.001, 0.01]$ depending on the choice for $\mathcal{L}_{\text{PDML}}$. In general, we found consistent improvements in this interval. Following [23, 38, 56] we utilize a high learning rate multiplication for the proxies (4000). We also saw this helping the Normalizing Flow and use 50 for all experiments. Finally, we found a warmup epoch to help; adapting the translation τ over pretrained features first before joint training.

Datasets. We use the standard benchmarks CUB200-2011 [57] (11,788 bird images, 200 classes), CARS196 [30] (16,185 car images, 196 classes) and Stanford Online Products [40] (SOP, 120,053 images, 22,634 product instances).

4.2. Effectiveness of Non-isotropy Regularisation

To evaluate the relative benefits of NIR, we follow protocols proposed in [49] to encourage exact comparability with no learning rate scheduling. Initial tuning of newly introduced hyperparameters is done on a random 15% validation split (see e.g. [23,49]). Note that we don't perform joint hyperparameter tuning of auxiliary proxy objectives $\mathcal{L}_{\text{DML}}$ and NIR to see how well NIR can be applied to blackbox proxy approaches. For $\mathcal{L}_{\text{DML}}$, we select ProxyAnchor [23], SoftTriplet [43](10 centroids for CUB200-2011/CARS196 and 2 for SOP) and ProxyNCA [38] following Eq. 6. Results over multiple seeds are provided in Table 1, showing that generalization significantly improves for all proxy objectives across metrics and benchmarks, even for objectives with more than one proxy per class s.a. SoftTriple. For the latter, we find NIR to also improve convergence properties as shown in §4.3. Especially for datasets where a reasonable amount of samples per class is available (s.a. CUB200-2011 & CARS196) to learn meaningful class distributions we see major improvements, e.g. for state-of-the-art ProxyAnchor from 82.4% to 85.2 & 64.4 to 66.0% on CARS196 & CUB200-2011, respectively. However, even for datasets such as SOP with a small number of samples per class a consistent performance improvement can be seen, highlighting the general benefit of NIR for proxy-based DML.

To compare to the overall corpus of DML methods, we also provide a literature comparison in Table 2, with approaches divided based on backbone architecture and embedding dimensionality; both of which drive generalization performance independent of DML objectives [39, 49]. Results reported here use stepwise learning rate scheduling at most twice, with parameters determined by performance on a random, 15% validation subset [23, 34, 49]. As can

Table 1. *Relative comparison.* We follow protocols proposed in [49][4], with no learning rate scheduling, to ensure exact comparability. The results show significant improvements over very strong proxy objectives on all benchmarks, but especially on CUB200 and CARS196 where a more significant number of samples per class is available.

BENCHMARKS→	CUB200-2011			CARS196			SOP		
APPROACHES ↓	R@1	NMI	mAP@1000	R@1	NMI	mAP@1000	R@1	NMI	mAP@1000
Multisimilarity	62.8 ± 0.2	67.8 ± 0.4	31.1 ± 0.3	81.6 ± 0.3	69.6 ± 0.5	31.7 ± 0.1	76.0 ± 0.1	89.4 ± 0.1	43.3 ± 0.1
ProxyAnchor [23]	64.4 ± 0.3	68.4 ± 0.2	33.2 ± 0.3	82.4 ± 0.4	69.0 ± 0.3	34.2 ± 0.3	78.0 ± 0.1	90.1 ± 0.1	45.5 ± 0.1
+ NIR	66.0 ± 0.3	69.6 ± 0.1	34.2 ± 0.2	85.2 ± 0.3	71.6 ± 0.3	36.4 ± 0.2	78.9 ± 0.1	90.4 ± 0.1	46.5 ± 0.1
ProxyNCA [38]	64.2 ± 0.2	68.6 ± 0.3	33.1 ± 0.2	82.1 ± 0.4	68.2 ± 0.2	32.4 ± 0.5	78.3 ± 0.1	90.0 ± 0.1	45.5 ± 0.1
+ NIR	66.1 ± 0.2	69.8 ± 0.2	34.3 ± 0.1	84.3 ± 0.3	70.6 ± 0.6	34.5 ± 0.3	79.1 ± 0.1	90.2 ± 0.1	46.2 ± 0.1
SoftTriplet [43]	62.3 ± 0.3	68.2 ± 0.2	31.6 ± 0.2	80.7 ± 0.2	66.4 ± 0.3	30.4 ± 0.2	76.9 ± 0.2	89.6 ± 0.1	43.5 ± 0.1
+ NIR	63.8 ± 0.4	68.5 ± 0.2	34.0 ± 0.4	83.4 ± 0.4	68.8 ± 0.5	35.5 ± 0.2	77.6 ± 0.1	90.0 ± 0.1	44.9 ± 0.1

Table 2. *Literature comparison* using ProxyAnchor (PA) + NIR. Across backbones/dim. and benchmarks, we find competitive and even state-of-the-art performance against much more complex methods. [X]: Combination of pooling operations in backbone as done in [23]. **Bold** denotes best in a given *Architecture/Dimensionality* setting. **Bluebold** denotes best overall.

BENCHMARKS →			CUB200 [57]			CARS196 [30]			SOP [40]		
METHODS ↓	Venue	Arch/Dim.	R@1	R@2	NMI	R@1	R@2	NMI	R@1	R@10	NMI
Div&Conq [51]	*CVPR '19*	R50/128	65.9	76.6	69.6	84.6	90.7	70.3	75.9	88.4	90.2
MIC [46]	*ICCV '19*	R50/128	66.1	76.8	69.7	82.6	89.1	68.4	77.2	89.4	90.0
PADS [47]	*CVPR '20*	R50/128	67.3	78.0	69.9	83.5	89.7	68.8	76.5	89.0	89.9
RankMI [21]	*CVPR '20*	R50/128	66.7	77.2	**71.3**	83.3	89.8	69.4	74.3	87.9	90.5
PA+NIR	-	R50/128	66.9 ± 0.2	77.7 ± 0.3	69.8 ± 0.2	85.3 ± 0.2	91.1 ± 0.2	72.1 ± 0.2	**79.6 ± 0.1**	**90.7 ± 0.1**	**90.5 ± 0.1**
	-	R50/128[X]	**67.9 ± 0.2**	**78.3 ± 0.2**	**71.4 ± 0.4**	**86.5 ± 0.3**	**92.0 ± 0.2**	**72.7 ± 0.2**	79.4 ± 0.1	**90.7 ± 0.1**	**90.6 ± 0.1**
ProxyAnchor (PA) [23]	*CVPR '20*	IBN/512[X]	68.4	79.2	-	86.8	91.6	-	79.1	90.8	-
ProxyGML [69]	*NeurIPS '20*	IBN/512	66.6	77.6	69.8	85.5	91.8	72.4	78.0	90.6	90.2
DRML [68]	*ICCV '21*	IBN/512	68.7	78.6	69.3	86.9	92.1	72.1	71.5	85.2	88.1
PA + *MemVir* [28]	*ICCV '21*	IBN/512	69.0	79.2	-	86.7	92.0	-	**79.7**	**91.0**	-
PA+NIR	-	IBN/512	69.4 ± 0.2	79.7 ± 0.2	**71.1 ± 0.1**	87.1 ± 0.2	92.5 ± 0.1	73.1 ± 0.2	79.4 ± 0.1	90.5 ± 0.1	90.3 ± 0.2
	-	IBN/512[X]	**70.1 ± 0.1**	**80.1 ± 0.2**	**71.0 ± 0.2**	**87.9 ± 0.2**	**92.8 ± 0.1**	**73.7 ± 0.2**	79.3 ± 0.1	90.4 ± 0.1	90.2 ± 0.2
EPSHN [64]	*WACV '20*	R50/512	64.9	75.3	-	82.7	89.3	-	78.3	90.7	-
Circle [55]	*CVPR '20*	R50/512	66.7	77.2	-	83.4	89.7	-	78.3	90.5	-
DiVA [34]	*ECCV '20*	R50/512	69.2	79.3	71.4	87.6	92.9	72.2	79.6	91.2	90.6
DCML-MDW [67]	*CVPR '21*	R50/512	68.4	77.9	71.8	85.2	91.8	73.9	79.8	90.8	**90.8**
PA+NIR	-	R50/512	69.1 ± 0.2	79.6 ± 0.2	72.0 ± 0.2	87.7 ± 0.2	92.5 ± 0.1	74.2 ± 0.2	**80.7 ± 0.1**	**91.5 ± 0.1**	**90.9 ± 0.1**
	-	R50/512[X]	**70.5 ± 0.1**	**80.6 ± 0.2**	**72.5 ± 0.3**	**89.1 ± 0.2**	**93.4 ± 0.2**	**75.0 ± 0.3**	80.4 ± 0.1	91.4 ± 0.2	90.6 ± 0.1

be seen, NIR-equipped ProxyAnchor achieves competitive performance across settings and benchmarks with a new highest overall score. In addition, PA+NIR beats much more complex methods such as *DiVA* [34] using joint multi-task and self-supervised training or MIC [46] using external feature mining. This supports the benefit of learning global alignments with proxies while jointly refining local sample alignment. Taking into account that NIR retains the superior convergence of proxy-based approaches (see §4.3), this makes NIR very attractive for practical usage, and provides a strong proof-of-concept on the benefits of intraclass context for proxy-DML.

4.3. Convergence Properties

A primary motivation for NIR, besides generalization improvements, is to retain fast convergence speeds. We investigate this following the same setup used for Tab. 1 (§4.2). Results visualized in Fig. 3 show mean test generalisation performance after every epoch [23, 38, 51] (unlike Tab. 1 showing overall mean test performance) for every auxiliary proxy objective. Besides significant improvements in generalisation performance, convergence speeds and behaviour are either retained or even improved e.g. for SoftTriple, presumably due to better resolved class structures allowing for better alignment of multiple learned class centroids. Furthermore, the addition of the Normalizing Flow adds only limited additional computational overhead since we operate in feature space. Especially with respect to the large backbone, we find changes in walltime and required additional GPU memory to be negligible ($< 1\%$).

4.4. Qualitative differences in alignment

In this section, we investigate how NIR changes the structural properties of the representation spaces learned by proxy-based objectives. For our experiments, we select ProxyAnchor as stand-in for $\mathcal{L}_{\text{PDML}}$. We then compare the structure of representation space learned with and without NIR by looking at different structural metrics: **(1)**

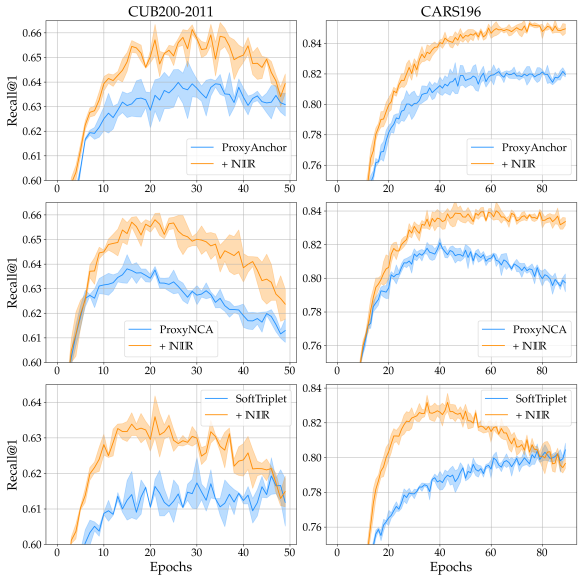

Figure 3. *Impact of* $\mathbb{NIR}$ *on convergence.* We find that $\mathbb{NIR}$ increases generalisation performance while retaining or even improving the fast convergence behaviour.

Table 3. *Change in Structural Properties.* Applying $\mathbb{NIR}$ increase feature diversity ρ and uniformity of learned representations G_2, reduces overclustering π_{density} and encourages a higher degree in class concentration difference between classes (σ_κ^2).

Dataset	**Setup**	$\rho \downarrow$	$\pi_{\text{density}} \uparrow$	$\sigma_\kappa^2 \uparrow$	$G_2 \downarrow$
CUB200	PA	0.19 ± 0.01	0.68 ± 0.04	0.37 ± 0.02	0.078 ± 0.001
	+ $\mathbb{NIR}$	$\mathbf{0.13} \pm 0.02$	$\mathbf{0.79} \pm 0.03$	$\mathbf{0.44} \pm 0.02$	$\mathbf{0.072} \pm 0.002$
CARS196	PA	0.17 ± 0.01	0.59 ± 0.01	0.32 ± 0.01	0.079 ± 0.001
	+ $\mathbb{NIR}$	$\mathbf{0.13} \pm 0.01$	$\mathbf{0.68} \pm 0.02$	$\mathbf{0.38} \pm 0.02$	$\mathbf{0.072} \pm 0.001$

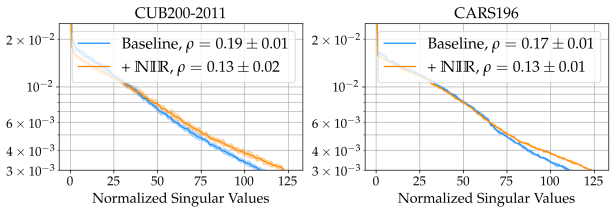

Figure 4. *Singular Value Distribution* used to estimate feature diversity ρ, showing $\mathbb{NIR}$ to introduce more directions of variance.

Feature richness measured by spectral decay [49] $\rho(\Psi) = \text{KL}(\mathcal{U}||S(\Psi))$ with singular value decomposition $S(\Psi)$ of feature space Ψ. $\rho(\Psi)$ measures the number of significant directions of variance in the learned feature space - lower scores indicate a higher feature variety, linked to improved generalisation in [34, 49]. **(2)** Representational Uniformity/Density [49] $\pi_{\text{density}} = {}^{\pi_{\text{intra}}}/_{\pi_{\text{inter}}}$, which measures the ratio of mean intraclass and interclass distance π_{intra} and π_{inter}, respectively. π_{density} relates class concentration against overall alignment across the hypersphere. Higher values indicate lower class concentrations and overclustering, a potential link to better generalisation [49]. **(3)** Embedding space uniformity [60] $G_{t=2}(u,v) = e^{-t\|u-v\|_2^2}$ evaluating uniformity of embedding spaces measured by a Radial Basis Function kernel [3]. Lower values have been linked in [60] to improved downstream performance in contrastive self-supervised learning. **(4)** Variance of class-concentrations σ_κ^2 approximated by the mean distances to the class centers of mass (relative to the mean interclass distance π_{inter} to account for the overall scale of representations). As $\mathbb{NIR}$ allows for different class-conditional distributions to be learned, we assume a higher σ_κ^2.

Table 4. *Structural Ablations.*

BENCHMARKS→	CUB200-2011		CARS196	
SETUPS ↓	R@1	mAP @1000	R@1	mAP @1000
PA + $\mathbb{NIR}$	66.0 ± 0.3	34.2 ± 0.2	85.2 ± 0.2	36.4 ± 0.2
(a) Normalizing Flows Training				
$f(\cdot)$ = SoftPlus	66.3 ± 0.2	34.0 ± 0.2	85.1 ± 0.2	36.3 ± 0.3
$f(\cdot, t = 0.3)$ = Exp	66.1 ± 0.2	34.1 ± 0.1	85.0 ± 0.2	36.2 ± 0.2
$f(\cdot, t = 3)$ = Exp	65.8 ± 0.3	33.8 ± 0.2	84.8 ± 0.3	36.1 ± 0.2
Grad. Clipping	65.9 ± 0.1	34.2 ± 0.1	85.3 ± 0.1	36.5 ± 0.1
No Proxy Backprop	66.2 ± 0.4	34.1 ± 0.3	85.0 ± 0.2	36.3 ± 0.1
w/ Negative Pairs	64.9 ± 0.3	33.8 ± 0.3	83.5 ± 0.4	34.9 ± 0.3
$\omega = 0$	60.0 ± 0.4	30.1 ± 0.3	73.5 ± 0.6	27.2 ± 0.5
(b) Normalizing Flows Architecture				
D15 - W64	66.5 ± 0.5	34.0 ± 0.2	85.1 ± 0.4	36.6 ± 0.3
D5 - W512	66.1 ± 0.3	34.1 ± 0.1	84.9 ± 0.2	36.1 ± 0.2
D3 - W1024	65.8 ± 0.4	34.1 ± 0.2	85.3 ± 0.1	36.3 ± 0.2
Condition - Start	66.4 ± 0.4	34.1 ± 0.2	85.1 ± 0.2	36.2 ± 0.3
Condition - Mid	66.1 ± 0.2	34.1 ± 0.2	84.9 ± 0.2	36.0 ± 0.2
Condition - End	65.7 ± 0.2	33.8 ± 0.3	84.8 ± 0.3	36.0 ± 0.2

Results in Tab. 3 indeed show higher feature diversity ρ (over 30%, cf. Fig. 4 for the sorted singular value spectra), reduced overclustering ($> 15\%$) as measured by π_{density} and increased uniformity in learned representation spaces (up to 9% as evaluated via G_2) - all linked to better generalization as noted above. This links well with our initial motivation for $\mathbb{NIR}$ to allow for better explicit resolution of local structures and clusters, which requires local separability of representations and the introduction of auxiliary features [34, 49]. We also find that the variance of class-concentrations increases, supporting that $\mathbb{NIR}$ helps proxy-based objectives learn class-dependent sample distributions.

4.5. Ablations

We now ablate $\mathbb{NIR}$. Results are provided in Tab. 4. We use ProxyAnchor as stand-in for $\mathcal{L}_{\text{PDML}}$

(a) Training Normalizing Flows. We first ablate the scaling $f(\cdot)$ (§3.2, Eq. 10). As can be seen, the exact choice of exponential function does not matter, with changes in temperature ($f(\cdot, t)$) or a Softplus ($f(x) = \log(1 + \exp x)$) performing similarly. We also experiment with gradient clipping ("*Grad. Clipping*"), but found no benefits. Furthermore, we investigate joint negative log-likelihood (NLL) *maximization* for sample-proxy-pairs with different classes

Table 5. *Self-Regularisation* by reversing τ to generate synthetic samples does not benefit generalisation.

BENCHMARKS→	CUB200-2011		CARS196	
APPROACH ↓	R@1	mAP @1000	R@1	mAP @1000
PA + $\mathbb{NIR}$	66.0 ± 0.3	34.2 ± 0.2	85.2 ± 0.2	36.4 ± 0.2
Generate	64.8 ± 0.4	33.0 ± 0.3	84.6 ± 0.6	35.9 ± 0.3
Reverse Match	63.2 ± 0.1	31.0 ± 0.1	83.4 ± 0.1	32.8 ± 0.1
Generate & Match	63.5 ± 0.2	31.6 ± 0.2	83.9 ± 0.4	35.7 ± 0.3

("*w/ Negative Pairs)*" following Eq. 10, but found no benefits over just minimizing the NLL for same-class pairs. This supports our hypothesis that the benefits of $\mathbb{NIR}$ indeed lie in improved resolution of class-local structures. This is also supported through the minimal impact of proxy updates through $\mathcal{L}_{\mathbb{NIR}}$ ("*No Proxy Backprop*"), as proxies primarily help with global cluster alignment while $\mathbb{NIR}$ is developed for local refinement. Completely removing $\mathcal{L}_{\text{PDML}}$ ("$\omega = 0$") gives similar insights - while we see decent performance solely through non-isotropy regularisation, the absence of a global alignment objective for proxies incorporated through $\mathcal{L}_{\text{PDML}}$ ("$\omega = 0$") results in a notable performance drop and reduction in convergence speeds.

(b) Normalizing Flows Architecture. We ablate the number of coupling blocks (*D)* and subnets widths (*W*) used in $\mathbb{NIR}$ and find dataset-dependent optima with slight improvements over the default (§4.1). For consistency, we report all other results using the default setup. We also examine the conditioning of the Normalizing Flow - inserting proxies at the start, mid or end ("*Condition start/mid/end*", as opposed to every coupling block by default). As can be seen, the exact choice of conditioning influences generalization performance somewhat, but with significant improvements regardless of the exact conditioning. Overall, these results show that in an applied setting, further improvements can be found by more aggressive, but not necessarily principled hyperparameter tuning of $\mathbb{NIR}$.

4.6. Self-regularization

Finally, we study a natural extension to $\mathbb{NIR}$ by leveraging the generative process defined through our Normalizing Flow, which provides a translation from a probability density we can sample from (in this case just $\mathcal{N}(0,1)$) to the representation space for a respective class y (conditioned on ρ_y). Similar to [31] or [66], which saw benefits in generalisation through synthetic samples in sample-based DML, we investigate in Tab. 5 whether sampling from the residual prior $\zeta \sim q(\cdot)$ and traversing the Normalizing Flow in reverse can generate synthetic samples that offer additional self-regularisation. More specifically, we investigate whether synthetic samples $\psi^s = \tau(\zeta|\rho)$ can be used in $\mathcal{L}_{\text{PDML}}(\psi^{s^X}, \mathcal{P})$ to learn more generic proxies ("*Generate*"), detach the synthetic samples $(^X)$. Similarly, we investigate whether the generated samples can be used to refine the quality of the Normalizing Flow by evaluating the matching quality with the learned proxies via $\mathcal{L}_{\text{PDML}}(\psi^s, \mathcal{P}^X)$ ("*Match*"), as well as doing both steps jointly ($\mathcal{L}_{\text{PDML}}(\psi^s, \mathcal{P})$, "*Generate & Match*). In its current setting, generalization and convergence suffer from artificial samples. Especially for the former, we see drops in performance from 66.0% back down to 64.8% when introducing artificial samples and even 63.2% when applying reverse distribution matching. We hypothesize that this is due to the introduction of noisy samples especially in earlier training stages and the interdependence of $\mathbb{NIR}$ on the quality of the learned proxies. We believe a better adapted incorporation following e.g. a hardness-aware heuristic [66] could better leverage the benefits of such self-regularization. We leave this to future work to address.

5. Conclusion

This work proposes $\mathbb{NIR}$ - non-isotropy regularisation for proxy-based Deep Metric Learning (DML). $\mathbb{NIR}$ tackles the inherent problem of proxy-based objectives to resolve local structures and clusters to learn non-discriminative features that facilitate generalization, without influencing the superior convergence of proxy-based DML. $\mathbb{NIR}$ achieves this by refining the sample-distributional prior optimized in standard proxy-based DML through unique sample-proxy relation constraints. Extensive experiments support the idea of $\mathbb{NIR}$ and, besides the retention of fast convergence speeds, show significant improvements in the generalisation performance of proxy-based objectives, achieving competitive and state-of-the-art performance on all benchmarks.

Limitations. $\mathbb{NIR}$ relies on learning meaningful translations from (class-)proxies to respective samples. With high proxy count, the quality of these translations is impacted, evident in the performance on SOP. In addition, our current setting can not yet leverage the sample-generative process induced by the Normalizing Flows for additional regularization (which is also bottlenecked by high proxy-counts/low number of samples per class).

Broader Impact. Our work significantly benefits proxy-based DML, making application in DML-driven domains s.a image & video retrieval, but also face ReID, very attractive. For the latter, a potential for misuse given. However, while notable, improvements through $\mathbb{NIR}$ are not sufficient to drive a significant change in the societal usage in these domains.

Acknowledgements. This work has been partially funded by the ERC (853489 - DEXIM) and DFG (2064/1 – Project number 390727645). K.R. thanks the International Max Planck Research School for Intelligent Systems (IMPRS-IS) and the European Laboratory for Learning and Intelligent Systems (ELLIS) PhD program for support.

References

[1] Lynton Ardizzone, Jakob Kruse, Carsten Lüth, Niels Bracher, Carsten Rother, and Ullrich Köthe. Conditional invertible neural networks for diverse image-to-image translation. *CoRR*, abs/2105.02104, 2021. 4

[2] Lynton Ardizzone, Carsten Lüth, Jakob Kruse, Carsten Rother, and Ullrich Köthe. Conditional invertible neural networks for guided image generation, 2020. 2, 4

[3] Piotr Bojanowski and Armand Joulin. Unsupervised learning by predicting noise. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 517–526. JMLR.org, 2017. 7

[4] 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. 1, 2

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Everest Hinton. A simple framework for contrastive learning of visual representations. 2020. 1, 2

[6] 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. 5

[7] J. Deng, J. Guo, N. Xue, and S. Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4685–4694, 2019. 1, 2

[8] Laurent Dinh, David Krueger, and Yoshua Bengio. NICE: non-linear independent components estimation. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Workshop Track Proceedings*, 2015. 2, 4

[9] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real NVP. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. 4

[10] Michael Dorkenwald, Timo Milbich, Andreas Blattmann, Robin Rombach, Konstantinos G. Derpanis, and Bjorn Ommer. Stochastic image-to-video synthesis using cinns. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3742–3753, June 2021. 4

[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. 2

[12] Natalie Dullerud, Karsten Roth, Kimia Hamidieh, Nicolas Papernot, and Marzyeh Ghassemi. Is fairness only metric deep? evaluating and addressing subgroup gaps in deep metric learning. In *International Conference on Learning Representations*, 2022. 2

[13] Weifeng Ge. Deep metric learning with hierarchical triplet loss. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 269–285, 2018. 2

[14] Siddharth Gopal and Yiming Yang. Von mises-fisher clustering models. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 154–162, Bejing, China, 22–24 Jun 2014. PMLR. 3

[15] 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. 2

[16] 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. 1, 2

[17] Md. Abul Hasnat, Julien Bohné, Jonathan Milgram, Stéphane Gentric, and Liming Chen. von mises-fisher mixture model-based deep learning: Application to face verification. *CoRR*, abs/1706.04264, 2017. 3

[18] 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. 1, 2

[19] J. Hu, J. Lu, and Y. Tan. Discriminative deep metric learning for face verification in the wild. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014. 1, 2

[20] 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. 2

[21] Mete Kemertas, Leila Pishdad, Konstantinos G. Derpanis, and Afsaneh Fazly. Rankmi: A mutual information maximizing ranking loss. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 6

[22] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning, 2020. 1, 2

[23] 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. 1, 2, 3, 5, 6

[24] Sungyeon Kim, Dongwon Kim, Minsu Cho, and Suha Kwak. Embedding transfer with label relaxation for improved metric learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3967–3976, June 2021. 3, 4

[25] 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. 2

[26] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 5

[27] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. 4, 5

[28] Byungsoo Ko, Geonmo Gu, and Han-Gyu Kim. Learning with memory-based virtual classes for deep metric learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11792–11801, October 2021. 6

[29] Ivan Kobyzev, Simon Prince, and Marcus A. Brubaker. Normalizing flows: An introduction and review of current methods. 2019. cite arxiv:1908.09257Comment: Updated version, currently under review in a journal. 4

[30] 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. 2, 5, 6

[31] 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. 2, 4, 8

[32] 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. 1, 2

[33] Sébastien Marcel and Yann Rodriguez. Torchvision the machine-vision package of torch. In *Proceedings of the 18th ACM International Conference on Multimedia*, MM '10, page 1485–1488, New York, NY, USA, 2010. Association for Computing Machinery. 5

[34] Timo Milbich, Karsten Roth, Homanga Bharadhwaj, Samarth Sinha, Yoshua Bengio, Björn Ommer, and Joseph Paul Cohen. Diva: Diverse visual feature aggregation for deep metric learning. *CoRR*, abs/2004.13458, 2020. 2, 4, 5, 6, 7

[35] T. Milbich, K. Roth, B. Brattoli, and B. Ommer. Sharing matters for generalization in deep metric learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2020. 2, 4

[36] Timo Milbich, Karsten Roth, Samarth Sinha, Ludwig Schmidt, Marzyeh Ghassemi, and Björn Ommer. Characterizing generalization under out-of-distribution shifts in deep metric learning. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021. 2

[37] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 6706–6716. IEEE, 2020. 2

[38] 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. 1, 2, 3, 5, 6

[39] Kevin Musgrave, Serge J. Belongie, and Ser-Nam Lim. A metric learning reality check. *CoRR*, abs/2003.08505, 2020. 2, 5

[40] 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. 2, 5, 6

[41] George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density estimation. 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. 2, 4

[42] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017. 5

[43] 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. 1, 2, 3, 5, 6

[44] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, page 1530–1538. JMLR.org, 2015. 4

[45] Robin Rombach, Patrick Esser, and Bjorn Ommer. Network-to-network translation with conditional invertible neural networks. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 2784–2797. Curran Associates, Inc., 2020. 4

[46] 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. 2, 4, 6

[47] 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. 1, 2, 6

[48] Karsten Roth, Timo Milbich, Bjorn Ommer, Joseph Paul Cohen, and Marzyeh Ghassemi. Simultaneous similarity-based self-distillation for deep metric learning. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 9095–9106. PMLR, 18–24 Jul 2021. 2

[49] 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. 2, 4, 5, 6, 7

[50] Marco Rudolph, Bastian Wandt, and Bodo Rosenhahn. Same same but differnet: Semi-supervised defect detection with normalizing flows. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1907–1916, January 2021. 4

[51] 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. 2, 6

[52] 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. 1, 2

[53] Jenny Denise Seidenschwarz, Ismail Elezi, and Laura Leal-Taixé. Learning intra-batch connections for deep metric learning. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 9410–9421. PMLR, 18–24 Jul 2021. 2

[54] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *Advances in Neural Information Processing Systems*, pages 1857–1865, 2016. 2

[55] Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. Circle loss: A unified perspective of pair similarity optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 6

[56] 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 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XXIV*, volume 12369 of *Lecture Notes in Computer Science*, pages 448–464. Springer, 2020. 1, 2, 3, 5

[57] 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. 2, 5, 6

[58] Feng Wang, Xiang Xiang, Jian Cheng, and Alan Loddon Yuille. Normface: L¡sub¿2¡/sub¿ hypersphere embedding for face verification. In *Proceedings of the 25th ACM International Conference on Multimedia*, MM '17, page 1041–1049, New York, NY, USA, 2017. Association for Computing Machinery. 2

[59] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition, 2018. 2

[60] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. *arXiv preprint arXiv:2005.10242*, 2020. 2, 7

[61] 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. 2

[62] Ross Wightman. Pytorch image models. `https://github.com/rwightman/pytorch-image-models`, 2019. 5

[63] 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. 1, 2

[64] 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. 6

[65] Dingyi Zhang, Yingming Li, and Zhongfei Zhang. Deep metric learning with spherical embedding. In *NeurIPS*, 2020. 2

[66] 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. 2, 8

[67] Wenzhao Zheng, Chengkun Wang, Jiwen Lu, and Jie Zhou. Deep compositional metric learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9320–9329, June 2021. 6

[68] Wenzhao Zheng, Borui Zhang, Jiwen Lu, and Jie Zhou. Deep relational metric learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12065–12074, October 2021. 6

[69] Yuehua Zhu, Muli Yang, Cheng Deng, and Wei Liu. Fewer is more: A deep graph metric learning perspective using fewer proxies. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 17792–17803. Curran Associates, Inc., 2020. 2, 6

[70] Roland S. Zimmermann, Yash Sharma, Steffen Schneider, Matthias Bethge, and Wieland Brendel. Contrastive learning inverts the data generating process. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 12979–12990. PMLR, 18–24 Jul 2021. 3