Deep Adversarial Metric Learning

Yueqi Duan, Jiwen Lu, *Senior Member, IEEE*, Wenzhao Zheng, and Jie Zhou, *Senior Member, IEEE*

***Abstract*— Learning an effective distance measurement between sample pairs plays an important role in visual analysis, where the training procedure largely relies on hard negative samples. However, hard negative samples usually account for the tiny minority in the training set, which may fail to fully describe the data distribution close to the decision boundary. In this paper, we present a deep adversarial metric learning (DAML) framework to generate synthetic hard negatives from the original negative samples, which is widely applicable to existing supervised deep metric learning algorithms. Different from existing sampling strategies which simply ignore numerous easy negatives, our DAML aim to exploit them by generating synthetic hard negatives adversarial to the learned metric as complements. We simultaneously train the feature embedding and hard negative generator in an adversarial manner, so that adequate and targeted synthetic hard negatives are created to learn more precise distance metrics. As a single transformation may not be powerful enough to describe the global input space under the attack of the hard negative generator, we further propose a deep adversarial multi-metric learning (DAMML) method by learning multiple local transformations for more complete description. We simultaneously exploit the collaborative and competitive relationships among multiple metrics, where the metrics display unity against the generator for effective distance measurement as well as compete for more training data through a metric discriminator to avoid overlapping. Extensive experimental results on five benchmark datasets show that our DAML and DAMML effectively boost the performance of existing deep metric learning approaches through adversarial learning.**

***Index Terms*— Metric learning, deep learning, adversarial learning, hard negative generation, multi-metric.**

I. Introduction

METRIC learning aims to learn a similarity measurement, which makes the following clustering and classification tasks much simpler. Metric learning methods have been widely used in numerous visual analysis tasks, such as face recognition [19], [38], image classification [6], [7], [68], person re-identification [37], [72], [74], and visual tracking [22], [66]. Existing methods can be mainly divided into two categories: linear and nonlinear [30]. Conventional linear metric learning approaches learn a Mahalanobis distance metric [7], [19], [68], while nonlinear approaches usually apply kernel tricks or deep neural networks to model high-order relationships [6], [27], [41], [53]–[55], [63].

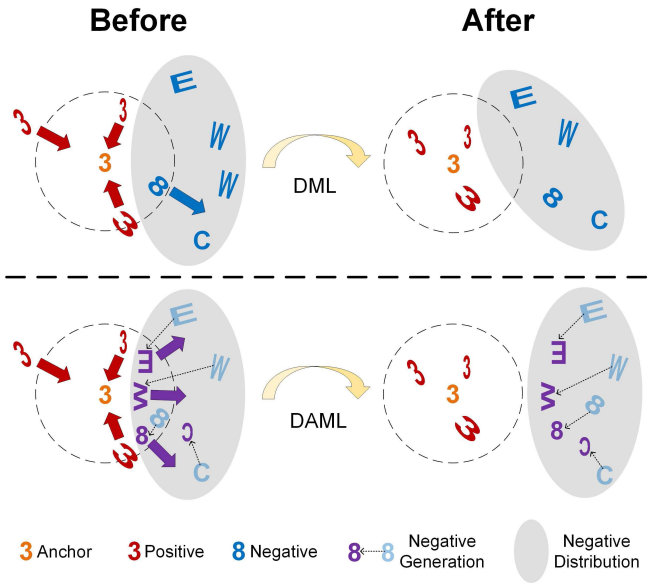


Fig. 1. Comparisons of conventional deep metric learning (DML) methods and the proposed DAML. In this figure, we utilize the number of "3" as the anchor and positives while other numbers and alphabets are negatives for intuitive demonstration. We compute the distances between samples according to the similarity of shapes. Existing DML methods rely on the hard negatives in the training set, pushing the distribution of negatives to the lower right. However, they fail to handle the potential hard negatives at the upper right of the decision boundary. For DAML, we aim to generate synthetic hard negatives from existing negatives adversarial to the metric, which exploits numerous easy negatives as complements.

For most supervised deep metric learning (DML) approaches, the training objective is to maximize the inter-class variations as well as minimize the intra-class variations [21], [38]. Therefore, the hard negatives in the training set will produce large gradients while others are close to zero. Here, hard negatives are negative samples that are close to anchors, while easy negatives are those far away from anchors. As hard negatives usually account for the tiny minority in the training set, the vast majority of easy negatives samples make little contribution to metric learning. A natural question is raised: are easy negatives really useless?

In this work, we consider that easy negatives should not be ignored as they may have potential to generate important complements. For example, the shape of the letter "W" is different from the number "3". However, after a rotation of 90 degrees counter-clockwise, it would become a dangerous hard negative. Fig. 1 illustrates the reason of this phenomenon. In the training set, hard negatives usually account for the

Manuscript received March 13, 2019; revised July 31, 2019 and September 12, 2019; accepted October 16, 2019. Date of publication October 25, 2019; date of current version December 30, 2019. This work was supported in part by the National Key Research and Development Program of China under Grant 2016YFB1001004 and in part by the National Natural Science Foundation of China under Grant 61822603, Grant U1813218, Grant U1713214, Grant 61672306, and Grant 61572271. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Yun Fu. *(Corresponding author: Jiwen Lu.)*

The authors are with the Beijing National Research Center for Information Science and Technology (BNRist), the State Key Lab of Intelligent Technologies and Systems, and the Department of Automation, Tsinghua University, Beijing, 100084, China (e-mail: duanyq14@mails.tsinghua.edu.cn; lujiwen@tsinghua.edu.cn; zhengwz18@mails.tsinghua.edu.cn; jzhou@tsinghua.edu.cn).

This article has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors.

Digital Object Identifier 10.1109/TIP.2019.2948472

tiny minority, which may not fully describe the distribution of negatives close to the decision boundary. Existing DML methods simply maximize the relative distance of the observed hard negative space (which is spanned by the samples in the training set), while the unobserved hard negatives are still in danger.

In this paper, we present a deep adversarial metric learning (DAML) framework to address the limitation, which can be generally applied to existing supervised DML methods. Instead of simply using the original negatives in the training set, our goal is to generate synthetic hard negatives in an adversarial manner, so that easy negatives can also be exploited to provide important complements. We simultaneously train the feature embedding and hard negative generator to obtain adequate and targeted synthetic hard negatives. Adequate hard negatives illustrate a complete description of the sample distribution close to the decision boundary, while targeted hard negatives expose the limitations of the current feature embedding. Fig. 1 shows the comparisons between existing DML methods and DAML.

While DAML learns effective distance measurement through the generated hard negative samples, only one global metric may not be discriminative enough to describe the relationships between samples especially under the attack of the generator. Once the learned metric fails to have the ability to correctly classify the synthetic negative samples through training, the effectiveness of the hard negative generator is weakened due to the unbalanced fight. To this end, we further propose a deep adversarial multi-metric learning (DAMML) method by learning multiple local metrics for more precise description through a metric generator. We simultaneously exploit the collaborative and competitive relationships among metrics. On one hand, all the metrics share the same objective against the generator by maximizing the inter-class variations as well as minimizing the intra-class variations. On the other hand, we learn a metric discriminator for input pairs to decide the weights for each local metric. In the training procedure, local metrics are required to compete for more weights of each training pair as the weights are normalized. As a result, each metric gains large weights for part of the training samples so that the local regions described by multiple metrics are separated. In the test procedure, we also utilize the weights from the metric discriminator to compute the final distance between a pair of images in multiple local metrics. Extensive experimental results on five benchmark datasets illustrate that the proposed DAML and DAMML improve the performance of the existing supervised deep metric learning methods in an adversarial manner.

This paper is an extended version of our conference paper [14], where we make the following new contributions:

1) We further present a new DAMML method based on DAML in the conference version by learning multiple local metrics for more precise distance measurement, which present stronger discriminative power against the hard negative generator.
2) We design a metric discriminator to simultaneously exploit the collaborative and competitive relationships among multiple metrics. With the discriminator, local metrics learn proper weights for each input sample pair for more precise description, and also compete for the weights of training samples to avoid overlapping.
3) We conduct more experiments on recent public benchmark datasets including In-Shop Clothes Retrieval and VehicleID to demonstrate the effectiveness of the proposed methods, and present more experimental analysis for in-depth discussions.

II. Related Work

In this section, we briefly review three related topics: metric learning, hard negative mining, and generative adversarial networks.

A. Metric Learning

Metric learning has witnessed great development over the past decade, which aims to learn effective distance measurement of the input samples. Conventional metric learning methods learn a linear Mahalanobis distances, where a number of methods have been presented [7], [17], [31], [45], [50], [51], [68]. For example, Weinberger and Saul [68] presented a large margin nearest neighbor (LMNN) method by enforcing the anchor to share the same labels with its nearest neighbors by a margin, which is one of the most popular methods in the literature. Davis *et al.* [7] proposed an information-theoretic metric learning (ITML) method to formulate the problem by minimizing a regularizer of LogDet divergence.

As linear metric learning methods may suffer from non-linear correlations of samples, kernel tricks are usually employed [15], [69]. However, it is usually empirical for these methods to choose a kernel function, which limits their discriminative power. With the encouraging performance of deep learning on various tasks, deep metric learning (DML) approaches have been presented to learn non-linear mappings [5], [6], [12], [13], [16], [23], [28], [32], [33], [38], [41]–[43], [54], [55], [59], [63]. For example, Liu *et al.* [38] presented a discriminative deep metric learning (DDML) method with deep neural networks. Song *et al.* [55] proposed a lift structure to better exploit training batches. Ustinova and Lempitsky [59] presented a histogram loss for deep metric learning by estimating the distribution of similarities for sample pairs. Wang *et al.* [63] presented an angular loss by constraining the angle relationships inside the triplets. Isola *et al.* [26] proposed a deep cross-triplet embedding algorithm as well as the corresponding sampling strategy for cross-domain feature representations. Bai *et al.* [1] presented a group-sensitive triplet embedding (GS-TRE) method to better model the intra-class variance for vehicle reidentification.

Besides global metric methods which aim to obtain a single metric for all instances, some multi-metric learning methods also have been presented to exploit locality specific information [11], [42], [71]. For example, Ding and Fu [8] proposed a robust transfer metric learning (RTML) method to transfer the knowledge from the well-labeled domain to the unlabeled target domain. Wang *et al.* [64] designed a towards-young cross-generation model by learning an intermediate domain for kinship verification. Ding *et al.* [9] presented generative semantic dictionary learning (GSDL) method with two-stage

GANs to identify new objects which are not included in training data. Bunte *et al.* [3] proposed a limited rank matrix learning method, which extended the symmetric squared matrices in Generalized Metric Learning Vector Quantization (GMLVQ) [48] to rectangular transformation matrices for low-dimensional representations. Ye *et al.* [71] proposed a unified multi-metric learning (UM^{2}L) framework to exploit multiple semantic linkages. Opitz *et al.* [42] presented an online gradient boosting method to reduce the correlation of the learners.

B. Hard Negative Mining

Hard negative mining is employed to better exploit large-scale negative samples for model training in many visual analysis tasks [10], [21], [49], [52], [54], [65], [73]. Hard negative mining can be considered as a problem of bootstrapping, which gradually chooses negatives that trigger false alarms [52]. For example, Schroff *et al.* [49] trained FaceNet with the selected "semi-hard" negatives, which are hard but the distances are still farther than that of positive-anchor pairs. Shrivastava *et al.* [52] presented an online hard example mining (OHEM) approach to train region-based object detectors. Wu *et al.* [70] illustrated the importance of sample selection in DML and proposed distance weighted sampling strategy with margin based loss. Harwood *et al.* [21] presented a smart mining method to efficiently select training samples for DML. Yuan *et al.* [73] proposed a hard-aware deeply cascaded (HDC) embedding method by mining negatives at multiple hard levels according to the models. Rather than sampling existing negatives for data mining, we focus on the exploitation of easy negatives which may have potential to generate synthetic hard negatives as essential complements.

C. Generative Adversarial Networks

Generative adversarial networks (GANs) have gained much attention due to their impressive results during the past four years [4], [18], [25], [35], [39], [44], [47], [61], [67], [75], where the initial methods have been designed for image generation [4], [18], [25], [75]. For example, Goodfellow *et al.* [18] firstly proposed the framework of GANs to recover the training data distribution for image generation. Chen *et al.* [4] presented InfoGAN to learn disentangled representations with a mutual information objective. Zhu *et al.* [75] proposed CycleGAN by introducing a cycle consistency loss for unpaired image-to-image translation. The key of GANs is the idea of adversarial learning, where the generator aims to generate synthetic images that are undistinguishable from the real ones. Both the generator and the discriminator gradually become more and more powerful in the process of adversarial training. Due to the great success of GANs, adversarial learning methods have been applied to other visual analysis tasks more resently, such as object detection [67], domain adaptation [35], [58], face recognition [57], image inpainting [44], and video analysis [39], [61]. However, to our best knowledge, very few relevant adversarial learning works have focused on the fundamental problem of metric learning, which is of significant importance in image classification and clustering. Different from most existing adversarial learning methods which aim to model the image distributions, the proposed DAML and DAMML tap the potential of the training data in the feature space to enhance the discriminative power of the learned distance metric. We aim to generate synthetic negative samples from existing ones that shorten the distance to the anchor, minimize the difference between synthetic and observed negative samples, and confuse the learned metric. Moreover, we evaluate the effectiveness of DAML and DAMML on more complex benchmark datasets than digit data which many GAN methods have employed.

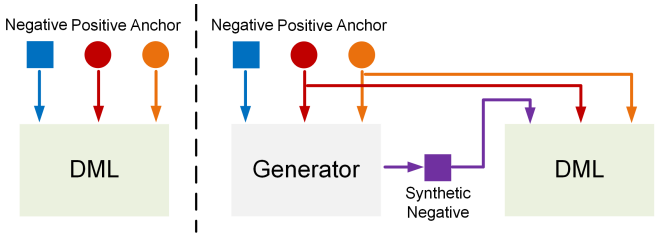


Fig. 2. Frameworks of conventional metric learning with triplet embedding and our proposed DAML. In DAML, we utilize the generated adequate hard negatives to learn the distance metric instead of the observed negatives to fully exploit the potential of each negative sample. We train the generator and the distance metric simultaneously in an adversarial manner, where the training procedure of the generator follows a carefully designed objective function J_{gen}.

III. Deep Adversarial Metric Learning

In this section, we first present the hard negative generator, and then elaborate on the approach of deep adversarial metric learning.

A. Hard Negative Generator

To our best knowledge, existing metric learning methods take advantage of the observed data to learn distance metrics, where the hard negative samples produce gradients with large magnitudes. However, as hard negatives usually account for the tiny minority, there are two main limitations of the existing approaches:

1) The observed hard negatives may not be enough to fully characterize the distributions of negative samples near the decision boundary, as shown in Fig. 1. In some cases, most hard negatives only belong to a few identities, which suffer from limited diversities. The use of inadequate hard negatives may lead to local optimal distance metrics, where potential hard negatives in the unobserved space would probably be misclassified.
2) A large number of easy negative samples are wasted since they produce gradients close to zero. However, some of the easy negatives may have potential to generate synthetic hard negative samples as important complements to the observed hard negatives, which may be misclassified by the learned metric.

In this paper, we generate synthetic hard negatives from observerd easy ones against the trained metric to simultaneously address the above two limitations. Fig. 2 shows the framework of the proposed DAML compared with the conventional deep metric learning methods. Let $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_n]$

be the input data and $\mathbf{Y} = [y_1, \cdots, y_n]$ be the corresponding labels, where $y_i \in \{1, \cdots, C\}$. We employ the widely-used triplet embeddings and contrastive embeddings for explanation. The triplet input $\{\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-\}$ consists of an anchor point $\mathbf{x}_i$, a positive point $\mathbf{x}_i^+$ with its label $y_i^+ = y_i$, and a negative point $\mathbf{x}_i^-$ with its label $y_i^- \neq y_i$, while the pairwise input for contrastive embedding utilizes $\{\mathbf{x}_i, \mathbf{x}_i^+\}$ and $\{\mathbf{x}_j, \mathbf{x}_j^-\}$.

In general, the objective of metric learning is to learn a feature embedding to measure the distance of an input pair:

$$D(\mathbf{x}_i, \mathbf{x}_j) = f(\theta_f; \mathbf{x}_i, \mathbf{x}_j), \tag{1}$$

where D is the distance between the input pair under the trained metric, f is the metric function, and θ_f is the learned parameters of f.

For example, in the conventional linear Mahalanobis distance metric learning, we have

$$f(\theta_f; \mathbf{x}_i, \mathbf{x}_j) = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{M} (\mathbf{x}_i - \mathbf{x}_j)}, \tag{2}$$

where θ_f is the learned matrix $\mathbf{M}$.

Most supervised metric learning methods aim to obtain the parameters θ_f through optimizing a well-designed objective function:

$$\theta_f = \arg\min_{\theta_f} J_{\mathrm{m}}(\theta_f; \mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-, f), \tag{3}$$

where one of the $\mathbf{x}_i^+$ and $\mathbf{x}_i^-$ is set default for the contrastive embedding.

In this paper, we aim to enhance the training procedure using adversarial hard negative generation. We simultaneously train the distance metric and the generator in an adversarial manner by utilizing the synthetic hard negatives as adversary:

$$\theta_f^a = \arg\min_{\theta_f} J_{\mathrm{m}}(\theta_f; \mathbf{x}_i, \mathbf{x}_i^+, \widetilde{\mathbf{x}}_i^-, f), \tag{4}$$

where $\widetilde{\mathbf{x}}_i^-$ is the generated negative sample:

$$\widetilde{\mathbf{x}}_i^- = g(\theta_g; \mathbf{x}_i^-, \mathbf{x}_i, \mathbf{x}_i^+), \tag{5}$$

and θ_g is the parameters of the generator g.

In (5), we aim to generate synthetic negative samples from original ones, so that we can exploit more easy negatives as complements to the observed data. As each negative point would generate different synthetic samples depending on the anchor and positive point, we simultaneously utilize $\mathbf{x}_i^-$, $\mathbf{x}_i$ and $\mathbf{x}_i^+$ as the input of the generator, where we set $\mathbf{x}_i^+ = \mathbf{x}_i$ for the negative pairwise input. Our goal is to train the generator and the distance metric simultaneously, and we formulate the objective function of the generator as follows:

$$\begin{aligned}\min_{\theta_g} J_{\mathrm{gen}} &= J_{\mathrm{hard}} + \lambda_1\ J_{\mathrm{reg}} + \lambda_2\ J_{\mathrm{adv}} \\ &= \sum_{i=1}^{N} (||\widetilde{\mathbf{x}}_i^- - \mathbf{x}_i||_2^2 + \lambda_1 ||\widetilde{\mathbf{x}}_i^- - \mathbf{x}_i^-||_2^2 \\ &\quad + \lambda_2 [D(\widetilde{\mathbf{x}}_i^-, \mathbf{x}_i)^2 - D(\mathbf{x}_i^+, \mathbf{x}_i)^2 - \alpha]_+)\end{aligned} \tag{6}$$

where N is the number of the inputs, α is an enforced positive distance margin between positive-anchor pairs and negative-anchor pairs, the operation of $[\cdot]_+$ refers to the hinge function $\max(0, \cdot)$, and λ_1 and λ_2 are two parameters to balance the contributions of different terms.

The goal of J_{hard} is to make the synthetic negatives close to the anchor, which would produce large gradient magnitudes for the training procedure of metric learning. J_{reg} is a self-regularization term to minimize the difference between the generated negatives and the original ones. J_{adv} aims to generate the negative samples on which the learned metric would misclassify, encouraging the difference between the distances of negative-anchor pairs and the corresponding positive-anchor pairs smaller than a margin α. The procedure of adversarial training enhances the discriminative power of the learned metrics to deal with potential unobserved hard negatives.

B. DAML

The framework of adversarial metric learning can be generally applied to various loss functions of supervised metric learning, where we simultaneously train the hard negative generator and the distance metric using the following objective function:

$$\min_{\theta_g, \theta_f} J = J_{\mathrm{gen}} + \lambda J_{\mathrm{m}}, \tag{7}$$

where λ is the parameter to balance the contributions of different terms, and we develop various embeddings of J_{m} to demonstrate the effectiveness of the proposed adversarial metric learning.

DAML (cont): For contrastive embeddings, we employ [20], [38] to define the objective function as:

$$J_{\mathrm{m}} = \sum_{i=1}^{N_i} D(\mathbf{x}_i^+, \mathbf{x}_i)^2 + \sum_{j=1}^{N_j} [\alpha - D(\widetilde{\mathbf{x}}_j^-, \mathbf{x}_j)^2]_+, \tag{8}$$

where N_i and N_j represent the numbers of positive and negative pairs, respectively.

DAML (tri): For triplet embeddings, we employ [49], [68] to define the objective function, which is widely used for the triplet input:

$$J_{\mathrm{m}} = \sum_{i=1}^{N} [D(\mathbf{x}_i^+, \mathbf{x}_i)^2 - D(\widetilde{\mathbf{x}}_i^-, \mathbf{x}_i)^2 + \alpha]_+, \tag{9}$$

where the objective enforces the distances of negative-anchor pairs to be larger than the corresponding positive-anchor pairs by a margin.

DAML (lifted): We also employ [55] for the lifted structure to define the objective function as follows:

$$J_{\mathrm{m}} = \frac{1}{2N_i} \sum_{i=1}^{N_i} \max(0, J_{i^+,i}), \tag{10}$$

$$\begin{aligned}J_{i^+,i} &= \max(\max \alpha - \widetilde{D}(\mathbf{x}_i^+), \max \alpha - \widetilde{D}(\mathbf{x}_i)) \\ &\quad + D(\mathbf{x}_i^+, \mathbf{x}_i),\end{aligned} \tag{11}$$

where $\widetilde{D}(\mathbf{x})$ represents the distances of the negative pairs for $\mathbf{x}$. We suggest referring [55] for more details.

DAML (N-pair): In the N-pair loss [53], the anchor from each class $\mathbf{x}_c$ would have one positive sample $\mathbf{x}_c^+$ and $C - 1$

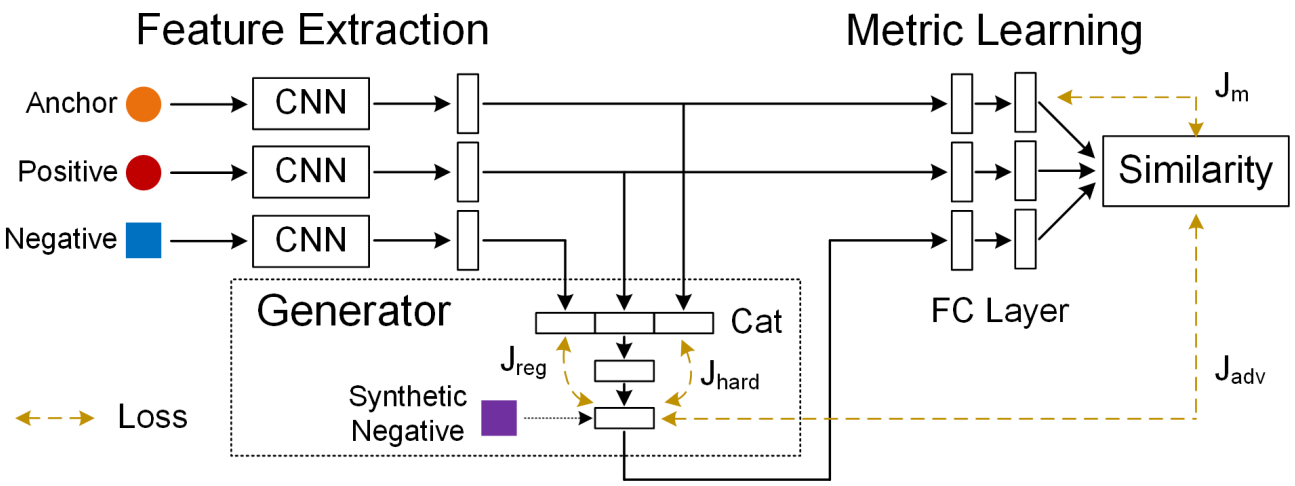


Fig. 3. The overall network architecture of the proposed DAML for the triplet input. We simultaneously train the distance metric and the hard negative generator, where the CNNs and fully connected layers share the same architectures and parameters. The generator takes as input the features extracted from CNNs, and then generates synthetic hard negatives for deep metric learning.

Algorithm 1 DAML

Input: Training image set, parameters λ, λ_1 and λ_2, margin α, and iteration numbers T.
Output: Parameters of the hard negative generator θ_g, and parameters of the metric function θ_f.
1: Pre-train θ_f without the hard negative generator.
2: Pre-train θ_g.
3: **for** $iter = 1, 2, \cdots, T$ **do**
4: Sample minibatch of m training images.
5: Produce triplet or pairwise inputs from the batch.
6: Jointly optimize θ_g and θ_f using (7).
7: **end for**
8: **return** θ_g and θ_f.

negative samples $\mathbf{x}_{c'}^{+}$, where C is the number of classes and $c' \neq c$. For each $\mathbf{x}_c$ and $\mathbf{x}_c^{+}$, we generate $C-1$ synthetic hard negatives $\widetilde{\mathbf{x}}_{c'}^{+}$ from $\mathbf{x}_{c'}^{+}$ using the generator. The metric objective term of DAML (N-pair) is defined as follows:

$$J_{\mathrm{m}} = \frac{1}{C}\sum_{c=1}^{C}\log\Big(1+\sum_{c'\neq c}\exp\big(D(\mathbf{x}_c,\widetilde{\mathbf{x}}_{c'}^{+}) - D(\mathbf{x}_c,\mathbf{x}_c^{+})\big)\Big) \quad (12)$$

where $D(\mathbf{x}_i,\mathbf{x}_j) = \mathbf{f}_i^T\mathbf{f}_j$ is the similarity measure used in the N-pair loss, and $\mathbf{f}_i$ and $\mathbf{f}_j$ are the embedded features. See [53] for complete details.

We simultaneously train the hard negative generator and the distance metric in a joint manner, and Fig. 3 shows the overall network architecture. In the training procedure, we first pre-train the deep metric learning model without the hard negative generator. Then, we initialize the generator adversarially to the pre-trained metric. Lastly, we jointly optimize both networks during each iteration end-to-end, where the synthetic hard negatives are used to train the distance metric. In the test procedure, as the CNNs and fully connected layers have the same structures and parameters, we apply the metric network for similarity measurement without the generator. Algorithm 1 details the approach of DAML.

IV. Deep Adversarial Multi-Metric Learning

In this section, we first detail the approach of deep adversarial multi-metric learning, and then we introduce the implementation details of the proposed DAML and DAMML.

A. DAMML

While DAML successfully taps the potentials of the easy negative samples, it only utilize one global metric to describe the holistic input space, which may not be discriminative enough especially under the attack of the hard negative generator. In order to address the limitation, we further propose a deep adversarial multi-metric learning (DAMML) method to learn a more precise distance measurement though multiple local metrics. There are two key challenges in multi-metric learning [42], [71]: 1) the integration of multiple metrics to obtain the final distance of a pair of samples, and 2) the independence of multiple metrics to avoid overlapping. In this paper, we simultaneously address the problems by learning a metric discriminator, which decides the weights for each sample pair on different local metrics:

1) We normalize the feature embedding of each local metric and learn adaptive weights for different input pairs. Compared with the hand-crafted methods which use the same integration strategy for all the samples, we consider that the integration method for varying inputs should be data-adaptive. To better measure the distance between sample pairs in the embedded space, the weights of multiple metrics should be different according to the local information of the specific inputs.
2) Objective function and training samples determine the learned deep metric. If we optimize the network with the same loss function and training data, the learned local metrics will be highly correlated and present no improvements. To address the problem, we still employ the same loss as a cooperative objective for all the local metrics, but each sample has different weights for training varying local metrics. To this end, the correlation

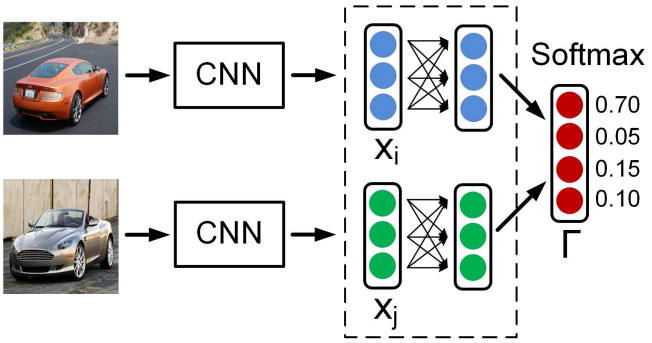


Fig. 4. The network architecture of the metric discriminator. In this figure, we set $K = 4$ for easy illustration. Given a pair of samples $\{\mathbf{x}_i, \mathbf{x}_j\}$, we obtain the weights for local metrics. We share the parameters for the two flows in the dashed box, so that the metric discriminator is invariant to the sequence of the sample pair.

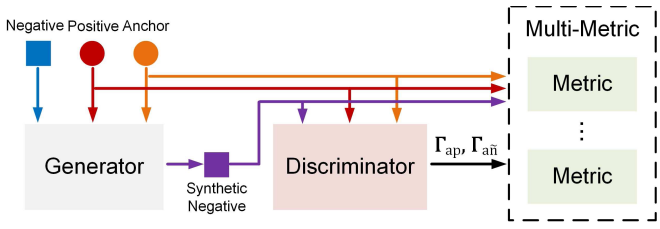


Fig. 5. The framework of the proposed DAMML approach. In this figure, we utilize $\mathbf{\Gamma}_{ap}$ and $\mathbf{\Gamma}_{a\tilde{n}}$ to represent $\mathbf{\Gamma}(\mathbf{x}_a, \mathbf{x}_p)$ and $\mathbf{\Gamma}(\mathbf{x}_a, \mathbf{x}_{\tilde{n}})$ for short, and we input two of the samples into the metric discriminator at each time. Though the metric discriminator, we obtain the weights of local metrics when computing the final distance of each sample pair.

is reduced as the training weights are different for each input sample on each local metric.

Let h be the function of the metric discriminator, with the input as a pair of samples and the output as the weights of local metrics:

$$\mathbf{\Gamma}(\mathbf{x}_i, \mathbf{x}_j) = h(\theta_h; \mathbf{x}_i, \mathbf{x}_j), \tag{13}$$

where θ_h is the learned parameters of h, and $\mathbf{\Gamma}(\mathbf{x}_i, \mathbf{x}_j) = [\gamma_{ij}^1, \cdots, \gamma_{ij}^K]^T$ is the weights of K local metrics for the input pair $\{\mathbf{x}_i, \mathbf{x}_j\}$, subject to $\gamma_{ij}^k \geq 0$ and $\sum_{k=1}^K \gamma_{ij}^k = 1$. We employ a 2-layer fully connected neural network for h, as shown in Fig. 4. As the weights should be invariant to the sequence of the input, i.e. $\mathbf{\Gamma}(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{\Gamma}(\mathbf{x}_j, \mathbf{x}_i)$, we share the parameters of the two flows of the network.

With the definition of the metric discriminator, we rewrite the distance between a pair of samples as a weighted sum:

$$D(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k=1}^{K} \gamma_{ij}^k f_k(\theta_f^k; \mathbf{x}_i, \mathbf{x}_j), \tag{14}$$

where f_k and θ_f^k are the kth local metric and its parameters, respectively.

In (14), local metrics have varying weights to compute the distance between a pair of samples. Through the metric discriminator, more relevant local metrics gain larger weights for distance measurement while the others make little contribution. If we set $K = 1$ in (14), γ_{ij}^k is equal to 1 and it will degenerate to the single metric version. Fig. 5 shows the framework of DAMML.

For DAMML, we simultaneously train the hard negative generator, the metric discriminator, and the multiple local metrics by optimizing the following objective function:

$$\min_{\theta_g, \theta_h, \theta_f^k} J = J_{\text{gen}} + \mu J_{\text{dis}} + \lambda J_{\text{m}}. \tag{15}$$

In (15), we directly employ the same J_{gen} and J_{m} as DAML, modifying the distance measurement to (14). As $\mathbf{\Gamma}$ simultaneously appears in J_{gen} and J_{m}, the metric discriminator is trained for better measurement of the final distances, and the hard negative generator works against both metric discriminator and multiple local metrics.

For J_{dis}, we encourage each sample pair to be mainly described by only one local metric rather than multiple local metrics to avoid overlapping, and we formulate the objective function to maximize the variance of the weights as follows:

$$\begin{aligned} J_{\text{dis}} &= -\sum_{\mathbf{x}_i, \mathbf{x}_j} \text{Var}(\mathbf{\Gamma}(\mathbf{x}_i, \mathbf{x}_j)) \\ &= -\sum_{\mathbf{x}_i, \mathbf{x}_j} (\mathbf{\Gamma}(\mathbf{x}_i, \mathbf{x}_j) - \frac{1}{K})^T (\mathbf{\Gamma}(\mathbf{x}_i, \mathbf{x}_j) - \frac{1}{K}), \\ \text{subject to } \gamma_{ij}^k &\geq 0, \ \sum_{k=1}^{K} \gamma_{ij}^k = 1, \end{aligned} \tag{16}$$

where J_{dis} achieves the minimum when one of the weights is equal to 1 and the others are zero.

In the training procedure, we simultaneously train the hard negative generator, the metric discriminator and the multiple local metrics in an end-to-end manner. As all the local metrics still share the same objective function of J_{m}, they cooperate to learn discriminative distance measurement by maximizing the inter-class variations and minimizing the intra-class variations. Meanwhile, for a pair of training samples, each local metric would receive a weight from the metric discriminator. The weights determine the proportions for local metrics in final distance computation, so that the metrics with large weights produce large gradients in training, and their correlations are reduced through the competition in weights. In the test procedure, we still employ the weighted sum as the distance measurement through the metric discriminator, which is a data-dependent integration method of the multiple local metrics. Fig. 6 shows the overall network architecture of DAMML and Algorithm 2 details the approach.

B. Implementation Details

We utilized the TensorFlow package through the experiments. We normalized the images into 256×256 at first, and then we performed standard random crop and horizontal mirroring for data augmentation. For the generator network, the dimension is $3072 \rightarrow 1024 \rightarrow 1024$, and the dimension of the discriminator is $2048 \rightarrow 1024 \rightarrow K$. For the metric network, we performed the initialization with GoogLeNet [56] which was pretrained on the ImageNet ILSVRC dataset [46], and randomly initialized an added fully connected layer. We optimized the new layer with 10 times learning rate compared with other layers for fast convergence. We used a 3-layer fully connected network as the generator by concatenating the

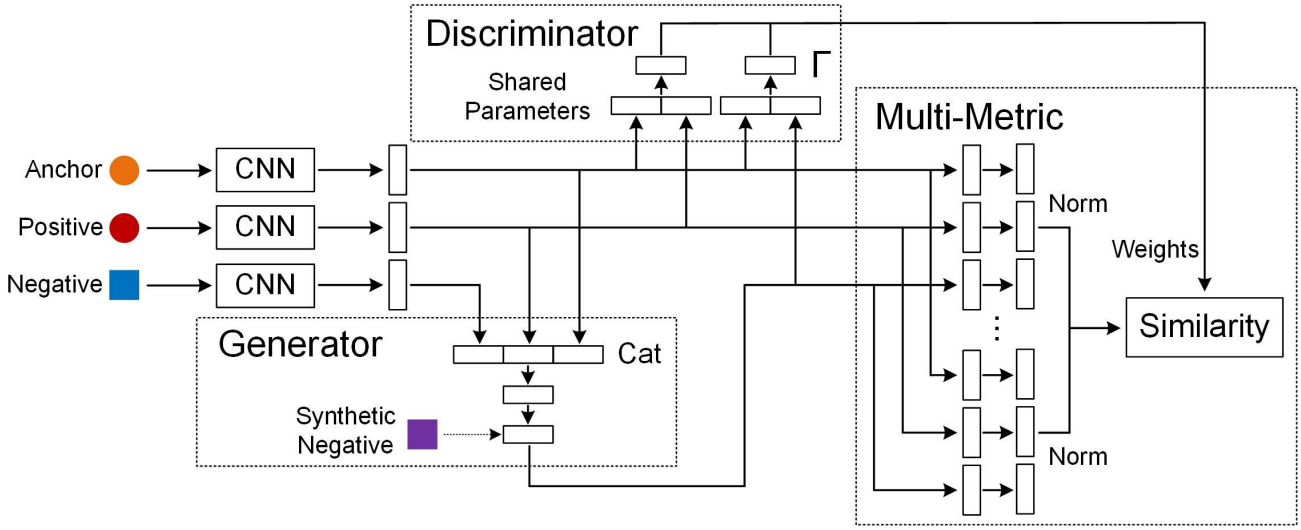


Fig. 6. The overall network architecture of the proposed DAMML for the triplet input. We simultaneously train the hard negative generator, the metric generator, and the multiple local metrics. To compute the distance between a pair of input samples, we employ the metric discriminator to calculate the weights for local metrics, where the two networks of the discriminator share the parameters. Then, we obtain the final distance based on normalized local feature embeddings and the corresponding weights.

Algorithm 2 DAMML

Input: Training image set, parameters λ, λ_1, λ_2, and μ, margin α, and iteration numbers T.
Output: Parameters of the hard negative generator θ_g, the metric discriminator θ_h, and the multiple local metric functions θ_f^k.
1: Initialize $\mathbf{\Gamma} = 1/K$.
2: Pre-train θ_f^k without the hard negative generator.
3: Pre-train θ_g and θ_h.
4: **for** $iter = 1, 2, \cdots, T$ **do**
5: Sample minibatch of m training images.
6: Produce triplet or pairwise inputs from the batch.
7: Jointly optimize θ_g, θ_h and θ_f^k using (15).
8: **end for**
9: **return** θ_g, θ_h and θ_f^k.

features as the input and generating the synthetic negative as the output. We empirically fixed the parameters λ, λ_1, λ_2 and μ as 1, 1, 50 and 10^5 to balance the weights of different terms based on the parameter analysis in Fig. 7, respectively, and we followed [68] by setting α to 1. For DAML, as an experimental study in [55] shows that the embedding size does not largely affect the performance, we followed [63] to fix the embedding size to 512 throughout the experiment. We also conducted an experiment to show the influence of different embedding sizes. For DAMML, we fixed the sum of multiple embedding sizes to 512 for fair comparisons, where the embedding size for each local metric was $\frac{512}{K}$. We fixed the maximum training iteration to 20,000 and set the batchsize as 128 for the pairwise input and 120 for the triplet input.

V. Discussion

In this section, we compare the proposed DAML and DAMML with relevant methods to highlight the differences.

A. Difference With Existing Hard Negative Mining Methods

Hard negative mining has been widely applied in many visual analysis tasks and has successfully boosted the performance of metric learning [21], [73]. The core idea of hard negative mining is to gradually select dangerous negative samples which are misclassified by the current machines. In this paper, we argue that some easy negatives that are not selected by the miner in the original form may have potential to become very dangerous. For example, the letter "W" may not be selected by the hard negative miner for the number "3". However, it is able to create a really dangerous synthetic negative by rotating it by 90 degrees counterclockwise, which may be even harder than all the observed negatives. In general, hard negative mining selects useful existing observed samples, while DAML taps their potential. Moreover, we emphasize that DAML does not conflict with hard negative mining, where we can generate more negative samples at first for the following full selections.

B. Difference With Existing Data Augmentation Methods

The goal of data augmentation is to apply transformation to the images without altering the labels, which have been widely applied to improve the performance of CNN and prevent from overfitting [40]. The key difference between DAML and data augmentation is that we simultaneously learn the generator and feature embedding in an adversarial manner to obtain metric-specific synthetic hard negatives instead of applying fixed transformation to all the images. The generated samples especially target at the limitations of the current feature embedding for effective direction, while data augmentation methods utilize the same transformed samples despite of the current state of the learned metric. Moreover, different from most existing data augmentation methods which employ simple geometric transformations such as mirroring, rotating and oversampling, we generate synthetic samples in the feature space with stronger flexibility.

C. Difference With Existing Multi-Metric Learning Methods

There have been many studies on multi-metric learning which learn multiple metrics for more precise description, such

as [2], [24], [42], [71]. However, most of these multi-metric learning methods are trained on the original data, where the challenge of how to simultaneously train the hard negative generator and the multi-metric still remains. As the training objectives of generator and multi-metric are opposite, it is hard for the generator to fool all the metrics at the same time, and DAMML is the first attempt to address this problem in multi-metric learning. While we cannot directly apply existing multi-metric methods, we design an additional discriminator to learn the weight of each metric. To this end, the discriminator decides better local metrics for complete description of each sample, the generator creates more powerful hard negatives that especially fool the metrics with higher weights, and the multiple local metrics learn better cooperation against the generator with J_{dis}.

D. Hard Positive Generation

As DAML and DAMML aim to generate synthetic hard negative samples from easier ones for full exploitation, an intuitive similar idea is to create hard positive samples. It is reasonable because hard positives also play more important roles than easy positives by producing larger gradients. The reason we choose to only generate hard negatives is that there are much more easy negative samples than easy positive samples that are ignored by the existing approaches, which makes hard negative generation much more important. Moreover, negative samples have larger variations compared with positive samples to generate more effective synthetic samples. In the experiments, we also design a similar algorithm to test the performance of hard positive generation.

VI. Experiments

In this section, we conducted experiments on five widely-used benchmark datasets for both retrieval and clustering tasks to demonstrate the effectiveness of the proposed DAML and DAMML, which included the CUB-200-2011 [62], Cars196 [29], Stanford Online Products [55], In-Shop Clothes Retrieval [36] and VehicleID [34] datasets.

For the clustering task, we followed [55] to perform K-means algorithm in the test set, and then use the normalized mutual information (NMI) and F_1 metrics. The input of NMI is a set of clusters $\Omega = \{\omega_1, \cdots, \omega_K\}$ and the ground truth classes $\mathbb{C} = \{c_1, \cdots, c_K\}$, where ω_i represents the samples belonging to the ith cluster, and c_j is the set of samples with the label of j. NMI is defined as the ratio of mutual information and the mean entropy of clusters and the ground truth:

$$\text{NMI}(\Omega, \mathbb{C}) = \frac{2I(\Omega; \mathbb{C})}{H(\Omega) + H(\mathbb{C})}, \tag{17}$$

and F_1 metric is the harmonic mean of precision and recall:

$$\text{F}_1 = \frac{2PR}{P + R}. \tag{18}$$

For the retrieval task, we computed the percentage of test samples which have at least one example with the same label in R nearest neighbors.

TABLE I
Overview of the Compared Baseline Methods

Method	Embedding	Supervision
DDML [23]	Contrastive	Weak
Tri+N-pair [63]	Batch	Strong
Angular [63]	Triplet	Strong
Contrastive [20]	Contrastive	Weak
Triplet [68]	Triplet	Strong
Lifted [55]	Batch	Strong
N-pair [53]	Batch	Strong

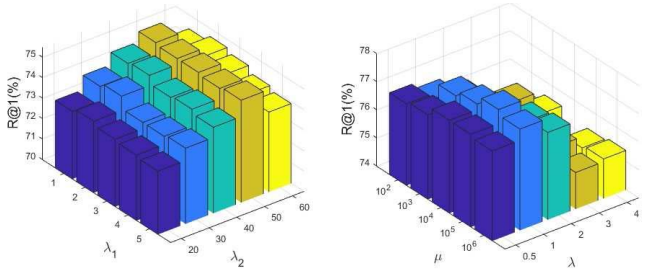
A. Datasets

We conducted experiments on five widely-used benchmark datasets to evaluate DAML and DAMML with the standard evaluation protocol [29], [34], [36], [55], [62] to demonstrate the effectiveness of the proposed methods:

1) The CUB-200-2011 dataset [62] includes 11,788 images of 200 bird species. We used the first 100 species with 5,864 images for training, and the rest 100 species with 5,924 images for testing.
2) The Cars196 dataset [29] contains 16,185 images of 196 car models. We used the first 98 models with 8,054 images for training, and the remaining 98 models with 8,131 images for testing.
3) The Stanford Online Products dataset [55] consists of 120,053 images of 22,634 products from eBay.com. We used the first 11,318 products with 59,551 images for training, and the other 11,316 products with 60,502 images for testing.
4) The In-Shop Clothes Retrieval dataset [36] has 54,642 images of 11,735 classes of clothes. We used the predefined 3,997 classes with 25,882 images for training, 3,985 classes with 14,218 images as the query set, and the other 3,985 classes with 12,612 images as the gallery set.
5) The VehicleID dataset [34] contains 221,763 images of 26,267 vehicles. We used the predefined 13,134 vehicles with 110,178 images for training, and the rest Small, Medium and Large subsets for testing.

