Deep Meta Metric Learning

Guangyi Chen[1,2,3], Tianren Zhang[1,2,3], Jiwen Lu[1,2,3,*], Jie Zhou[1,2,3]
[1]Department of Automation, Tsinghua University, China
[2]State Key Lab of Intelligent Technologies and Systems, China
[3]Beijing National Research Center for Information Science and Technology, China
{chen-gy16,ztr15}@mails.tsinghua.edu.cn; {lujiwen,jzhou}@tsinghua.edu.cn

Abstract

In this paper, we present a deep meta metric learning (DMML) approach for visual recognition. Unlike most existing deep metric learning methods formulating the learning process by an overall objective, our DMML formulates the metric learning in a meta way, and proves that softmax and triplet loss are consistent in the meta space. Specifically, we sample some subsets from the original training set and learns metrics across different subsets. In each sampled sub-task, we split the training data into a support set as well as a query set, and learn the set-based distance, instead of sample-based one, to verify the query cell from multiple support cells. In addition, we introduce hard sample mining for set-based distance to encourage the intra-class compactness. Experimental results on three visual recognition applications including person re-identification, vehicle re-identification and face verification show that the proposed DMML method outperforms most existing approaches.[1]

1. Introduction

Distance metric learning has been widely used in many visual analysis applications, which aims to learn an embedding space where the distance between similar samples is closer and that of dissimilar samples is farther. Conventional metric learning approaches learn the embedding space by a linear Mahalanobis distance metric [13, 25, 59]. As linear metric learning approaches usually suffer from nonlinear correlations of samples, deep metric learning methods have been proposed to learn discriminative nonlinear embeddings by deep neural networks [36, 45, 57].

One of the most important applications of deep metric learning is visual recognition, which attempts to match a probe sample from the large gallery set, such as person re-identification [4, 7], fine-grained recognition [9, 56, 65], and face recognition [42, 60]. Conventional deep metric learning methods apply the contrastive loss [8, 15] or triplet loss [3, 59] to learn discriminative feature space to measure the similarity of visual samples. Recently, N-pair loss [45] has been presented to take advantage of the whole training batch in each update. However, these methods hardly explain the generalization capacity of metric from training dataset to testing dataset. As shown in Figure 1, we take person re-identification as an example. Assuming that most persons in the training set are wearing colorful T-shirts and similar plain pants while a few of them are wearing vivid pants, global optimization methods (e.g. softmax) tend to identify the persons only from colorful T-shirts and neglect the potential information of pants. For the query samples with similar T-shirts and different pants, they might fail due to over-fitting on the training set. In contrast, without the

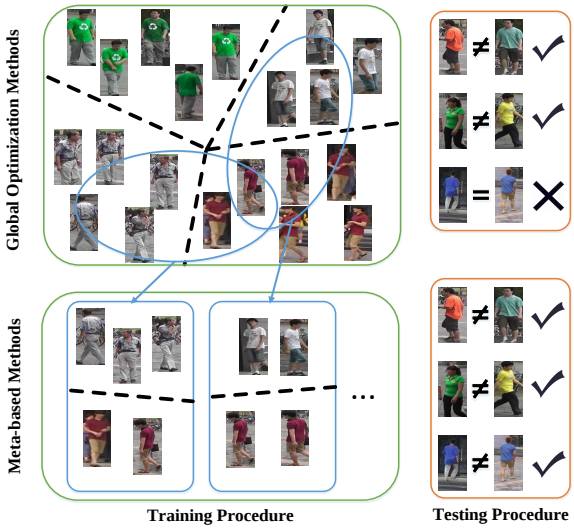

Figure 1. Difference between global optimization methods and meta-based methods. The top part shows that global optimization methods learn classifiers with all training samples, which usually over-fit on the "salient" feature of training data. The bottom part shows that meta-based methods learn the meta metrics across sampled multiple subsets, which mine the potential information for better generalization. (Best viewed in color)

*Corresponding author
[1]Code:https://github.com/CHENGY12/DMML

limit of global objective on the overall training set, the sampled sub-tasks from original task may be propitious to learn the potential transferable information.

In this paper, we propose a deep meta metric learning (DMML) method, which formulates the metric learning process in a meta way and learns set-based distances, instead of sample-based ones. In detail, we sample multiple subsets from the original training set and define a task distribution on these subsets. With the assumption that the unseen test task also satisfies this distribution, we aim to learn a general metric across different subsets, called meta metric, to well transfer to the tasks sampled from the task distribution. Specifically, in each episode, we sample a subset as a new task and split the training data into a support set as well as a query set. We define the support samples in each class as a "meta-cell", and optimize the model to match the query sample with positive meta-cell by set-based distance. In addition, we introduce the hard sample mining process and margin strategy for the proposed set-based distance to explicitly encourage intra-class compactness and inter-class separability. In the experiments, we demonstrate the superiority of our DMML method on some visual recognition problems to baseline deep metric learning and classification methods. To be specific, we improve the performance of vehicle re-identification task on the VeRi-776 [31] dataset compared with both baselines and state-of-the-art methods. We obtain consistent improvement over the performance of person re-identification on the Market-1501 [67] and DukeMTMC-reID [38] datasets as well as face verification accuracies on the Labeled Faces in the Wild (LFW) [23] and YouTube Faces (YTF) [61] databases.

2. Related Work

Metric Learning: Metric learning aims to learn a distance function to measure the similarity of a pair of samples, which gains great success on many visual recognition problems, including person re-identification, face verification, and vehicle re-identification. Early metric learning methods learn a linear Mahalanobis metric for similarity measurement [14,25,59]. For example, LMNN [59] attempts to ensure that the neighbors of each point always belong to the same class, while examples from different classes are separated by a large margin. To learn the nonlinear relationship between samples, kernel tricks are generally adopted in metric learning methods [11,51,63]. More recently, several deep metric learning methods [7, 17, 22, 33, 36, 37, 42, 45, 47, 52, 57] have been proposed to model the nonlinearity of data points, which unify feature learning and metric learning into a joint learning framework. In terms of the input structure in the training procedure, deep metric learning methods can mainly be divided into three categories: contrastive loss [8, 15] with pair-wise inputs, triplet loss [3, 59] with triplet inputs, and N-pair loss [45] with batch inputs. Contrastive loss takes the sample pairs as input and learns to shorten the distances of positive sample pairs and separate those negative samples. Triplet loss preserves the rank relationship with a margin among a triplet of data points. More recently, Sohn [45] addresses the slow convergence problem of conventional triplet loss by pushing away multiple negative examples simultaneously in one batch using a softmax-based objective. Besides, many methods [48, 55] introduce joint identification loss (e.g. softmax loss) in the metric learning framework to increase inter-class separability and reduce intra-class variations. In general, deep metric learning methods achieve great success with strong discriminative power. However, these methods hardly explain the generalization ability of the learned metrics and neglect the relation between intra-class samples. In this paper, we formulate metric learning from a meta perspective, which brings greater interpretability.

Meta Learning: The goal of meta learning is to enable a base learning algorithm to adapt to new tasks efficiently, by extracting some transferable knowledge from a set of auxiliary tasks. For example, several meta learning methods [1, 5, 19] interpret gradient update as a parametric and learnable function rather than a fixed ad-hoc routine. Another promising direction is proposed by MAML [12], which learns initial parameters of the learner for fast adaptation. Some recent works [24, 35, 40] retain the knowledge with memory-augmented models (e.g. the hidden activations of RNN or external memory) and access important and previously unseen information associated with newly encountered tasks. The most related methodologies to ours are Matching Networks [53] and its later developments [41, 44], which learn a set of classifiers with prior tasks and solve the few shot learning problem by weighting these nearest neighbor classifiers. Different from Matching Networks and Prototypical Networks [44], whose goal is mapping few-shot samples into correct classes by their neighbors in the support set, we focus on more general metric learning for visual recognition problems instead of few-shot learning. Additionally, we improve the set-based distance of meta formulation with hard sample mining strategy to accelerate the learning process.

3. Approach

3.1. Deep Meta Metric Learning

Overall Formulation: Most of global optimization learning algorithms optimize an appropriate objective function $\mathcal{L}$ to learn the parameters of deep networks with a single overall observation of training data points,

$$\theta = \arg\min_{\theta} \mathcal{L}(\theta; \mathcal{X}, \mathcal{Y}), \tag{1}$$

where $\mathcal{X}$ represents all training data points and $\mathcal{Y} = \{1, \ldots, N\}$ are corresponding labels.

In our DMML method, instead of considering a single objective with the overall observation of training data, we formulate metric learning in a meta way, which better explains the learning process and generalization ability of the metric. We decompose the single training objective into multiple sub-tasks and learn the meta metric applicable for all sub-tasks. In our assumption, the test task and all sub-tasks are instances sampled from a task distribution $p(\mathcal{T})$.

We formulate the objective function of the proposed DMML method as:

$$\theta = \arg\min_{\theta} \mathbb{E}_{\mathcal{T}_k \sim p(\mathcal{T})} \left[\mathcal{L}_k(\theta; \mathcal{X}_k, \mathcal{Y}_k) \right], \tag{2}$$

where $\mathcal{L}_k(\theta; \mathcal{X}_k, \mathcal{Y}_k)$ denotes the objective function of sampled sub-task $\mathcal{T}_k$. Specifically, for a given N-class training set, we randomly sample $M(M \leq N)$ classes from the original task to construct a new task. Similar to the form of meta learning, we randomly sample a support set $\mathcal{S} = \{s_i^m | i = 1, \dots n_s^m\}$ and a query set $\mathcal{Q} = \{q_i^m | i = 1, \dots n_q^m\}$ for the sub-task $\mathcal{T}_k$, where $m = 1, \dots, M$ denotes the different classes. For simplicity, we set the number of support samples and query samples in different classes equal, i.e. $n_s^m = n_s$ and $n_q^m = n_q$. In each episode, we learn the meta metric to correctly verify the query sample from $\mathcal{Q}$ with support samples in $\mathcal{S}$. The overall formulation of our DMML method is:

$$\theta = \arg\min_{\theta} \mathbb{E}_{\mathcal{T}_k \sim p(\mathcal{T})} \left[\mathbb{E}_{\mathcal{S}, \mathcal{Q} \sim \mathcal{T}_k} \left[\mathcal{L}_k(\theta; \mathcal{Q}, \mathcal{S}) \right] \right]. \tag{3}$$

Learning in One Episode: To learn the meta metric in each episode, we assume all support data points of the same class lie in a manifold, which is defined as "meta-cell":

$$\mathcal{M}^m = \left\{ \sum_{i=1}^{n_s} \alpha_i^m f(s_i^m) \middle| \sum_{i=1}^{n_s} \alpha_i^m = 1, 0 \leq \alpha_i^m \leq 1 \right\}, \tag{4}$$

where the coefficients α_i^m are bounded by $[0, 1]$ to ensure the convexity of the meta-cell, and $f(\cdot)$ represents the embedding function, which is implemented by a deep neural network with parameters θ. Different from conventional metric learning methods which optimize the metrics of sample pairs, we learn the set-based distances which measure the distances between query sample and meta-cells. The set-based metric considers intra-class constraints among the meta-cells to learn discriminative distance metric. Specifically, we define the distance between the query sample and the meta-cell as:

$$d_j^m = D(q_j^{m'}, \mathcal{M}^m) = \sum_{i=1}^{n_s} \alpha_i^m d \left(f(q_j^{m'}), f(s_i^m) \right), \tag{5}$$

where $q_j^{m'}$ is the jth sample in the class m', $\mathcal{M}^m$ denotes the meta-cell with label m, and $d(\cdot, \cdot)$ denotes the distance between a query sample and a support sample.

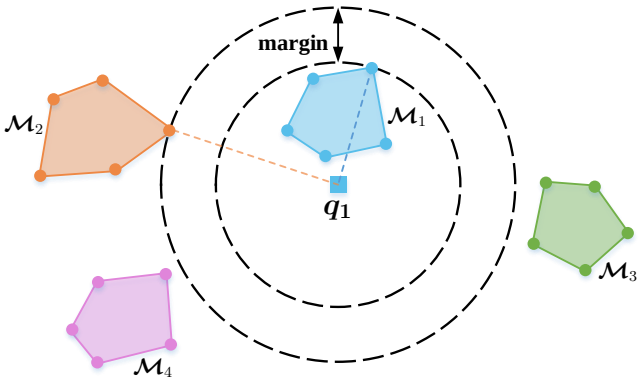

Figure 2. Schematic illustration of margin strategy in the DMML method. For a query sample, we learn a metric that maintains at least a margin between the distance to the positive meta-cell and negative meta-cells. (Best viewed in color)

For each query sample in the query set $\mathcal{Q}$, we optimize the model to minimize its distance to the meta-cell of the same class (i.e. the positive meta-cell) and push away other negative ones. Considering M meta-cells $\{\mathcal{M}^1, \dots, \mathcal{M}^M\}$ and one query sample $q_j^{m'}$, only $\{\mathcal{M}^m, m = m'\}$ is the positive meta-cell while the others are negative meta-cells. To preserve the rank relationship among each triplet of samples, we introduce the conventional triplet loss as follows:

$$\mathcal{L}_{tri}(q_j) = \sum_{m \neq m'} \max\left(0, d_j^{m'} - d_j^m + \tau\right) \tag{6}$$

where $m = m'$ and $m \neq m'$ represent positive and negative sample pairs respectively, and τ is a margin to limit the gap between positive and negative pairs. Then, we apply a continuous exponential function to replace $\max(0, x)$ and use a logarithmic function to limit the range [20], deriving our optimization objective that is equivalent with (6):

$$\begin{aligned} \mathcal{L}_{eps}(q_j) &= \log\left(1 + \sum_{m \neq m'} e^{(d_j^{m'} - d_j^m + \tau)}\right) \\ &= -\log \frac{e^{-d_j^{m'}}}{e^{-d_j^{m'}} + \sum_{m \neq m'} e^{-d_j^m + \tau}}. \end{aligned} \tag{7}$$

In practice, we limit the scale of margin τ with the constraint of $-d_n = \min(-d_j^m + \tau, 0)$, which ensures that the distance is large than zero. As shown in Figure 2, we expect that the distance between the query sample and the positive meta-cell is less than other negative meta-cells by a margin in the embedding space.

Note that (7) is also an approximation of the standard softmax loss, where the similarity between query sample $q_j^{m'}$ and meta-cell $\mathcal{M}^m$ denotes the probability predicting the q_j with label m. It indicates that the classification loss and rank based verification loss are almost identical in the meta space, where meta-cells in DMML serve as alternative classification labels. With this bridge, the effective

Algorithm 1 : DMML

Require: Training image set, the number of classes in the training set N, the number of classes sampled per episode M ($M \leq N$), the number of support instances per class n_s, the number of query instances per class n_q, margin parameter τ, max episode number T.

Ensure: Parameters θ of the embedding function $f(\cdot)$.

1: Initialize θ.
2: **for** $episode = 1, 2, \cdots, T$ **do**
3: Randomly sample M class indices from N.
4: Randomly sample support set and query set for each class.
5: Compute distances between each query sample and all meta-cells using (5).
6: Optimize θ using (7) and (8).
7: **end for**
8: **return** θ.

techniques of metric learning loss can be easily transferred into classification loss, and vice versa. Many softmax-based methods [29, 54] aim to learn classification boundaries with a margin for discriminative learning of the embedding space, which encourages the intra-class compactness and inter-class separability. Here, with the meta space, we naturally propose an additive negative margin softmax loss, which adds margins on the negative samples to optimize the distance metric where differently labeled inputs maintain a large margin of distance and do not threaten to "invade" each other's neighborhoods. Compared with L-softmax [29], A-softmax [28], and AM-softmax [54], our margin-based softmax is more simple and intuitive.

Given the loss function of each episode, we optimize the expectation objective under the task distribution and random splits of support set and query set. The final formulation of the proposed standard DMML method as:

$$\theta = \arg\min_{\theta} \mathbb{E}_{\mathcal{T}_k \sim p(\mathcal{T})} \left[\mathbb{E}_{\mathcal{S},\mathcal{Q} \sim T_k} \sum_{q_j^{m'} \in \mathcal{Q}} \mathcal{L}_{eps}(q_j^{m'}) \right]. \quad (8)$$

For a more clear explanation, we provide Algorithm 1 to detail the procedure of DMML.

Hard Sample Mining: In our DMML method, we propose to use set-based distance to replace sample-based ones for verifying the query cell from multiple support cells, which is formulated in (5). However, this general definition is difficult to optimize directly. In this subsection, we propose two alternative set-based distances: *center support distance* and *hard mining distance*.

The center support distance is a baseline set-based distance which uses the center point of samples in a meta-cell to represent the whole meta-cell and computes the point-to-point distance as the alternative distance [44].The formulation of center support distance is written as:

$$D_c(q_j^{m'}, \mathcal{M}^m) = d\big(f(q_j^{m'}), c^m\big), \quad (9)$$

where the center point of samples is obtained by an average pooling as $c^m = \frac{1}{n_s}\sum_{i=1}^{n_s} f(s_i^m)$. However, in center support distance, hard samples and easy samples are treated equally, which violates the principle of hard sample mining. Therefore, we propose a hard mining distance seeking hard samples in the point-to-set distance, which selects the farthest sample from query samples in each meta-cell to calculate intra-class distance, while selecting the nearest inter-class distance.

In the optimization process of metric learning, hard samples produce substantial gradients with a tiny minority of data. Therefore, hard sample mining of negative examples is considered as an essential component in many metric learning algorithms to improve the convergence speed and verification performance. Conventional hard sample mining algorithms gradually select negative samples that trigger false alarms for bootstrapping. However, negative data mining among different meta-cells is not necessary for DMML, since we have already considered the distances between query samples and all meta-cells in the objective. Instead, we add the hard sample mining process inside the set-based distance to reduce intra-class variances. Specifically, we reformulate the distance between query sample and meta-cell in (5) with hard mining strategy as:

$$D_h(q_j^{m'}, \mathcal{M}^m) = \begin{cases} \max\limits_i \big(d\big(f(q_j^{m'}), f(s_i^m)\big)\big) & m' = m \\ \min\limits_i \big(d\big(f(q_j^{m'}), f(s_i^m)\big)\big) & m' \neq m \end{cases}. \quad (10)$$

The hard sample mining process enhances the discriminative capacity of DMML by seeking the outliers in the meta-cell and punishing them for learning a robust embedding space. As shown in Figure 3, center support distance pushes away all points in the negative meta-cell simultaneously, while hard mining distance gives priority to the hard samples in each meta-cell, which tends to learn a more compact metric. In Section 4.1, we will detailedly discuss and analyze these two distance definitions.

3.2. Implementation Details

We utilized PyTorch to implement our method. We applied squared Euclidean distance $d(\boldsymbol{f}, \boldsymbol{f'}) = \|\boldsymbol{f} - \boldsymbol{f'}\|_2^2$ as the distance metric in (5) and the following equations. To demonstrate the generalization ability of the proposed method for different applications, we fixed all hyper-parameters of DMML in the experiments. In detail, we set the class number of each sub-task and the number of support samples in each meta-cell to $M = 32$ and $n_s = 5$ respectively, and fixed margin parameter $\tau = 0.4$ in our negative-margin-based objective function (7). During training, we

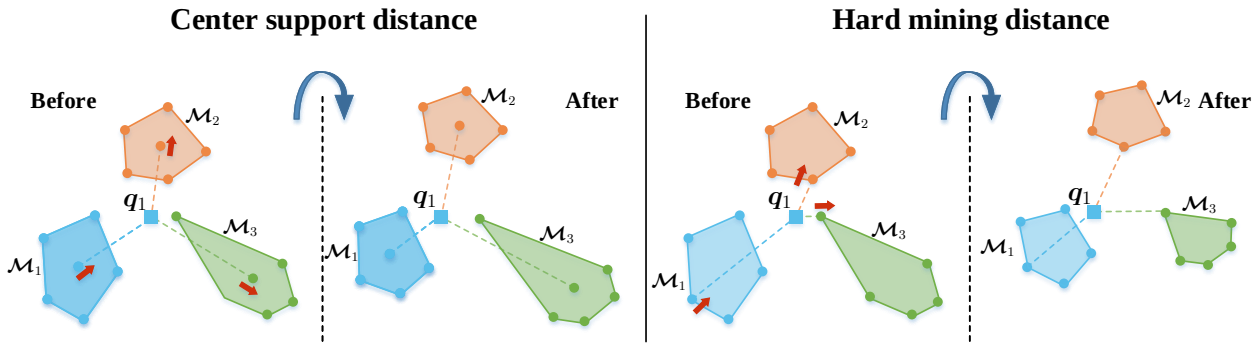

Figure 3. Center support distance and hard mining distance in DMML. **Left**: Center support distance calculates the center point by the mean of all support samples in the meta-cell and computes point-to-point distance between the center point and query sample. **Right**: Hard mining distance adaptively selects the nearest support point from each negative meta-cell and the farthest point from the positive meta-cell.

applied Adam Optimizer and set the base learning rate to 0.0002. The learning rate remained unchanged during the first half of the training stage and then started to decrease exponentially, finally to 0.005 times the base learning rate. Besides, we applied an L2 weight decay of 0.0001. The detailed implementation settings with different specifics of each application are introduced in Section 4.

4. Experiments

In this section, we evaluate the proposed DMML method on three visual recognition tasks: person re-identification, vehicle re-identification, and face verification. Different from the image classification problem which aims to identify query samples into classes emerging in the training procedure, the query samples in visual recognition are unseen for the model. Therefore, how to transfer the trained model to the test dataset without suffering from over-fitting is a bottleneck of visual recognition. We compare our method with abundant baseline approaches and other state-of-the-art methods to demonstrate the effectiveness and high generalization ability of our method. In addition, we conducted ablation experiments and did parameter analysis to investigate the robustness of DMML.

4.1. Person Re-identification

Datasets: Person ReID task aims to identify the pedestrian image of the same identity from a gallery with many negative examples. In our experiments, we applied our approach to two widely used datasets: Market-1501 [67] and DukeMTMC-reID [39]. Market-1501 dataset consists of 32,668 images of 1,501 identities detected by 6 cameras. The whole dataset is divided into a training set with 12,968 images of 751 identities and a test set containing 3,368 query images and 19,732 gallery images of 750 identities. DukeMTMC-reID dataset consists of 36,411 images of 1,404 persons captured by 8 cameras. Its training set includes 16,522 images of 702 persons, and its test set covers the remaining 702 persons, including 2,228 query images and 17,661 gallery images.

Experimental Settings: In person ReID experiments, we employed ResNet-50 [16] as our basic network architecture of the feature representation model, which was pre-trained on ImageNet [10] for rapid convergence. The last spatial down-sampling operation in the network was removed for high resolution. We resized input images to 256×128 and employed random horizontal flip and random erasing [72] for data augmentation. In addition, we introduced part models in the backbone network for further performance improvement. To be specific, we proposed a part-based DMML with an additional part branch after res_conv4_1 residual block, which consists of 3 vertical parts representing different body regions. We supervised the part branch and the basic backbone network with softmax loss and our DMML objective respectively. The input images were resized to 384×128 for enough resolution of the part model. There are two available protocols in the evaluation stage, single-query and multi-query, in terms of the number of images of the dependent query identities. In our experiments, results were all obtained in single-query mode. We applied the cumulative matching characteristic (CMC) curve and mean Average Precision (mAP) as evaluation metrics. CMC curves record the true matching within the top-k ranks, while mAP balances precision and recall to evaluate the overall performance of the method. We followed [67] to compute CMC scores by removing gallery samples with the same camera views as query samples, and then calculating average top-k accuracy over all the queries. We report CMC accuracy of our method at rank-1, rank-5 and rank-10. Moreover, for fairness and conciseness, we did not employ the re-ranking method [71] in our experiments, which could considerably improve the performance of person re-ID methods, especially for mAP.

Comparison with Baseline Methods: We compared our approach with several baseline methods, which include softmax loss [18], contrastive loss [8,15], triplet loss [3,59], as well as more recent N-pair loss [45], lifted structured em-

Table 1. Comparison with state-of-the-art methods on Market-1501 dataset.

Methods	Base model	R-1	R-5	mAP
SVDNet [49]	ResNet-50	82.3	92.3	62.1
PAN [70]	PAN*	82.8	93.5	63.4
DLE [68]	ResNet-50	79.5	-	59.9
TriNet [17]	ResNet-50	84.9	94.2	69.1
CamStyle [73]	ResNet-50	88.1	-	68.7
PoseTransfor [73]	ResNet-50	87.7	-	68.9
DML [66]	MobileNet	89.3	-	70.5
JLML [26]	ResNet-39*	85.1	-	65.5
DPFL [6]	Inception-V3	88.9	-	73.1
MGCAM [46]	ResNet-50	83.8	-	74.3
HA-CNN [27]	HA-CNN*	91.2	-	75.7
AlignedReID [64]	ResNet-50	91.8	97.1	79.3
PCB [50]	ResNet-50	92.3	97.2	77.4
DMML	ResNet-50	92.4	97.3	81.0
DMML+Part	ResNet-50	**93.5**	**97.6**	**81.6**

bedding [36] and proxy-NCA [34]. Softmax loss is widely adopted by many CNNs due to its simplicity and probabilistic interpretation. Contrastive loss is a basic form of conventional deep metric learning methods, which takes the sample pairs as input and learns for verification. Triplet loss additionally learns a large-margin metric which enhances the inter-class separability. Further, in lifted structured embedding, all negative samples in every batch are incorporated against each positive pair. N-pair loss improves conventional metric learning methods by sampling multiple negative instances and calculating a softmax-based loss on similarities. Proxy-NCA introduces trainable similar and dissimilar proxies which approximate the original data points and are optimized during training. Meanwhile, center loss [60] serves as an auxiliary loss to enlarge inter-class distances for face recognition and person ReID tasks. In our experiments, for fair comparisons, we employed the same network architecture for all methods. We make comparisons between our DMML approach and other baselines on two datasets, which are shown in Table 3. Our DMML method beats all baseline methods with a large margin on both rank-1 accuracy and mAP performance, which demonstrates the superiority of DMML compared with other deep metric learning or softmax-based methods. Specifically, we obtained the improvement of 1.2% and 2.6% respectively on rank-1 and mAP performance, in comparison with softmax + center loss and lifted structured embedding methods. Meanwhile, our DMML method outperforms the N-pair loss by 3.0% and 3.6% respectively.

Comparison with State-of-the-art Methods: Table 1 illustrates network architectures, CMC accuracies, and mAP scores of our method and state-of-the-arts on the Market-1501 dataset. The * in the table denotes that the network is individually designed. Methods in the top group are prior works that exploit the global feature of inputs as our basic DMML approach. The bottom group displays the results of works that use part features. As shown in Table 1, our basic DMML method outperforms most of the existing methods. For example, DLE [68], SVDNet [49], and TriNet [17] are the most similar approaches with ours, which learn the embedding of person image without part features and implement the model with ResNet-50. Our DMML obtained a large margin improvement over these methods due to the higher generalization ability from meta-knowledge. Moreover, by combining DMML with the part model, we further boosted the performance of our approach, achieving rank-1/mAP = 93.5%/81.6% on Market-1501. Table 2 summarizes the performance of the proposed method and other state-of-the-arts on the DukeMTMC-reID dataset. Our DMML method and its part-based variant significantly outperform most of existing approaches, achieving rank-1/mAP = 85.9%/73.7%.

Table 2. Comparison with state-of-the-art methods on DukeMTMC-reID dataset.

Methods	Base model	R-1	mAP
GAN [69]	ResNet-50	67.7	47.1
PAN [70]	PAN*	71.6	51.5
SVDNet [49]	ResNet-50	76.7	56.8
DPFL [6]	Inception-V3	79.2	60.6
CamStyle [73]	ResNet-50	75.3	53.5
PoseTransfor [73]	ResNet-50	78.5	56.9
HA-CNN [27]	HA-CNN*	80.5	63.8
PCB [50]	ResNet-50	81.8	66.1
DMML	ResNet-50	84.3	70.2
DMML+Part	ResNet-50	**85.9**	**73.7**

Ablation Study: To verify the effectiveness of components in DMML, we conducted ablation experiments on the Market-1501 dataset for person re-identification. First, to investigate the contribution of the designed hard sample mining method, we compare the performance of the DMML method with center-support distance and hard mining distance. Then, we compare three variants of our method with different margin strategies: no margin in the objective function, additive margin on positive samples [54], and proposed margin strategy with additive margin on negative samples. Table 4 summarizes the performance of the different variants of our DMML method.

1) *Hard Sample Mining:* The performance in Table 4 on center-support distance and hard mining distance demonstrates the significant improvement of the proposed hard sample mining process. By seeking and punishing the outliers in each meta-cell, our method tends to reduce intra-class variances and learn a discriminative feature embedding. Quantitatively, DMML with hard mining distance surpasses the one with center-support distance by a large

Table 3. Comparison with baseline methods on Market-1501, DukeMTMC-reID, VeRi-776, LFW and YTF datasets.

Datasets	**Market-1501**				**DukeMTMC-reID**				**VeRi-776**			**LFW**	**YTF**
Evaluation Metric	**R-1**	**R-5**	**R-10**	**mAP**	**R-1**	**R-5**	**R-10**	**mAP**	**R-1**	**R-5**	**mAP**	**VRF**	**VRF**
Contrastive	75.8	88.6	92.4	58.9	68.1	81.4	85.1	49.5	67.4	85.0	49.8	89.6	83.4
Triplet	89.6	96.2	97.6	76.2	80.7	90.7	93.1	65.4	90.0	95.2	68.1	91.0	84.2
N-pair	89.4	96.1	97.6	77.4	82.0	91.9	94.4	68.3	88.6	95.1	65.1	90.8	84.6
Lifted Struct	90.5	96.8	98.0	78.4	82.6	91.2	93.8	68.0	90.8	96.1	69.3	91.4	**85.6**
Proxy-NCA	88.0	95.4	97.1	71.0	77.9	88.2	91.6	58.1	86.7	93.3	56.4	88.1	81.4
Softmax	86.7	94.5	96.6	70.2	77.0	87.7	91.7	59.6	87.4	94.6	57.8	89.6	82.2
Softmax + Center Loss	91.2	96.5	97.9	77.6	82.3	91.7	93.6	66.3	90.8	95.6	66.0	91.3	84.4
DMML	**92.4**	**97.3**	**98.3**	**81.0**	**84.3**	**92.6**	**94.6**	**70.2**	**91.2**	**96.3**	**70.1**	**91.8**	85.3

Table 4. Results of ablation experiments on hard mining distance and margin strategy on Market-1501 dataset.

Methods	**R-1**	**R-5**	**R-10**	**mAP**
DMML w/o hard mining	87.1	95.6	97.5	70.3
DMML w/o Margin	91.7	97.1	98.1	80.8
DMML +AM [54]	91.9	96.9	98.2	80.7
DMML ($\tau = 0.4$)	**92.4**	**97.3**	**98.3**	**81.0**
DMML ($\tau = 0.2$)	92.0	97.2	98.2	80.6
DMML ($\tau = 0.6$)	91.7	96.9	**98.3**	80.9

margin with 4.3% on rank-1 accuracy and 10.7% on mAP score, respectively.

2) *Margin Strategy:* Compared with the plain DMML without margin parameter, the additive margin on the negative sample promotes 0.7% rank-1 accuracy. It demonstrates the contribution of proposed margin strategy that encourages to inter-class separability with the constraint among the triplet of data points. We evaluated the additive margin on positive samples [54] with the same margin of ours. The results with 0.5% rank-1 improvement show that the proposed positive margin on the negative samples is more effective in comparison with the negative margin on the positive sample.

Parameters Analysis: We also analyzed the influences of some important parameters and demonstrated the robustness of the proposed DMML method. We conducted the parameter analysis experiments on the Market-1501 dataset with three different parameters, including margin scale τ, the number of classes M in each sub-task, and the number of support samples n_s in each meta-cell. The bottom part of Table 4 displays the results with different margin settings, while the performance comparison on different scales of generated sub-tasks and the number of support samples are summarized in the Table 5.

1) *Margin Scale:* The experiments show that DMML is robust on different margin scales. As shown in the bottom part of Table 4, the performance varies smoothly with the change of margin scales. Experimentally, we achieved the best performance when the margin parameter $\tau = 0.4$, thus applied the setting on all experiments. When the margin setting fluctuates, our DMML method still remains comparable with the best setting on both rank-1 and mAP scores.

2) *Number of Classes in Each Sub-task:* As shown in the top group of Table 5, we compared experimental results using 16-class, 32-class, and 64-class sub-tasks with the same support samples in the meta-cell, and obtained improving performance with the increasing scale of sub-tasks. When the number of classes is small, the increase in class number brings relatively larger performance improvement. However, the improvement becomes slow when the scale of sub-tasks is enough to estimate the distribution of tasks. For example, the difference between the performance of 32-class and 64-class sub-tasks is small. We did not conduct experiments on a larger number of classes due to the above observation and limited computing resources.

3) *Number of Support Samples in the Meta-cell:* The bottom part of Table 5 shows the influence of a different number of support samples in each meta-cell. We compare the performance with one support sample, three support samples, and five support samples under the setting of 32-class sub-tasks. From Table 5, we can observe that the performance is improved correspondingly with the increase of support samples. To balance the number of sub-task classes and support samples, we finally set $M = 32$ and $n_s = 5$ respectively in our DMML method for all experiments. All the experiments were conducted with 2 GTX 1080Ti GPUs, except for the settings with $M = 64$ or $n_s = 7$.

4.2. Vehicle Re-identification

Datasets: The goal of vehicle ReID is to retrieve all the images of the same vehicle from a large gallery database. We evaluated our approach on a large-scale dataset: VeRi-776 [30]. This dataset contains over 50,000 images of 776 vehicles, which are captured by 20 surveillance cameras. Vehicles in this dataset cover 9 types and 10 colors, among which 576 are used for training and the rest 200 are used for testing. In total, VeRi-776 dataset consists of 37,778 training images, 1,678 query images, and 11,579 gallery images.

Table 5. Results with different numbers of selected classes M and support samples n_s on Market-1501 dataset.

M	n_s	R-1	R-5	R-10	mAP
16	5	91.0	97.1	98.2	79.5
32	**5**	92.4	97.3	**98.3**	81.0
64	5	92.2	**97.4**	**98.3**	81.2
32	1	90.0	96.2	97.7	76.9
32	3	91.5	97.3	98.2	80.4
32	7	**92.7**	97.1	98.2	**81.6**

Table 6. Comparison with state-of-the-art methods on VeRi-776 dataset.

Methods	Base model	R-1	R-5	mAP
FACT [30]	GoogLeNet	59.7	75.3	19.9
FACT+ST [31]	SNN*	61.4	78.9	27.8
OIFE+ST [58]	CNN*	68.3	89.7	51.4
CNN+LSTM [43]	ResNet-50	83.5	90.0	58.3
VAMI+ST [74]	F-Net*	85.9	91.8	61.3
STP [62]	ResNet-50	86.3	94.4	57.4
RAM [32]	RAM*	88.6	94.0	61.5
DMML	ResNet-50	**91.2**	**96.3**	**70.1**

Experimental Settings: Similar with our person ReID experiment settings, we applied ResNet-50 backbone pretrained on ImageNet for embedding architecture, with the input 224×224 images augmented by random horizontal flip. For fair comparisons, we followed the evaluation protocol in [30], which evaluates the methods with the CMC curve and mAP in the single query mode.

Results: We summarized comparisons between our approach and other baseline methods in Table 3. On this dataset, DMML achieves rank-1 = 91.2% and mAP = 70.1%, outperforming all baselines. Moreover, we also compare the proposed DMML method with other state-of-the-art methods, which additionally demonstrate the effectiveness of our method. As shown in Table 6, DMML surpasses the best prior method [32] by a large margin, both in rank-1 (+2.6%) and mAP (+8.6%).

4.3. Face Verification

Datasets: Face verification task aims to determine whether the given two face images are from the same identity. For this task, we trained our model on an abridged VGGFace2 database [2], and evaluated the verification performance on two other databases: Labeled Faces in the Wild (LFW) [23] and YouTube Faces (YTF) [61]. VGGFace2 database consists of a training set with 3,141,890 images of 8,631 identities, and a test set with 169,396 images of 500 identities. For simplicity, we selected the first 800 identities in the original training set, each with its first 20 images, to construct our new abridged database. This setting with a small scale of samples is more appropriate to evaluate the generalization capacity of methods. Our new training database consists of 16,000 images from 800 identities. LFW database serves as a widely-used benchmark for face verification tasks, which contains 13,233 web-collected images from 5,749 different identities. Images in this database form highly diverse sets of faces, varying in pose, expression, and lighting. YTF database contains 1,595 different people emerging in 3,425 videos downloaded from YouTube, with an average length of 181.3 frames.

Experimental Settings: For face verification task, we applied SE-ResNet-50 [21] as the network architecture, which is pretrained with a classification loss. In the training phase, we employed random gray-scale and random crop as data augmentation methods. For random crop, we first resized the input images to 256×256 and randomly cropped the patch to the size of 224×224. In the testing phase, we took mean verification accuracy (VRF) as our evaluation metric on both LFW and YTF databases. For the verification on LFW, we followed the standard protocol, providing test results of 6,000 face pairs. For YTF, we report the evaluation results on 5,000 face pairs divided into 10 splits.

Results: We compare DMML with other baselines, of which the results are displayed in Table 3. In this experiment, DMML yields comparable performance with the strongest baseline, surpassing the lifted structured embedding approach by 0.4% on VRF performance on the LFW dataset. This result is a favorable evidence to demonstrate the effectiveness of our DMML method.

5. Conclusion

In this work, we have proposed a deep meta metric learning (DMML) approach, which formulates the metric learning in a meta way and optimizes the set-based distance, instead of sample-based one. In our method, we first treat the single overall classification objective as multiple sub-tasks satisfying some unknown probability and randomly split the support and query sets in each sub-task in an episode. Then we learn the meta metric to verify the given query sample from multiple meta-cells in each episode with a margin-based objective function and a hard sample mining strategy. We evaluated our method on three visual recognition problems including person re-identification, vehicle re-identification, and face verification, and outperformed most of the existing methods. In the future, we will explore how to learn meta-knowledge by metric learning from different domains or modes.

Acknowledgements

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFA0700802, in part by the National Natural Science Foundation of China under Grant 61822603, Grant U1813218, Grant U1713214, Grant 61672306, and Grant 61572271.

References

[1] M. Andrychowicz, M. Denil, S. Gomez, M. W. Hoffman, D. Pfau, T. Schaul, B. Shillingford, and N. De Freitas. Learning to learn by gradient descent by gradient descent. In *NeurIPS*, pages 3981–3989, 2016.

[2] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *FG*, pages 67–74, 2018.

[3] G. Chechik, V. Sharma, U. Shalit, and S. Bengio. Large scale online learning of image similarity through ranking. *JMLR*, 11(Mar):1109–1135, 2010.

[4] W. Chen, X. Chen, J. Zhang, and K. Huang. Beyond triplet loss: a deep quadruplet network for person re-identification. In *CVPR*, volume 2, 2017.

[5] Y. Chen, M. W. Hoffman, S. G. Colmenarejo, M. Denil, T. P. Lillicrap, M. Botvinick, and N. de Freitas. Learning to learn without gradient descent by gradient descent. In *ICML*, 2017.

[6] Y. Chen, X. Zhu, and S. Gong. Person re-identification by deep learning multi-scale representations. In *ICCVW*, pages 2590–2600, 2017.

[7] D. Cheng, Y. Gong, S. Zhou, J. Wang, and N. Zheng. Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In *CVPR*, pages 1335–1344, 2016.

[8] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*, volume 1, pages 539–546, 2005.

[9] 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.

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009.

[11] Z. Feng, R. Jin, and A. Jain. Large-scale image annotation by efficient and robust kernel metric learning. In *ICCV*, pages 1609–1616, 2013.

[12] C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, 2017.

[13] A. Globerson and S. T. Roweis. Metric learning by collapsing classes. In *NeurIPS*, pages 451–458, 2006.

[14] M. Guillaumin, J. Verbeek, and C. Schmid. Is that you? metric learning approaches for face identification. In *ICCV*, pages 498–505, 2009.

[15] R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, pages 1735–1742, 2006.

[16] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[17] A. Hermans, L. Beyer, and B. Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017.

[18] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

[19] S. Hochreiter, A. S. Younger, and P. R. Conwell. Learning to learn using gradient descent. In *ICANN*, pages 87–94, 2001.

[20] J. Hu, J. Lu, and Y.-P. Tan. Discriminative deep metric learning for face verification in the wild. In *CVPR*, pages 1875–1882, 2014.

[21] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. *arXiv preprint arXiv:1709.01507*, 7, 2017.

[22] C. Huang, C. C. Loy, and X. Tang. Local similarity-aware deep feature embedding. In *NeurIPS*, pages 1262–1270, 2016.

[23] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. In *ECCVW*, 2008.

[24] Ł. Kaiser, O. Nachum, A. Roy, and S. Bengio. Learning to remember rare events. In *ICLR*, 2018.

[25] M. Koestinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof. Large scale metric learning from equivalence constraints. In *CVPR*, pages 2288–2295, 2012.

[26] W. Li, X. Zhu, and S. Gong. Person re-identification by deep joint learning of multi-loss classification. In *IJCAI*, 2017.

[27] W. Li, X. Zhu, and S. Gong. Harmonious attention network for person re-identification. In *CVPR*, volume 1, page 2, 2018.

[28] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song. Sphereface: Deep hypersphere embedding for face recognition. In *CVPR*, volume 1, page 1, 2017.

[29] W. Liu, Y. Wen, Z. Yu, and M. Yang. Large-margin softmax loss for convolutional neural networks. In *ICML*, pages 507–516, 2016.

[30] X. Liu, W. Liu, H. Ma, and H. Fu. Large-scale vehicle re-identification in urban surveillance videos. In *ICME*, pages 1–6, 2016.

[31] X. Liu, W. Liu, T. Mei, and H. