

# Deep Adversarial Metric Learning

Yueqi Duan<sup>1,2,3</sup>, Wenzhao Zheng<sup>1</sup>, Xudong Lin<sup>1</sup>, Jiwen Lu<sup>1,2,3</sup>, Jie Zhou<sup>1,2,3</sup>

<sup>1</sup>Department of Automation, Tsinghua University, China

<sup>2</sup>State Key Lab of Intelligent Technologies and Systems, China

<sup>3</sup>Beijing National Research Center for Information Science and Technology, China

duanyq14@mails.tsi nghua. edu. cn; zhengwz14@mails. tsi nghua. edu. cn; l i nxd14@mails. tsi nghua. edu. cn;

l uj iwen@tsi nghua. edu. cn; j zhou@tsi nghua. edu. cn

## Abstract

Learning an effective distance metric between image pairs plays an important role in visual analysis, where the training procedure largely relies on hard negative samples. However, hard negatives in the training set usually account for the tiny minority, which may fail to fully describe the distribution of negative samples close to the margin. In this paper, we propose a deep adversarial metric learning (DAML) framework to generate synthetic hard negatives from the observed negative samples, which is widely applicable to supervised deep metric learning methods. Different from existing metric learning approaches which simply ignore numerous easy negatives, the proposed DAML exploits them to generate potential hard negatives adversarial to the learned metric as complements. We simultaneously train the hard negative generator and feature embedding in an adversarial manner, so that more precise distance metrics can be learned with adequate and targeted synthetic hard negatives. Extensive experimental results on three benchmark datasets including CUB-200-2011, Cars196 and Stanford Online Products show that DAML effectively boosts the performance of existing deep metric learning approaches through adversarial learning.

## 1. Introduction

Metric learning aims to learn a distance metric for image pairs to measure their similarities, which makes the following classification and clustering tasks much simpler. Metric learning approaches have been widely used in a variety of visual analysis tasks, such as face recognition [8, 12], person re-identification [50, 48, 19], visual tracking [42, 11], and image classification [4, 44, 3]. Existing metric learning methods can be divided into two categories: linear and non-linear [16]. Conventional linear metric learning methods

Figure 1. Comparisons of conventional metric learning methods and our DAML. For intuitive demonstration, we use the number of “3” as the anchor and positive samples while other numbers and alphabets are negative samples. We compute the distances between sample pairs based on the similarity of shapes. Existing metric learning methods largely rely on a few observed hard negatives, pushing the negative distribution to the lower right, which fail to handle the potential hard negatives in the upper right of the margin. For DAML, we aim to generate potential hard negatives from existing negative samples adversarial to the learned metric, where numerous easy negatives are exploited as complements. We simultaneously learn the generator and feature embedding to obtain better similarity estimation with adequate and targeted synthetic negative samples, where potential hard negatives in the unobserved space are also pushed away. (Best viewed in color.)

learn a Mahalanobis distance metric [4, 44, 8], while non-linear methods apply kernel tricks or deep neural networks to model high-order correlations [34, 3, 32, 33, 40, 22].

For most supervised metric learning methods, the training procedure is based on an objective that maximizes the inter-class variations and minimizes the intra-class varia-

\* Corresponding author

tions [12, 10]. Therefore, the hard negative and positive samples in the training set will produce gradients with large magnitudes while others are close to zero. As hard negatives usually account for the tiny minority, the vast majority of negative samples, which are considered as “easy negatives”, make little contribution to metric learning. A natural question is raised: are easy negatives really useless?

In this work, we provide an answer supporting the contrary by arguing that some easy negatives may have potential to generate important complements to existing hard negatives, which should not be ignored. For example, the letter “W” seems different from the number “3” and is regarded as an easy negative. However, it would become a dangerous hard negative after a rotation of 90 degrees counterclockwise. We consider another example in face recognition. Although an Asian female may look quite different from an European male, their son with a similar age may be his potential dangerous hard negative. While all the existing hard negatives (other European males) fail to provide effective guidance to this multiracial boy, his mother may have the ability as an easy negative. Figure 1 illustrates the reason of this phenomenon. Hard negatives usually account for the tiny minority in the training set, which may not be enough to fully describe the distribution of hard negative samples. Existing metric learning approaches simply maximize the relative distance of the observed hard negative space, while potential hard negatives in the unobserved space are still in danger with the learned distance metric.