B. Baseline Methods

We applied the framework of adversarial metric learning on four baseline methods as mentioned above for direct comparisons, which include the wildly-used contrastive embedding [20], triplet embedding [68] and the more recent lifted structure [55] and N-pair loss [53]. We also compared DAML and DAMML with other three baseline methods for evaluation including DDML [38], triplet loss with N-pair sampling [63] and angular loss [63]. For all the baseline methods and the proposed DAML and DAMML, we employed the same GoogLeNet architecture pre-trained on ImageNet for fair comparisons, while fixing the embedding size as 512. Table I shows an overview of the compared baseline methods. More recently, there are metric learning methods selecting multiple input pairs or triplets from one batch, where we summarize the



(a) λ_1 and λ_2 (b) μ and λ

Fig. 7. The R@1 performance (%) on the CUB-200-2011 dataset of (a) DAML (N-pair) under varying λ_1 and λ_2, and (b) DAMML (N-pair) under different μ and λ with $K = 4$.

TABLE II

THE R@1 PERFORMANCE (%) ON THE CARS196 DATASET OF DAML (N-PAIR) UNDER VARYING EMBEDDING SIZES

Embedding size	64	128	256	512	1024
DAML (N-pair)	73.4	73.2	74.4	**75.1**	74.8

type of embedding as "batch" in Table I. We observe that the compared baseline methods cover the widely-used contrastive, triplet and batch embedding methods, and they also contain both weak and strong supervision signals.

C. Quantitative Results

In this subsection, we show the quantitative experimental results of DAML and DAMML on the baseline datasets.

1) Parameter Analysis: We first tested the retrieval results of R@1 for DAML (N-pair) on the Cars196 dataset, under varying parameters λ_1 and λ_2 of J_{gen} fixing λ as 1. Fig. 7 (a) shows that the values of λ_1 and λ_2 should be set as 1 and 50, respectively. We observe that λ_2 largely influences the performance of DAML, where the goal of J_{adv} is to generate metric-specific hard negatives by targeting at the weakness of the current metric. Without J_{adv}, the hard negative generator acts more like a machine of data augmentation, which provide new data at first despite of the learned metric. However, data augmentation fails to generate targeted hard negative samples, which may not be able to target at the weakness of the current distance metric. Moreover, a too large λ_2 may also lead to uncontrolled hard negative generation. Then, we tested μ and λ for DAMML (N-pair), setting $K = 4$. Fig. 7 (b) shows that the best values are 10^5 and 1. We observe that the performance drops heavily when λ becomes larger. As λ is the weight for the metric loss J_{m}, large λ will weaken the effect of the hard negative generator. The experimental results demonstrate the effectiveness of the hard negative generation.

We also tested the influence of embedding size of DAML (N-pair) and the number of multiple metrics for DAMML (N-pair) on the Cars196 dataset. Table II shows the experimental performance under different embedding sizes. We have a similar observation with [55] that the embedding size does not largely affect the experimental performance, and we set the embedding size as 512 for the rest of the experiments for DAML. For the experiments of DAMML, we fix the total embedding size (the sum of the local embedding sizes) as 512 for fair comparisons, so that the embedding size of each local metric is smaller with a larger K. Table III shows that the performance is improved with the increase of K at the beginning, and then drops when K is too large. The reason is that the multiple metrics better describe the locality information with larger K at the beginning, yet the embedding size of each metric is too small for effective description when K is over large. In the following experiments, we fix $K = 4$ according to the results of Table III.

TABLE III

THE R@1 PERFORMANCE (%) ON THE CARS196 DATASET OF DAMML (N-PAIR) UNDER VARYING NUMBERS OF METRICS

Number of metrics	1	2	4	8	16
DAMML (N-pair)	75.1	75.8	**77.8**	77.6	76.3

TABLE IV

EXPERIMENTAL RESULTS (%) ON THE CUB-200-2011 DATASET COMPARED WITH BASELINE METHODS

Method	NMI	F_1	R@1	R@2	R@4	R@8
DDML [23]	47.3	13.1	31.2	41.6	54.7	67.1
Tri+N-pair [63]	54.1	20.0	42.8	54.9	66.2	77.6
Angular [63]	61.0	30.2	53.6	65.0	75.3	83.7
Contrastive [20]	47.2	12.5	27.2	36.3	49.8	62.1
DAML (cont)	**49.1**	**16.2**	**35.7**	**48.4**	**60.8**	**73.6**
DAMML (cont)	**50.8**	**17.1**	**36.9**	**49.4**	**61.4**	**74.1**
Triplet [68]	49.8	15.0	35.9	47.7	59.1	70.0
DAML (tri)	**51.3**	**17.6**	**37.6**	**49.3**	**61.3**	**74.4**
DAMML (tri)	**53.0**	**18.9**	**39.9**	**51.6**	**63.5**	**74.9**
Lifted [55]	56.4	22.6	46.9	59.8	71.2	81.5
DAML (lifted)	**59.5**	**26.6**	**49.0**	**62.2**	**73.7**	**83.3**
DAMML (lifted)	**60.3**	**28.8**	**50.4**	**63.6**	**74.9**	**84.2**
N-pair [53]	60.2	28.2	51.9	64.3	74.9	83.2
DAML (N-pair)	**61.3**	**29.5**	**52.7**	**65.4**	**75.5**	**84.3**
DAMML (N-pair)	**62.4**	**30.9**	**53.9**	**66.7**	**76.4**	**85.4**

2) Comparison With the Baseline Methods: Table IV-VIII show the experimental results of DAML and DAMML compared with baseline methods on the CUB-200-2011, Cars196, Stanford Online Product, In-Shop Clothes Retrieval and VehicleID datasets, respectively. In the tables, bold numbers represent that DAML improves the results of the original metric learning algorithms, and DAMML further boosts the performance of DAML with multiple local metrics. We use the red color to show the best results and numbers in blue color represent the second best performance.

We observe that the proposed DAML improves the performance of original metric learning approaches on all the benchmark datasets. In particular, although the contrastive embedding receives weak supervision where the generator is only applied to the negative pairs instead of all the inputs, DAML still improves the performance on both clustering and retrieval tasks. Combined with the effective Lifted structure and N-pair loss, the proposed DAML (lifted) and DAML

TABLE V

EXPERIMENTAL RESULTS (%) ON THE CARS196 DATASET COMPARED WITH BASELINE METHODS

Method	NMI	F_1	R@1	R@2	R@4	R@8
DDML [23]	41.7	10.9	32.7	43.9	56.5	68.8
Tri+N-pair [63]	54.3	19.6	46.3	59.9	71.4	81.3
Angular [63]	62.4	31.8	71.3	80.7	87.0	91.8
Contrastive [20]	42.3	10.5	27.6	38.3	51.0	63.9
DAML (cont)	**42.6**	**11.4**	**37.2**	**49.6**	**61.8**	**73.3**
DAMML (cont)	**44.1**	**12.6**	**39.1**	**51.1**	**63.2**	**74.3**
Triplet [68]	52.9	17.9	45.1	57.4	69.7	79.2
DAML (tri)	**56.5**	**22.9**	**60.6**	**72.5**	**82.5**	**89.9**
DAMML (tri)	**58.3**	**25.7**	**61.4**	**73.0**	**82.9**	**90.2**
Lifted [55]	57.8	25.1	59.9	70.4	79.6	87.0
DAML (lifted)	**63.1**	**31.9**	**72.5**	**82.1**	**88.5**	**92.9**
DAMML (lifted)	**65.0**	**33.5**	**73.5**	**82.8**	**89.5**	**93.7**
N-pair [53]	62.7	31.8	68.9	78.9	85.8	90.9
DAML (N-pair)	**66.0**	**36.4**	**75.1**	**83.8**	**89.7**	**93.5**
DAMML (N-pair)	**69.2**	**38.7**	**77.8**	**86.1**	**91.8**	**95.2**

TABLE VI

EXPERIMENTAL RESULTS (%) ON THE STANFORD ONLINE PRODUCTS DATASET COMPARED WITH BASELINE METHODS

Method	NMI	F_1	R@1	R@10	R@100
DDML [23]	83.4	10.7	42.1	57.8	73.7
Tri+N-pair [63]	86.4	21.0	58.1	76.0	89.1
Angular [63]	87.8	26.5	67.9	83.2	92.2
Contrastive [20]	82.4	10.1	37.5	53.9	71.0
DAML (cont)	**83.5**	**10.9**	**41.7**	**57.5**	**73.5**
DAMML (cont)	**83.9**	**11.4**	**42.3**	**57.9**	**73.9**
Triplet [68]	86.3	20.2	53.9	72.1	85.7
DAML (tri)	**87.1**	**22.3**	**58.1**	**75.0**	**88.0**
DAMML (tri)	**87.7**	**23.0**	**59.8**	**76.7**	**89.5**
Lifted [55]	87.2	25.3	62.6	80.9	91.2
DAML (lifted)	**89.1**	**31.7**	**66.3**	**82.8**	**92.5**
DAMML (lifted)	**90.2**	**32.8**	**67.0**	**83.4**	**93.2**
N-pair [53]	87.9	27.1	66.4	82.9	92.1
DAML (N-pair)	**89.4**	**32.4**	**68.4**	**83.5**	**92.3**
DAMML (N-pair)	**91.5**	**34.8**	**70.4**	**84.6**	**93.4**

TABLE VII

EXPERIMENTAL RESULTS (%) ON THE IN-SHOP CLOTHES RETRIEVAL DATASET COMPARED WITH BASELINE METHODS

Method	R@1	R@10	R@20	R@30	R@40
DDML [23]	24.4	47.8	55.6	60.4	64.2
Tri+N-pair [63]	57.5	82.2	87.6	89.8	92.8
Angular [63]	80.4	93.9	95.7	96.5	97.1
Contrastive [20]	23.7	47.5	55.9	60.6	63.8
DAML (cont)	**26.0**	**50.1**	**57.9**	**62.5**	**65.6**
DAMML (cont)	**28.0**	**54.2**	**62.3**	**67.2**	**70.4**
Triplet [68]	56.1	82.0	86.8	89.2	90.6
DAML (tri)	**59.1**	**84.5**	**89.1**	**90.9**	**92.3**
DAMML (tri)	**63.1**	**85.6**	**90.0**	**92.0**	**93.2**
Lifted [55]	75.3	93.1	95.5	96.4	97.0
DAML (lifted)	**77.3**	**93.9**	**96.0**	**96.8**	**97.2**
DAMML (lifted)	**79.7**	**94.3**	**96.3**	**97.1**	**97.5**
N-pair [53]	76.4	93.6	94.7	95.6	96.2
DAML (N-pair)	**78.9**	**93.8**	**95.7**	**96.6**	**97.1**
DAMML (N-pair)	**80.8**	**94.6**	**96.4**	**97.2**	**97.7**

TABLE VIII

EXPERIMENTAL RESULTS (%) ON THE VEHICLEID DATASET COMPARED WITH BASELINE METHODS

Method	Small		Medium		Large	
	R@1	R@5	R@1	R@5	R@1	R@5
DDML [23]	35.1	50.8	30.5	46.8	27.9	43.9
Tri+N-pair [63]	58.3	76.3	30.7	46.6	29.1	45.3
Angular [63]	65.4	76.7	60.9	72.7	57.5	69.0
Contrastive [20]	34.5	49.3	30.7	46.5	28.6	45.3
DAML (cont)	**35.2**	**50.1**	**31.4**	**47.1**	**30.3**	**47.5**
DAMML (cont)	**37.1**	**51.6**	**33.3**	**47.9**	**31.2**	**47.9**
Triplet [68]	58.2	75.9	51.5	70.2	46.7	65.4
DAML (tri)	**58.9**	**77.0**	**52.3**	**71.1**	**48.2**	**67.3**
DAMML (tri)	**60.1**	**77.9**	**52.8**	**71.7**	**48.9**	**68.2**
Lifted [55]	63.2	77.0	59.3	74.7	55.6	71.9
DAML (lifted)	**63.8**	**77.7**	**60.2**	**75.9**	**56.0**	**72.7**
DAMML (lifted)	**64.7**	**78.5**	**60.9**	**77.1**	**56.8**	**73.4**
N-pair [53]	70.2	84.5	64.8	80.6	61.8	78.2
DAML (N-pair)	**71.4**	**85.1**	**65.9**	**81.4**	**63.2**	**80.0**
DAMML (N-pair)	**72.5**	**86.1**	**66.9**	**82.5**	**65.1**	**81.0**

(N-pair) obtain encouraging performance on all the benchmark datasets. While the lifted structure and N-pair loss have obtained the outstanding results, DAML further boosts the performance to achieve the state-of-the-arts. Compared with existing methods which only exploit the observed negative samples in their original form, our DAML taps the potential of numerous easy negatives to fully describe the hard negative distributions. As DAML simultaneously trains the hard negative generator and feature embedding in an adversarial manner, the learned distance metric shows strong robustness with adequate and targeted synthetic hard negative samples. Moreover, we also find that DAMML further boosts the performance of DAML with different J_m. DAML only learns one global metric for the dataset, which may not be discriminative enough for effective distance measurement. Once a global metric fails to have the ability to correctly classify the synthetic hard negatives, the generator will not learn to create harder negatives. Instead, DAMML learns multiple local metrics for complete description of local areas, which balances the power of the hard negative generator and the distance metric. With the learned metric discriminator, the correlation of local metrics is minimized, so that they cover more local regions with less overlapping and obtains better results than DAML.

3) Comparison With Hard Negative Mining Methods: In this subsection, we compared our DAML with recent hard negative mining methods on the Cars196 dataset. We initialized with the same structure of GoogLeNet and employed the triplet loss, comparing the proposed DAML with recent sampling methods such as N-pair sampling [53], semi-hard sampling [49], and fast approximate nearest neighbour graph (FANNG) [21]. N-pair sampling aims to sample training

TABLE IX
EXPERIMENTAL RESULTS (%) ON THE CARS196 DATASET COMPARED WITH VARYING SAMPLING STRATEGIES

Method	R@1	R@2	R@4	R@8
Triplet [68]	45.1	57.4	69.7	79.2
N-pair sampling [53]	46.3	59.9	71.4	81.3
Semi-hard sampling [49]	52.4	65.2	75.1	84.3
FANNG sampling [21]	58.2	70.6	78.9	86.7
DAML	60.6	72.5	82.5	89.9
DAML + Semi-hard	**62.7**	**74.2**	**83.9**	**91.0**

triplets through the N-pair strategy. Semi-hard sampling is a easy but effective method, which aims to select hard negative samples with the constraints of easier than positive ones. FANNG develops a smart mining method to produce effective training samples with low computational costs. For fair comparisons, all the methods employ the same network structure of GoogLeNet, initialization, and triplet loss, where the only difference is the selection of the training data. Table IX shows that DAML achieves better results with the recent sampling methods. The key difference between DAML and conventional hard negative mining methods is that DAML exploits the potential of samples through hard negative generation rather than sampling. Moreover, while most existing sampling methods are especially designed for specific loss functions (usually contrastive and triplet losses), the proposed DAML presents stronger adaptability to more existing objectives such as the recent lifted and N-pair losses. From the experimental results shown in the above subsection, we observe that it is more important to boost the performance of the recent losses to achieve the state-of-the-art performance. As aforementioned, DAML does not conflict with hard negative mining, and Table IX also illustrates that the better performance is achieved by simultaneously employing hard negative generation and sampling methods. We still discover that the performance gap between DAML and DAML + Semi-hard is much smaller than that of Triplet and Triplet + Semi-hard, because DAML takes fully advantages of easy negative samples. The more important roles easy negative samples are played, the less advantages sampling strategies are brought. The experimental results also show that DAML better exploits large numbers of easy negative samples for effective model training.

4) Evaluation of the Metric Discriminator: The metric discriminator plays an important role in DAMML for data-adaptive weights allocation. In the training procedure, the metric discriminator encourages both collaborative and competitive relationships among local metrics by arranging varying weights to each metric, so that the local metrics learn effective and independent distance measurement against the hard negative generator. In the test procedure, the learned metric discriminator can also provide data-adaptive weights for local metrics to compute the final distance between a pair of samples. Compared with hand-crafted methods, the metric discriminator learns data-dependent weights for local metrics, where the distances between sample pairs obtain more precise measurement with proper local metrics.

TABLE X
EXPERIMENTAL RESULTS (%) ON THE CARS196 DATASET COMPARED WITH VARYING WEIGHTING STRATEGIES

Method	R@1	R@2	R@4	R@8
N-pair [53]	68.9	78.9	85.8	90.9
DAML	75.1	83.8	89.7	93.5
DAMML ($\mathbf{\Gamma} = \frac{1}{K}$)	75.4	83.9	89.9	93.5
DAMML	**77.8**	**86.1**	**91.8**	**95.2**

In order to test the effectiveness of the metric discriminator, we conducted an experiment by fixing $\mathbf{\Gamma} = \frac{1}{K}$ for both training and test procedures. Table X shows the experimental comparisons on the Cars196 dataset with N-pair loss for J_{m} and $K = 4$. In the training procedure, a fixed weight leads to highly correlated local metrics, as they share the same objective function and training data, and the differences among multiple metrics mainly come from varying initializations. In the test procedure, multiple local metrics present equal importance to measure the final distance between varying sample pairs, which fails to completely exploit the local information of each sample pair. We observe that with fixed weights for local metrics, DAMML still obtains comparable results with DAML due to the exploitation of local information, while the learned discriminator successfully boosts the performance of multi-metric learning with less dependency and more precise description of different pairs.

5) Stability Analysis: In order to show the stability of the proposed DAML and DAMML, we first tested the performance on the Cars196 dataset for 10 times. Table XI shows that the results are relatively stable for multiple tests.

In Fig. 7, we tested the influence of the hyperparameters. However, range of values is relatively narrow in Fig. 7. Therefore, we further show the results when λ_1 or λ_2 is not well chosen. More specifically, we fixed λ_1 and λ_2 to 1 and 50, respectively, and tested the influence of the other hyperparameter on the Cars196 dataset. Table XII and XIII show the results. We observe that the results are still stable even when choosing bad λ_1 and λ_2 for 0.1 and 1000.

6) Evaluation of Hard Positive Generation: As DAML focuses on tapping the potential of numerous negative samples, it is an interesting idea to generate hard positive samples for more effective training. As minimizing intra-class variations is one of the most important objectives for existing loss functions, hard positives also provide gradients with large magnitudes and present more important roles in model training. In this experiment, we tested the performance of hard positive generation with the triplet embedding, using the same network structure and similar objective function with J_{gen}:

$$\begin{aligned}\min_{\theta_g^p} J_{\mathrm{gen}}^p &= J_{\mathrm{hard}}^p + \lambda_1\ J_{\mathrm{reg}}^p + \lambda_2\ J_{\mathrm{adv}}^p \\ &= \sum_{i=1}^{N}(-||\widetilde{\mathbf{x}}_i^+ - \mathbf{x}_i||_2^2 + \lambda_1||\widetilde{\mathbf{x}}_i^+ - \mathbf{x}_i^+||_2^2 \\ &\quad + \lambda_2[D(\mathbf{x}_i^-, \mathbf{x}_i)^2 - D(\widetilde{\mathbf{x}}_i^+, \mathbf{x}_i)^2 - \alpha]_+). \end{aligned} \tag{19}$$

In (19), the first term aims to generate hard positive samples far from the anchors in the original feature space. The second

TABLE XI
EXPERIMENTAL RESULTS (%) ON THE CARS196 DATASET

Method	NMI	F_1	R@1	R@2	R@4	R@8
DAML (cont)	42.4 ± 1.1	11.5 ± 0.5	37.4 ± 0.4	49.5 ± 0.5	61.5 ± 0.4	73.1 ± 0.1
DAMML (cont)	44.3 ± 1.6	12.4 ± 1.2	38.9 ± 1.2	51.2 ± 1.0	63.2 ± 0.5	74.2 ± 0.5
DAML (tri)	56.6 ± 1.6	22.9 ± 1.1	60.7 ± 1.1	72.4 ± 0.9	82.3 ± 0.6	89.8 ± 0.6
DAMML (tri)	58.4 ± 1.0	25.6 ± 0.5	61.4 ± 0.4	73.1 ± 0.4	83.0 ± 0.4	90.2 ± 0.1
DAML (lifted)	63.0 ± 1.8	32.0 ± 1.4	72.3 ± 1.6	82.0 ± 1.2	88.3 ± 0.9	92.8 ± 0.5
DAMML (lifted)	64.9 ± 1.5	33.5 ± 1.1	73.6 ± 1.3	82.5 ± 0.9	89.3 ± 0.6	93.4 ± 0.4
DAML (N-pair)	66.2 ± 1.6	36.6 ± 1.1	75.3 ± 1.2	83.9 ± 0.8	89.7 ± 0.5	93.6 ± 0.4
DAMML (N-pair)	69.2 ± 1.4	38.6 ± 1.2	77.6 ± 1.3	86.1 ± 0.8	91.9 ± 0.5	95.2 ± 0.3

TABLE XII
THE R@1 PERFORMANCE (%) ON THE CARS196 DATASET OF DAML (N-PAIR) UNDER VARYING λ_1

λ_1	0.1	1	5	10	50	100	500	1000
R@1	72.2	72.9	73.6	74.4	**75.1**	75.0	74.6	73.1

TABLE XIII
THE R@1 PERFORMANCE (%) ON THE CARS196 DATASET OF DAML (N-PAIR) UNDER VARYING λ_2

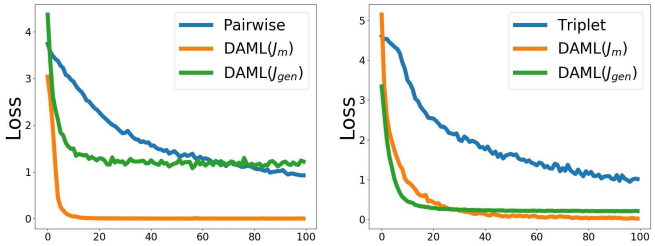
λ_2	0.1	1	5	10	50	100	500	1000
R@1	74.6	**75.1**	74.8	74.9	74.1	74.2	73.4	72.5

TABLE XIV
EXPERIMENTAL RESULTS (%) ON THE CARS196 DATASET COMPARED WITH HARD POSITIVE GENERATION

Method	R@1	R@2	R@4	R@8
Triplet	45.1	57.4	69.7	79.2
DAML (positive)	46.2	58.6	70.7	79.6
DAML (negative)	**60.6**	**72.5**	**82.5**	**89.9**

term minimizes the distance between the generated positive samples and the corresponding original samples to preserve the annotation information. The third term tries to fool the learned distance metric in a "semi-hard" manner with the margin α. Table XIV shows the comparison of hard negative generation and hard positive generation on the Cars196 dataset. We observe that the hard positive generation method obtains comparable results with the original triplet loss on the dataset. The reason is that there are usually much more negatives than positives in the training set, and it is less important to generate hard positive samples. The proposed hard negative generation approach largely outperforms hard positive generation due to the larger numbers and variations of negative samples.

7) Convergence Time: Our hardware configuration comprises of a 2.8-GHz CPU and a 32G RAM. As we applied the GoogLeNet to initialize our CNN, we utilized a GTX 1080 Ti GPU for acceleration. We compared the loss plots of DAML as well as the corresponding baselines on the Cars196 dataset as shown in Fig. 8. We plotted the average loss for each epoch,



(a) Pairwise loss (b) Triplet loss

Fig. 8. Loss plots of J_{m} and J_{gen} in DAML and different corresponding methods.

and drew the curves of J_{m} and J_{gen} with the parameter λ to balance the weights for DAML, respectively. We observe that DAML effectively accelerates the convergence of the metric term compared with the corresponding methods due to the generation of hard negative samples. Moreover, the training curves of J_m in DAML are more smooth than the original method. As DAML generates targeted hard negative samples in the training procedure, the input synthetic samples target at the weakness of the current metric for effective training at each iteration.

D. Qualitative Results

Fig. 9 shows the t-SNE [60] visualization results of DAMML (N-pair) on the CUB-200-2011 dataset. We highlight some local regions with red boxes. Due to the limitation in space, we put the visualization results of DAML (N-pair) for the CUB-200-2011, Cars196, Stanford Online Products, In-Shop Clothes Retrieval and VehicleID datasets in the supplementary material. We enlarge several specific regions to highlight the representative classes at the corner of each figure. The visualization results will be relatively dense if the dataset contains more images. We observe that even though the images from the same class suffer from large variations such as different backgrounds, viewpoints, colors, poses and configurations, the proposed DAMML (N-pair) is still able to group similar samples with effective distance measurement. With hard negative generation, the negative samples are pushed away even in their most dangerous forms, which improves the performance of the visualization results. The visualization results on the benchmark datasets demonstrate

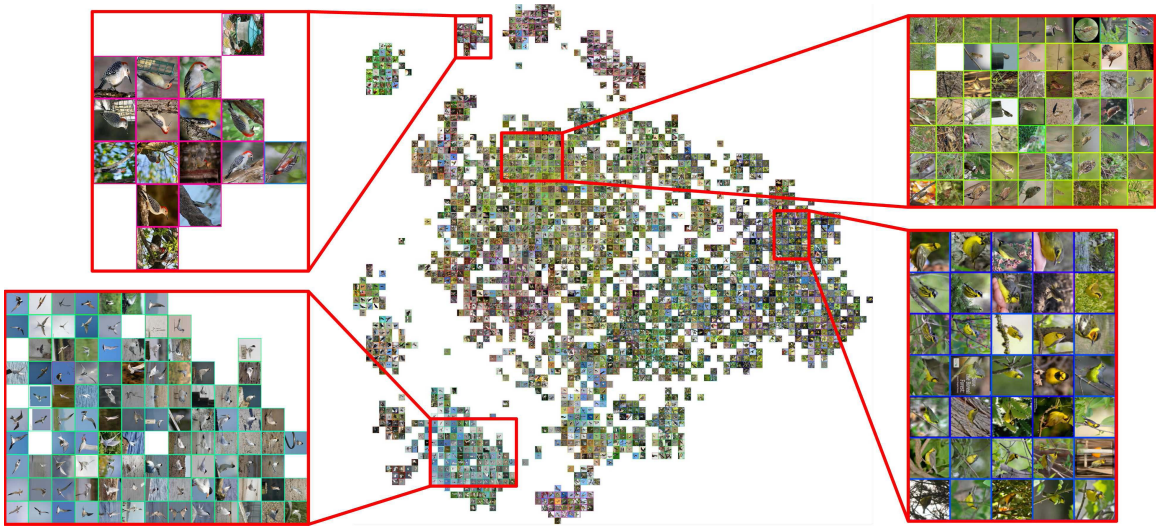


Fig. 9. Barnes-Hut t-SNE [60] visualization of the proposed DAMML (N-pair) on the CUB-200-2011 dataset, where the color of the border for each image represents the label. (Best viewed on a monitor when zoomed in.)

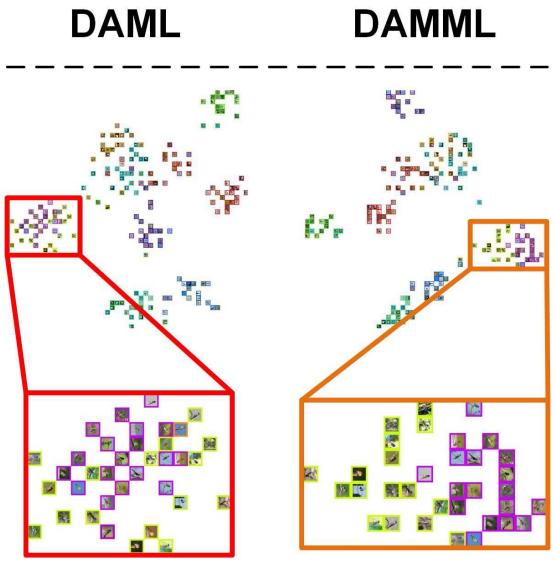


Fig. 10. Barnes-Hut t-SNE [60] visualization comparison between DAML and DAMML on the same subset of CUB-200-2011.

the effectiveness of DAMML in an intuitive manner. In order to show the comparisons between DAML and DAMML, we further visualized the embeddings of DAML (N-pair) and DAMML (N-pair) on the same randomly-selected 10 classes of CUB-200-2011 and Fig. 10 shows the result. We observe that in the highlight areas, DAMML better classifies the similar negative samples.

VII. Conclusion

In this paper, we have proposed a framework of deep adversarial metric learning (DAML), which can be generally applied to various supervised metric learning approaches. Unlike existing metric learning approaches which simply ignore a large number of easy negative samples, DAML exploits easy negatives to generate synthetic hard negatives adversarial to the learned metric as important complements to the observed samples. While the widely-used hard negative mining methods mainly focus on selecting negative samples that trigger false alarms, DAML aims to fully tap the potential of each negative sample. As the global metric may not be powerful enough to describe the whole feature space especially under the attack of the hard negative generator, we have further presented a deep adversarial multi-metric learning (DAMML) method for more precise description of local information. We have designed a metric discriminator to learn the weights for local metrics of each input pair, which encourages both collaborative and competitive relationships among metrics against the hard negative generator. With the metric discriminator, multiple local metrics present more precise final distance measurement with less correspondence. Experimental results show that DAML and DAMML effectively improve the performance of existing deep metric learning methods in an adversarial manner.

References

[1] Y. Bai, Y. Lou, F. Gao, S. Wang, Y. Wu, and L. Duan, "Group-sensitive triplet embedding for vehicle reidentification," *IEEE Trans. Multimedia*, vol. 20, no. 9, pp. 2385–2399, Sep. 2018.

[2] J. Bohné, Y. Ying, S. Gentric, and M. Pontil, "Large margin local metric learning," in *Proc. ECCV*, 2014, pp. 679–694.

[3] K. Bunte, P. Schneider, B. Hammer, F.-M. Schleif, T. Villmann, and M. Biehl, "Limited rank matrix learning, discriminative dimension reduction and visualization," *Neural Netw.*, vol. 26, pp. 159–173, Feb. 2012.

[4] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 2172–2180.

[5] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proc. CVPR*, Jun. 2005, pp. 539–546.

[6] Y. Cui, F. Zhou, Y. Lin, and S. Belongie, "Fine-grained categorization and dataset bootstrapping using deep metric learning with humans in the loop," in *Proc. CVPR*, Jun. 2016, pp. 1153–1162.

[7] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon, "Information-theoretic metric learning," in *Proc. ICML*, Jun. 2007, pp. 209–216.

[8] Z. Ding and Y. Fu, "Robust transfer metric learning for image classification," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 660–670, Feb. 2017.

[9] Z. Ding, M. Shao, and Y. Fu, "Generative zero-shot learning via low-rank embedded semantic dictionary," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published.

[10] Y. Duan, L. Chen, J. Lu, and J. Zhou, "Deep embedding learning with discriminative sampling policy," in *Proc. CVPR*, Jun. 2019, pp. 4964–4973.

[11] Y. Duan, J. Lu, J. Feng, and J. Zhou, "Deep localized metric learning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 10, pp. 2644–2656, Oct. 2018.

[12] Y. Duan, J. Lu, Z. Wang, J. Feng, and J. Zhou, "Learning deep binary descriptor with multi-quantization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1924–1938, Aug. 2019.

[13] Y. Duan, Z. Wang, J. Lu, X. Lin, and J. Zhou, "GraphBit: Bitwise interaction mining via deep reinforcement learning," in *Proc. CVPR*, Jun. 2018, pp. 8270–8279.

[14] Y. Duan, W. Zheng, X. Lin, J. Lu, and J. Zhou, "Deep adversarial metric learning," in *Proc. CVPR*, Jun. 2018, pp. 2780–2789.

[15] Z. Feng, R. Jin, and A. Jain, "Large-scale image annotation by efficient and robust kernel metric learning," in *Proc. ICCV*, Dec. 2013, pp. 1609–1616.

[16] W. Ge, "Deep metric learning with hierarchical triplet loss," in *Proc. ECCV*, Sep. 2018, pp. 269–285.

[17] A. Globerson and S. T. Roweis, "Metric learning by collapsing classes," in *Proc. Adv. Neural Inf. Process. Syst.*, 2006, pp. 451–458.

[18] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[19] M. Guillaumin, J. Verbeek, and C. Schmid, "Is that you? Metric learning approaches for face identification," in *Proc. CVPR*, Sep./Oct. 2009, pp. 498–505.

[20] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *Proc. CVPR*, Jun. 2006, pp. 1735–1742.

[21] B. Harwood, G. Carneiro, I. Reid, and T. Drummond, "Smart mining for deep metric learning," in *Proc. ICCV*, Oct. 2017, pp. 2821–2829.

[22] J. Hu, J. Lu, and Y.-P. Tan, "Deep metric learning for visual tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 11, pp. 2056–2068, Nov. 2016.

[23] C. Huang, C. C. Loy, and X. Tang, "Local similarity-aware deep feature embedding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 1262–1270.

[24] S. Huang, J. Lu, J. Zhou, and A. K. Jain, "Nonlinear local metric learning for person re-identification," 2015, *arXiv:1511.05169*. [Online]. Available: https://arxiv.org/abs/1511.05169

[25] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. CVPR*, Jul. 2017, pp. 1125–1134.

[26] S. Jiang, Y. Wu, and Y. Fu, "Deep bidirectional cross-triplet embedding for online clothing shopping," *ACM Trans. Multimedia Comput., Commun.*, vol. 14, no. 1, p. 5, 2018.

[27] D. Kedem, S. Tyree, F. Sha, G. R. Lanckriet, and K. Q. Weinberger, "Non-linear metric learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 2573–2581.

[28] W. Kim, B. Goyal, K. Chawla, J. Lee, and K. Kwon, "Attention-based ensemble for deep metric learning," in *Proc. ECCV*, Sep. 2018, pp. 736–751.

[29] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3D object representations for fine-grained categorization," in *Proc. ICCV*, Jun. 2013, pp. 554–561.

[30] B. Kulis, "Metric learning: A survey," *Found. Trends Mach. Learn.*, vol. 5, no. 4, pp. 287–364, 2013.

[31] M. T. Law, N. Thome, and M. Cord, "Quadruplet-wise image similarity learning," in *Proc. ICCV*, Dec. 2013, pp. 249–256.

[32] S. Liao, Y. Hu, X. Zhu, and S. Z. Li, "Person re-identification by local maximal occurrence representation and metric learning," in *Proc. CVPR*, Jun. 2015, pp. 2197–2206.

[33] X. Lin, Y. Duan, Q. Dong, J. Lu, and J. Zhou, "Deep variational metric learning," in *Proc. ECCV*, Sep. 2018, pp. 689–704.

[34] H. Liu, Y. Tian, Y. Yang, L. Pang, and T. Huang, "Deep relative distance learning: Tell the difference between similar vehicles," in *Proc. CVPR*, Jun. 2016, pp. 2167–2175.

[35] M.-Y. Liu and O. Tuzel, "Coupled generative adversarial networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 469–477.

[36] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, "DeepFashion: Powering robust clothes recognition and retrieval with rich annotations," in *Proc. CVPR*, Jun. 2016, pp. 1096–1104.

[37] Z. Liu, D. Wang, and H. Lu, "Stepwise metric promotion for unsupervised video person re-identification," in *Proc. ICCV*, Oct. 2017, pp. 2429–2438.

[38] J. Lu, J. Hu, and Y.-P. Tan, "Discriminative deep metric learning for face and kinship verification," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4269–4282, Sep. 2017.

[39] B. Mahasseni, M. Lam, and S. Todorovic, "Unsupervised video summarization with adversarial LSTM networks," in *Proc. CVPR*, Jul. 2017, pp. 202–211.

[40] I. Masi, A. T. Trần, J. T. Leksut, T. Hassner, and G. G. Medioni, "Do we really need to collect millions of faces for effective face recognition?" in *Proc. ECCV*, 2016, pp. 579–596.

[41] Y. Movshovitz-Attias, A. Toshev, T. K. Leung, S. Ioffe, and S. Singh, "No fuss distance metric learning using proxies," in *Proc. ICCV*, Oct. 2017, pp. 360–368.

[42] M. Opitz, G. Waltner, H. Possegger, and H. Bischof, "Deep metric learning with BIER: Boosting independent embeddings robustly," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published.

[43] S. Paisitkriangkrai, C. Shen, and A. van den Hengel, "Learning to rank in person re-identification with metric ensembles," in *Proc. CVPR*, Jun. 2015, pp. 1846–1855.

[44] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proc. CVPR*, Jun. 2016, pp. 2536–2544.

[45] Q. Qian, J. Tang, H. Li, S. Zhu, and R. Jin, "Large-scale distance metric learning with uncertainty," in *Proc. CVPR*, Jun. 2018, pp. 8542–8550.

[46] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[47] T. Salimans *et al.*, "Improved techniques for training GANs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 2234–2242.

[48] P. Schneider, M. Biehl, and B. Hammer, "Adaptive relevance matrices in learning vector quantization," *Neural Comput.*, vol. 21, no. 12, pp. 3532–3561, 2009.

[49] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. CVPR*, Jun. 2015, pp. 815–823.

[50] M. Schultz and T. A. Joachims, "Learning a distance metric from relative comparisons," in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 41–48.

[51] S. Shalev-Shwartz, Y. Singer, and A. Y. Ng, "Online and batch learning of pseudo-metrics," in *Proc. ICML*, 2004, p. 94.

[52] A. Shrivastava, A. Gupta, and R. Girshick, "Training region-based object detectors with online hard example mining," in *Proc. CVPR*, Jun. 2016, pp. 761–769.

[53] K. Sohn, "Improved deep metric learning with multi-class n-pair loss objective," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 1849–1857.

[54] H. O. Song, S. Jegelka, V. Rathod, and K. Murphy, "Deep metric learning via facility location," in *Proc. CVPR*, Jul. 2017, pp. 5382–5390.

[55] H. O. Song, Y. Xiang, S. Jegelka, and S. Savarese, "Deep metric learning via lifted structured feature embedding," in *Proc. CVPR*, Jun. 2016, pp. 4004–4012.

[56] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. CVPR*, Jun. 2015, pp. 1–9.

[57] L. Tran, X. Yin, and X. Liu, "Disentangled representation learning gan for pose-invariant face recognition," in *Proc. CVPR*, Jul. 2017, pp. 1415–1424.

[58] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. CVPR*, Jul. 2017, pp. 7167–7176.

[59] E. Ustinova and V. Lempitsky, "Learning deep embeddings with histogram loss," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 4170–4178.

[60] L. van der Maaten, "Accelerating t-SNE using tree-based algorithms," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 3221–3245, Oct. 2014.

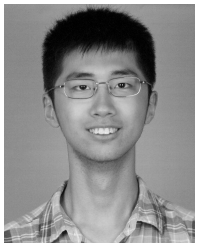
[61] C. Vondrick and A. Torralba, "Generating the future with adversarial transformers," in *Proc. CVPR*, Jul. 2017, pp. 1020–1028.

[62] C. Wah, S. Branson, P. Welinder, P. Perona, and S. J. Belongie, "The caltech-ucsd birds-200-2011 dataset," California Inst. Technol., Pasadena, CA, USA, Tech. Rep. CNS-TR-2011-001, 2011.

[63] J. Wang, F. Zhou, S. Wen, X. Liu, and Y. Lin, "Deep metric learning with angular loss," in *Proc. ICCV*, Oct. 2017, pp. 2593–2601.

[64] S. Wang, Z. Ding, and Y. Fu, "Cross-generation kinship verification with sparse discriminative metric," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 11, pp. 2783–2790, Nov. 2019.

[65] X. Wang and A. Gupta, "Unsupervised learning of visual representations using videos," in *Proc. ICCV*, Dec. 2015, pp. 2794–2802.
[66] X. Wang, G. Hua, and T. X. Han, "Discriminative tracking by metric learning," in *Proc. ECCV*, 2010, pp. 200–214.
[67] X. Wang, A. Shrivastava, and A. Gupta, "A-Fast-RCNN: Hard positive generation via adversary for object detection," in *Proc. CVPR*, Jul. 2017, pp. 2606–2615.
[68] K. Q. Weinberger and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," *J. Mach. Learn. Res.*, vol. 10, pp. 207–244, Feb. 2009.
[69] K. Q. Weinberger and G. Tesauro, "Metric learning for kernel regression," in *Proc. Artif. Intell. Statist.*, 2007, pp. 612–619.
[70] C.-Y. Wu, R. Manmatha, A. J. Smola, and P. Krahenbuhl, "Sampling matters in deep embedding learning," in *Proc. ICCV*, Oct. 2017, pp. 2840–2848.
[71] H.-J. Ye, D.-C. Zhan, Y. Jiang, and Z.-H. Zhou, "What makes objects similar: A unified multi-metric learning approach," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 5, pp. 1257–1270, May 2019.
[72] H.-X. Yu, A. Wu, and W.-S. Zheng, "Cross-view asymmetric metric learning for unsupervised person re-identification," in *Proc. ICCV*, Oct. 2017, pp. 994–1002.
[73] Y. Yuan, K. Yang, and C. Zhang, "Hard-aware deeply cascaded embedding," in *Proc. ICCV*, Oct. 2017, pp. 814–823.
[74] J. Zhou, P. Yu, W. Tang, and Y. Wu, "Efficient online local metric adaptation via negative samples for person re-identification," in *Proc. ICCV*, Oct. 2017, pp. 2420–2428.
[75] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. ICCV*, Oct. 2017, pp. 2223–2232.



Yueqi Duan received the B.S. and Ph.D. degrees from the Department of Automation, Tsinghua University, China, in 2014 and 2019, respectively. He is currently a Postdoctoral Researcher with the Computer Science Department, Stanford University. His current research interests include 3D vision, unsupervised learning, metric learning, and binary representation learning. He has authored 14 scientific articles in these areas, where 11 articles are published in top journals and conferences, including the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON IMAGE PROCESSING, CVPR, and ECCV. He received the National Scholarship of Tsinghua in 2017 and 2018, respectively. He serves as a regular reviewer member for a number of journals and conferences, e.g., TPAMI, IJCV, TIP, TIFS, TCSVT, *Pattern Recognition*, CVPR, ICCV, ICME, and ICIP.

Jiwen Lu (M'11–SM'15) received the B.Eng. degree in mechanical engineering and the M.Eng. degree in electrical engineering from the Xi'an University of Technology, Xi'an, China, in 2003 and 2006, respectively, and the Ph.D. degree in electrical engineering from Nanyang Technological University, Singapore, in 2012. He is currently an Associate Professor with the Department of Automation, Tsinghua University, Beijing, China. His current research interests include computer vision, pattern recognition, and machine learning. He has authored/coauthored over 200 scientific articles in these areas, where over 70 of them are IEEE Transactions papers and over 50 of them are CVPR/ICCV/ECCV articles. He is a member of the Multimedia Signal Processing Technical Committee and the Information Forensics and Security Technical Committee of the IEEE Signal Processing Society, and a member of the Multimedia Systems and Applications Technical Committee and the Visual Signal Processing and Communications Technical Committee of the IEEE Circuits and Systems Society. He was a recipient of the National Science Fund of China for Excellent Young Scholars. He serves as the Co-Editor-of-Chief of the *Pattern Recognition Letters*, an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE TRANSACTIONS ON BIOMETRICS, BEHAVIOR, AND IDENTITY SCIENCE, and *Pattern Recognition*.

Wenzhao Zheng received the B.S. degree from the Department of Physics, Tsinghua University, China, in 2018, where he is currently pursuing the Ph.D. degree with the Department of Automation. His research interests include computer vision, deep learning, and metric learning.

Jie Zhou (M'01–SM'04) received the B.S. and M.S. degrees from the Department of Mathematics, Nankai University, Tianjin, China, in 1990 and 1992, respectively, and the Ph.D. degree from the Institute of Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology (HUST), Wuhan, China, in 1995. Since 1997, he has been a Postdoctoral Fellow with the Department of Automation, Tsinghua University, Beijing, China, where he has been a Full Professor, since 2003. He has authored over 100 articles in peer-reviewed journals and conferences. Among them, over 70 articles have been published in top journals and conferences, such as the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON IMAGE PROCESSING, and CVPR. His research interests include computer vision, pattern recognition, and image processing. He received the National Outstanding Youth Foundation of China Award. He is an Associate Editor for the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE and two other journals.