In this paper, we propose a deep adversarial metric learning (DAML) framework to address the limitation, which can be generally adapted to existing supervised deep metric learning approaches. Instead of simply utilizing the observed data, our goal is to generate potential hard negatives from easy ones, so that a large number of easy negatives can be exploited to provide important synthetic complements. The procedure of hard negative generation simultaneously follows three losses: 1) the synthetic samples should be close to the anchor in the original feature space, 2) the synthetic samples should preserve the annotation information, and 3) the synthetic samples should be misclassified by the learned metric. We simultaneously train the hard negative generator and feature embedding in an adversarial manner to obtain adequate and targeted synthetic hard negatives. Adequate hard negatives give a complete description of the negative distribution close to the margin, while targeted hard negatives aim at the limitations of the current feature embedding. Figure 1 shows the comparisons between conventional metric learning methods and DAML. It is important to notice that the proposed hard negative generation framework does not conflict with the widely-used hard negative mining, as we can perform negative generation at first to provide plenty of hard negatives for sampling. Extensive experimental results on three benchmark datasets

illustrate that the proposed DAML framework improves the performance of the existing supervised deep metric learning methods in an adversarial manner.

## 2. Related Work

**Metric Learning:** The field of metric learning has witnessed great progress over the past decade, which aims to learn effective metrics to measure the similarities of the input image pairs. Conditional metric learning approaches learn a linear Mahalanobis distance to measure the similarities of samples, where a number of methods have been proposed [29, 30, 6, 44, 4]. For example, Weinberger et al. [44] proposed a large margin nearest neighbor (LMNN) approach by enforcing an anchor point to share the same labels with its nearest neighbors by a relatively large margin, which is one of the most popular linear metric learning approaches in the literature. Davis et al. [4] presented an information-theoretic metric learning (ITML) approach to formulate the problem as a constrained optimization task by minimizing a regularizer of LogDet divergence.

As linear metric learning approaches may suffer from nonlinear correlations of samples, kernel tricks are usually adopted to address the limitation [45, 5, 20, 47]. However, it is quite empirical for choosing a kernel, and their discriminative power is also limited. With the outstanding performance of deep learning in various visual analysis tasks, deep metric learning approaches have been proposed to learn non-linear mappings [24, 17, 3, 34, 13, 36, 12, 33, 40, 22, 23]. For example, Hu et al. [12] learned a discriminative distance metric with deep neural networks. Song et al. [34] presented a lift structure to take full advantage of training batches. Ustinova and Lempitsky [36] proposed a histogram loss for deep embedding learning by estimating the distribution of similarities for positive and negative pairs. Wang et al. [40] proposed an angular loss by constraining the angle relationships inside the triplets at the negative point. In general, deep metric learning approaches present strong discriminative power, which achieve the state-of-the-art performance. However, these methods ignore a large number of easy negatives, which may have potential to generate essential complements.

**Hard Negative Mining:** In many visual analysis tasks, hard negative mining is applied to better exploit large-scale negative data for model training [41, 28, 31, 33, 10, 49]. Hard negative mining can be seen as a problem of bootstrapping, which gradually selects negative samples that trigger false alarms [31]. For example, Schroff et al. [28] selected “semi-hard” negative samples to train FaceNet with triplet loss, which are hard but still farther than the positive-anchor pairs. Shrivastava et al. [31] proposed an online hard example mining (OHEM) algorithm to train region-based object detectors. Wu et al. [46] showed the significant importance of sample selection in embedding learning and pro-

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

posed distance weighted sampling with margin based loss. Harwood et al. [10] proposed a smart mining procedure to efficiently select effective training samples for deep metric learning. Yuan et al. [49] presented a hard-aware deeply cascaded (HDC) embedding approach by mining negatives at multiple hard levels based on the models.

Rather than sampling existing hard negative samples for data mining, we focus on the exploitation of a large number of easy negative samples which may have potential to generate essential complements. Inspired by the recent works in adversarial learning [7, 27, 25, 18, 2, 46, 43, 38, 14], we aim to generate synthetic negatives from existing ones that shorten the distance to the anchor, minimize the difference between synthetic and observed negative samples, and confuse the learned metric. Different from most existing adversarial learning methods which aim to model the image distributions, the proposed DAML taps the potential of the training data in the feature space to enhance the discriminative power of the learned distance metric.

### 3. Proposed Approach

In this section, we first present the hard negative generator, and then detail the approach of deep adversarial metric learning. Lastly, we present the discussions to highlight the difference with existing methods and introduce the implementation details of the proposed DAML.

#### 3.1. Hard Negative Generator

To the best of our knowledge, existing metric learning approaches take advantage of the observed data to train 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 key limitations of the existing approaches:

- 1) The observed hard negatives may not be enough to fully describe the distributions of negative samples near

the margin, as shown in Figure 1. In some cases, most hard negative samples only belong to a few identities, which suffer from limited diversities. The inadequate hard negatives may lead to local optimal distance metrics, where potential hard negatives in the unobserved space would be misclassified.

- 2) A large number of easy negative samples are wasted which produce gradients close to zero. However, some of the easy negatives have potential to generate synthetic 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 easy ones against the learned metric to simultaneously address the above two limitations. Figure 2 shows the framework of the proposed DAML compared with the conventional metric learning methods. Let  $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n]$  be the input features and  $\mathbf{Y} = [y_1, \dots, y_n]$  be the corresponding labels, where  $y_i \in \{1, \dots, C\}$ . We employ the widely-used triplet embeddings and contrastive embeddings for explanation. The triplet input  $\{\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-\}$  composes 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  $y_i^- = y_i$ , while the pairwise input for contrastive embedding utilizes  $\{\mathbf{x}_i, \mathbf{x}_i^+\}$  and  $\{\mathbf{x}_j, \mathbf{x}_j^-\}$ .

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

$$D(\mathbf{x}_i, \mathbf{x}_j) = f(\boldsymbol{\phi}; \mathbf{x}_i, \mathbf{x}_j), \quad (1)$$

where  $D$  is the distance between an input pair under the learned metric,  $f$  is the metric function, and  $\boldsymbol{\phi}$  is the learned parameters of  $f$ .

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

$$f(\boldsymbol{\phi}; \mathbf{x}_i, \mathbf{x}_j) = \overline{(\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{M} (\mathbf{x}_i - \mathbf{x}_j)}, \quad (2)$$

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

Most supervised metric learning approaches aim to obtain the parameters  $\boldsymbol{\phi}$  through optimizing a well-designed objective function:

$$\boldsymbol{\phi} = \arg \min_{\boldsymbol{\phi}} J_m(\boldsymbol{\phi}; \mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-, \mathbf{f}), \quad (3)$$

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

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

$$\boldsymbol{\phi}^a = \arg \min_{\boldsymbol{\phi}} J_m(\boldsymbol{\phi}; \mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-, \mathbf{f}), \quad (4)$$

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

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

$$\mathbf{x}_i^- = G(\theta_g; \mathbf{x}_i^-, \mathbf{x}_i, \mathbf{x}_i^+), \quad (5)$$

and  $\theta_g$  is the parameters of the generator  $G$ .

In (5), we aim to generate synthetic negative samples from original ones, so that more easy negatives can be exploited as complements to the observed data. As each negative point would generate different synthetic samples according to 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:

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

where  $N$  is the number of the inputs,  $\gamma$  is a enforced margin between positive-anchor pairs and negative-anchor pairs, the operation of  $[\cdot]_+$  refers to the hinge function  $\max(0, \cdot)$ , and  $\lambda_1$  and  $\lambda_2$  are two parameters to balance the weights of different terms.

$J_{\text{hard}}$  aims to make the synthetic negatives close to the anchor, which would produce large magnitudes for the training procedure of metric learning.  $J_{\text{reg}}$  is a self-regularization term to minimize the difference between the generated negatives and the original negatives. The goal of  $J_{\text{adv}}$  is 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  $\gamma$ . The procedure of adversarial training enhances the discriminative power of the learned metrics to address potential unobserved hard negatives.

### 3.2. Deep Adversarial Metric Learning

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

$$\min_{\theta_g, \theta_f} J = J_{\text{gen}} + \lambda J_m, \quad (7)$$

where  $\lambda$  is the parameter to balance the weights 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 [9, 12] to define the objective function as:

$$J_m = \sum_{i=1}^{N_i} D(\mathbf{x}_i^+, \mathbf{x}_i)^2 + \sum_{j=1}^{N_j} [D(\mathbf{x}_j^-, \mathbf{x}_j)^2 - \gamma]_+, \quad (8)$$

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

**DAML (tri):** For triplet embeddings, we employ [44, 28] to define the objective function, which is widely used for the triplet input:

$$J_m = \sum_{i=1}^N [D(\mathbf{x}_i^+, \mathbf{x}_i)^2 - D(\mathbf{x}_i^-, \mathbf{x}_i)^2 + \gamma]_+, \quad (9)$$

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

---

**Algorithm 1: DAML**

---

**Input:** Training image set, parameters  $\alpha_1$  and  $\alpha_2$ , margin  $\alpha$ , and iteration numbers  $T$ .

**Output:** Parameters of the hard negative generator  $\theta_g$ , and parameters of the metric function  $\theta_f$ .

```

1: Pre-train  $\theta_f$  without the hard negative generator.
2: Initialize  $\theta_g$ .
3: for iter = 1, 2, ..., T do
4:   Sample minibatch of  $m$  training images.
5:   Produce triplet or pairwise inputs from the batch.
6:   Jointly optimize  $\theta_g$  and  $\theta_f$  using (7).
7: end for
8: return  $\theta_g$  and  $\theta_f$ .
```

---

**DAML (lifted):** We also employ [34] for the lifted structure to define the objective function as follows:

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

$$J_{i^+, i} = \max(\max_{c \neq i} -D(x_i^+, x_c), \max_{c \neq i} -D(x_i, x_c) + D(x_i^+, x_i)), \quad (11)$$

where  $D(x)$  represents the distances of the negative pairs for  $x$ . We suggest referring [34] for more details.

**DAML (N-pair):** In the N-pair loss [32], anchor from each class  $x_c$  would have one positive sample  $x_c^+$  and  $C - 1$  negative samples  $x_c^-$ , where  $C$  is the number of classes and  $c = 1, \dots, C$ . For each  $x_c$  and  $x_c^+$ , we generate  $C - 1$  synthetic hard negatives  $x_c^+$  from  $x_c^+$  through the generator. The metric term of DAML (N-pair) is defined as follows:

$$J_m = \frac{1}{C} \sum_{c=1}^C \log(1 + \exp(D(x_c, x_c^+) - D(x_c, x_c^+))) \quad (12)$$

where  $D(x_i, x_j) = f_i^T f_j$  is the similarity used in the N-pair loss, and  $f_i$  and  $f_j$  are the embedded features. See [32] for complete details.

We train the hard negative generator and the distance metric in a joint manner, and Figure 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 adversarial to the pre-trained metric. Lastly, we jointly optimize both networks during each iteration end-to-end, where the synthetic hard negatives are used for training the distance metric. In the test procedure, as the CNNs and fully connected layers have the shared structures and parameters, we apply the metric network for similarity measurement without the generator. Algorithm 1 details the approach of DAML.

### 3.3. Discussion

In this subsection, we compare the proposed DAML with hard negative mining and data augmentation respectively to highlight the differences.

**Difference with Existing Hard Negative Mining Methods:** Hard negative mining has been widely used in many visual analysis tasks and has successfully boosted the performance of metric learning [10, 49]. The key 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 chosen by the miner in their 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 90 degrees counter-clockwise, which may be even harder than all the observed negatives. In general, hard negative mining selects useful existing samples, while DAML taps their potential. Moreover, we emphasize that DAML does not conflict with hard negative mining, where it can generate more negative samples at first for the following full selections.

**Difference with Existing Data Augmentation Methods:** Data augmentation aims to apply transformation to the images without altering the labels, which have been widely used to improve the performance of CNN and prevent from overfitting [21]. The key difference between DAML and data augmentation is that we simultaneously train the generator and feature embedding in an adversarial manner to obtain metric-specific synthetic hard negatives rather than applying fixed transformation to all the images. The generated samples especially target at the limitations of the current feature embedding for effective direction. 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 which presents stronger flexibility.

### 3.4. Implementation Details

We utilized the TensorFlow [1] package through the experiments. We normalized the images into  $256 \times 256$  at first, and then we performed standard random crop and horizontal mirroring for data augmentation. For the metric network, we performed the initialization with GoogLeNet [35] which was pretrained on the ImageNet ILSVRC dataset [26], 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 features as the input and generating the synthetic negative as the output. We empirically fixed the parameters  $\alpha_1$  and  $\alpha_2$  as 1, 1 and 50 to balance the weights of different terms, respectively, and we followed [44] by setting



to 1. As an experimental study in [34] shows that the embedding size does not largely affect the performance, we followed [40] to fix the embedding size to 512 throughout the experiment. 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.

## 4. Experiments

In this section, we conducted experiments on three widely-used benchmark datasets for both retrieval and clustering tasks to demonstrate the effectiveness of the proposed DAML, which included the CUB-200-2011 [39], Cars196 [15] and Stanford Online Products [34] datasets. For the clustering task, we followed [34] to perform K-means algorithm in the test set, using the normalized mutual information (NMI) and  $F_1$  metrics. The input of NMI is a set of clusters  $\mathcal{C} = \{c_1, \dots, c_K\}$  and the ground truth classes  $\mathcal{C} = \{c_1, \dots, c_K\}$ , where  $c_i$  represents the samples that belong to the  $i$ th 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  $NMI(\mathcal{C}, \mathcal{C}) = \frac{2I(\mathcal{C}, \mathcal{C})}{H(\mathcal{C}) + H(\mathcal{C})}$ , and  $F_1$  metric is the harmonic mean of precision and recall as follows  $F_1 = \frac{2PR}{P+R}$ . For the retrieval task, we computed the percentage of test samples which have at least one example from the same category in  $R$  nearest neighbors.

### 4.1. Datasets

We conducted experiments on three widely-used benchmark datasets to evaluate the proposed DAML with the standard evaluation protocol [34, 33, 40].

- 1) The CUB-200-2011 dataset [39] includes 11,788 images of 200 bird species. We used the first 100 species with 5,864 images for training and the rest for testing.
- 2) The Cars196 dataset [15] contains 16,185 images of 196 car models. We used the first 98 models with 8,054 images for training and the remaining for testing.
- 3) The Stanford Online Products dataset [34] has 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 others for testing.

### 4.2. Baseline Methods

We applied the framework of adversarial metric learning on three baseline methods as aforementioned for direct comparisons, which include the widely-used contrastive embedding [9], triplet embedding [44] and the more recent lifted structure [34] and N-pair loss [32]. We also compared DAML with other four baseline methods for evaluation including DDML [12], triplet loss with N-pair sam-

Table 1. 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	47.3	13.1	31.2	41.6	54.7	67.1
Triplet+N-pair	54.1	20.0	42.8	54.9	66.2	77.6
Angular	<b>61.0</b>	<b>30.2</b>	<b>53.6</b>	<b>65.0</b>	<b>75.3</b>	<b>83.7</b>
Contrastive	47.2	12.5	27.2	36.3	49.8	62.1
DAML (cont)	<b>49.1</b>	<b>16.2</b>	<b>35.7</b>	<b>48.4</b>	<b>60.8</b>	<b>73.6</b>
Triplet	49.8	15.0	35.9	47.7	59.1	70.0
DAML (tri)	<b>51.3</b>	<b>17.6</b>	<b>37.6</b>	<b>49.3</b>	<b>61.3</b>	<b>74.4</b>
Lifted	56.4	22.6	46.9	59.8	71.2	81.5
DAML (lifted)	<b>59.5</b>	<b>26.6</b>	<b>49.0</b>	<b>62.2</b>	<b>73.7</b>	<b>83.3</b>
N-pair	60.2	28.2	51.9	64.3	74.9	83.2
DAML (N-pair)	<b>61.3</b>	<b>29.5</b>	<b>52.7</b>	<b>65.4</b>	<b>75.5</b>	<b>84.3</b>

Table 2. Experimental results (%) on the Cars196 dataset compared with baseline methods.

Method	NMI	$F_1$	R@1	R@2	R@4	R@8
DDML	41.7	10.9	32.7	43.9	56.5	68.8
Triplet+N-pair	54.3	19.6	46.3	59.9	71.4	81.3
Angular	62.4	31.8	71.3	80.7	87.0	91.8
Contrastive	42.3	10.5	27.6	38.3	51.0	63.9
DAML (cont)	<b>42.6</b>	<b>11.4</b>	<b>37.2</b>	<b>49.6</b>	<b>61.8</b>	<b>73.3</b>
Triplet	52.9	17.9	45.1	57.4	69.7	79.2
DAML (tri)	<b>56.5</b>	<b>22.9</b>	<b>60.6</b>	<b>72.5</b>	<b>82.5</b>	<b>89.9</b>
Lifted	57.8	25.1	59.9	70.4	79.6	87.0
DAML (lifted)	<b>63.1</b>	<b>31.9</b>	<b>72.5</b>	<b>82.1</b>	<b>88.5</b>	<b>92.9</b>
N-pair	62.7	31.8	68.9	78.9	85.8	90.9
DAML (N-pair)	<b>66.0</b>	<b>36.4</b>	<b>75.1</b>	<b>83.8</b>	<b>89.7</b>	<b>93.5</b>

Table 3. Experimental results (%) on the Stanford Online Products dataset compared with baseline methods.

Method	NMI	$F_1$	R@1	R@10	R@100
DDML	83.4	10.7	42.1	57.8	73.7
Triplet+N-pair	86.4	21.0	58.1	76.0	89.1
Angular	87.8	26.5	<b>67.9</b>	<b>83.2</b>	92.2
Contrastive	82.4	10.1	37.5	53.9	71.0
DAML (cont)	<b>83.5</b>	<b>10.9</b>	<b>41.7</b>	<b>57.5</b>	<b>73.5</b>
Triplet	86.3	20.2	53.9	72.1	85.7
DAML (tri)	<b>87.1</b>	<b>22.3</b>	<b>58.1</b>	<b>75.0</b>	<b>88.0</b>
Lifted	87.2	25.3	62.6	80.9	91.2
DAML (lifted)	<b>89.1</b>	<b>31.7</b>	<b>66.3</b>	<b>82.8</b>	<b>92.5</b>
N-pair	87.9	27.1	66.4	82.9	92.1
DAML (N-pair)	<b>89.4</b>	<b>32.4</b>	<b>68.4</b>	<b>83.5</b>	<b>92.3</b>

pling [40] and angular loss [40]. For all the baseline methods and DAML, we employed the same GoogLeNet archi-

Figure 4. Visualization of the proposed DAML (N-pair) with Barnes-Hut t-SNE [37] 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.)

Figure 5. Visualization of the proposed DAML (N-pair) with Barnes-Hut t-SNE [37] on the Cars196 dataset, where the color of the border for each image represents the label. (Best viewed on a monitor when zoomed in.)

ture pre-trained on ImageNet for fair comparisons.

### 4.3. Quantitative Results

Table 1-3 show the experimental results of DAML compared with baseline methods on the CUB-200-2011, Cars196 and Stanford Online Product datasets, respectively. In the tables, bold numbers represent that DAML improves the results of the original metric learning algorithms. 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 boosts the performance of original metric learning approaches on all the benchmark datasets. In particular, although the contrastive embedding receives weak supervision where the generator only works on 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 (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 improves 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 for full description of hard negative distributions. As DAML simultaneously trains the hard negative generator and feature embedding in an adversarial manner, the learned distance metric presents strong robustness with adequate and targeted synthetic hard negative samples.

### 4.4. Qualitative Results

Figure 4-6 show the visualization results of DAML (N-pair) using t-SNE [37] on the CUB-200-2011, Cars196 and

Figure 6. Visualization of the proposed DAML (N-pair) with Barnes-Hut t-SNE [37] on the Stanford Online Products dataset, where the color of the border for each image represents the label. (Best viewed on a monitor when zoomed in.)

(a) Pairwise loss (b) Triplet loss

Figure 7. Loss plots of  $J_m$  and  $J_{gen}$  in DAML and different corresponding methods. (Best viewed in color.)

Stanford Online Products datasets, respectively. We enlarge the specific regions to highlight the representative classes at the corner of each figure. The visualization result on the Stanford Online Products dataset is relatively dense as it contains much more images than the other benchmark datasets. We observe that although the images from the same class suffer from large variations such as different backgrounds, colors, poses, viewpoints and configurations, the proposed DAML (N-pair) is still able to group similar objects. The visualization results of the benchmark datasets demonstrate the effectiveness of DAML in an intuitive manner. We also compared the loss plots of DAML as well as the corresponding baselines on the Cars196 dataset as shown in Figure 7. We plotted the average loss for each epoch, and drew the curves of  $J_m$  and  $J_{gen}$  with the parameter 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.

## 5. Conclusion

In this paper, we have proposed a framework of deep adversarial metric learning (DAML), which is generally applicable 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 hard negatives adversarial to the learned metric as important complements of 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 exploit the potential of each negative sample. Experimental results on the CUB-200-2011, Cars196 and Stanford Online Products datasets show that DAML effectively improves the performance of existing deep metric learning methods in an adversarial manner. As DAML focuses on tapping the potential of numerous negative samples, it is an interesting future work to simultaneously generate hard positive samples for data augmentation, so that the quantity gap between negative and positive samples can be reduced.

## Acknowledgement

This work was supported in part by the National Key Research and Development Program of China under Grant 2016YFB1001001, by the National Natural Science Foundation of China under Grant 61672306, Grant U1713214, Grant 61572271, and Grant 61527808, in part by the National 1000 Young Talents Plan Program, in part by the National Basic Research Program of China under Grant 2014CB349304, in part by the Shenzhen Fundamental Research Fund (Subject Arrangement) under Grant JCYJ20170412170602564.



## References

- [1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. J. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Józefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. G. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. A. Tucker, V. Vanhoucke, V. Vasudevan, F. B. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous distributed systems. *arXiv, abs/1603.04467*, 2016. **5**
- [2] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel. InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets. In *NIPS*, pages 2172–2180, 2016. **3**
- [3] 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 *CVPR*, pages 1153–1162, 2016. **1, 2**
- [4] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning. In *ICML*, pages 209–216, 2007. **1, 2**
- [5] Z. Feng, R. Jin, and A. Jain. Large-scale image annotation by efficient and robust kernel metric learning. In *ICCV*, pages 1609–1616, 2013. **2**
- [6] A. Globerson and S. T. Roweis. Metric learning by collapsing classes. In *NIPS*, pages 451–458, 2006. **2**
- [7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*, pages 2672–2680, 2014. **3**
- [8] M. Guillaumin, J. Verbeek, and C. Schmid. Is that you? Metric learning approaches for face identification. In *CVPR*, pages 498–505, 2009. **1**
- [9] R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, pages 1735–1742, 2006. **4, 6**
- [10] B. Harwood, G. Carneiro, I. Reid, and T. Drummond. Smart mining for deep metric learning. In *ICCV*, pages 2821–2829, 2017. **2, 3, 5**
- [11] J. Hu, J. Lu, and Y.-P. Tan. Deep metric learning for visual tracking. *TCSVT*, 26(11):2056–2068, 2016. **1**
- [12] J. Hu, J. Lu, and Y.-P. Tan. Discriminative deep metric learning for face and kinship verification. *TIP*, 26(9):4269–4282, 2017. **1, 2, 4, 6**
- [13] C. Huang, C. C. Loy, and X. Tang. Local similarity-aware deep feature embedding. In *NIPS*, pages 1262–1270, 2016. **2**
- [14] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, pages 1125–1134, 2017. **3**
- [15] J. Krause, M. Stark, J. Deng, and L. Feifei. 3d object representations for fine-grained categorization. In *ICCVW*, 2013. **6**
- [16] B. Kulis. Metric learning: A survey. *Foundations and Trends in Machine Learning*, 5(4):287–364, 2013. **1**
- [17] S. Liao, Y. Hu, X. Zhu, and S. Z. Li. Person re-identification by local maximal occurrence representation and metric learning. In *CVPR*, pages 2197–2206, 2015. **2**
- [18] M.-Y. Liu and O. Tuzel. Coupled generative adversarial networks. In *NIPS*, pages 469–477, 2016. **3**
- [19] Z. Liu, D. Wang, and H. Lu. Stepwise metric promotion for unsupervised video person re-identification. In *ICCV*, pages 2429–2438, 2017. **1**
- [20] J. Lu, G. Wang, and P. Moulin. Image set classification using holistic multiple order statistics features and localized multi-kernel metric learning. In *ICCV*, pages 329–336, 2013. **2**
- [21] I. Masi, A. Tran, J. T. Leksut, T. Hassner, and G. G. Medioni. Do we really need to collect millions of faces for effective face recognition. In *ECCV*, pages 579–596, 2016. **5**
- [22] Y. Movshovitz-Attias, A. Toshev, T. K. Leung, S. Ioffe, and S. Singh. No fuss distance metric learning using proxies. In *ICCV*, pages 360–368, 2017. **1, 2**
- [23] M. Opitz, G. Waltner, H. Possegger, and H. Bischof. Bier - boosting independent embeddings robustly. In *ICCV*, pages 5189–5198, 2017. **2**
- [24] S. Paisitkriangkrai, C. Shen, and A. van den Hengel. Learning to rank in person re-identification with metric ensembles. In *CVPR*, pages 1846–1855, 2015. **2**
- [25] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros. Context encoders: Feature learning by inpainting. In *CVPR*, pages 2536–2544, 2016. **3**
- [26] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, et al. Imagenet large scale visual recognition challenge. *I-JCV*, 115(3):211–252, 2015. **5**
- [27] T. Salimans, I. J. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training GANs. In *NIPS*, pages 2234–2242, 2016. **3**
- [28] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, pages 815–823, 2015. **2, 4**
- [29] M. Schultz and T. Joachims. Learning a distance metric from relative comparisons. In *NIPS*, pages 41–48, 2004. **2**
- [30] S. Shalev-Shwartz, Y. Singer, and A. Y. Ng. Online and batch learning of pseudo-metrics. In *ICML*, 2004. **2**
- [31] A. Shrivastava, A. Gupta, and R. Girshick. Training region-based object detectors with online hard example mining. In *CVPR*, pages 761–769, 2016. **2**
- [32] K. Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *NIPS*, pages 1849–1857, 2016. **1, 5, 6**
- [33] H. O. Song, S. Jegelka, V. Rathod, and K. Murphy. Deep metric learning via facility location. In *CVPR*, pages 5382–5390, 2017. **1, 2, 6**
- [34] H. O. Song, Y. Xiang, S. Jegelka, and S. Savarese. Deep metric learning via lifted structured feature embedding. In *CVPR*, pages 4004–4012, 2016. **1, 2, 5, 6**
- [35] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, pages 1–9, 2015. **5**

- [36] E. Ustinova and V. Lempitsky. Learning deep embeddings with histogram loss. In NIPS, pages 4170–4178, 2016. [2](#)
- [37] L. Van Der Maaten. Accelerating t-sne using tree-based algorithms. JMLR, 15(1):3221–3245, 2014. [7](#), [8](#)
- [38] C. Vondrick and A. Torralba. Generating the future with adversarial transformers. In CVPR, pages 1020–1028, 2017. [3](#)
- [39] C. Wah, S. Branson, P. Welinder, P. Perona, and S. J. Belongie. The Caltech-UCSD Birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. [6](#)
- [40] J. Wang, F. Zhou, S. Wen, X. Liu, and Y. Lin. Deep metric learning with angular loss. In ICCV, pages 2593–2601, 2017. [1](#), [2](#), [6](#)
- [41] X. Wang and A. Gupta. Unsupervised learning of visual representations using videos. In ICCV, pages 2794–2802, 2015. [2](#)
- [42] X. Wang, G. Hua, and T. X. Han. Discriminative tracking by metric learning. In ECCV, pages 200–214, 2010. [1](#)
- [43] X. Wang, A. Shrivastava, and A. Gupta. A-Fast-RCNN: Hard positive generation via adversary for object detection. In CVPR, pages 2606–2615, 2017. [3](#)
- [44] K. Q. Weinberger and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. JMLR, 10(Feb):207–244, 2009. [1](#), [2](#), [4](#), [5](#), [6](#)
- [45] K. Q. Weinberger and G. Tesauro. Metric learning for kernel regression. In AISTATS, pages 612–619, 2007. [2](#)
- [46] C.-Y. Wu, R. Manmatha, A. J. Smola, and P. Krahenbuhl. Sampling matters in deep embedding learning. In ICCV, pages 2840–2848, 2017. [2](#), [3](#)
- [47] F. Xiong, M. Gou, O. Camps, and M. Szaier. Person re-identification using kernel-based metric learning methods. In ECCV, pages 1–16, 2014. [2](#)
- [48] H.-X. Yu, A. Wu, and W.-S. Zheng. Cross-view asymmetric metric learning for unsupervised person re-identification. In ICCV, pages 994–1002, 2017. [1](#)
- [49] Y. Yuan, K. Yang, and C. Zhang. Hard-aware deeply cascaded embedding. In ICCV, pages 814–823, 2017. [2](#), [3](#), [5](#)
- [50] J. Zhou, P. Yu, W. Tang, and Y. Wu. Efficient online local metric adaptation via negative samples for person re-identification. In ICCV, pages 2420–2428, 2017. [1](#)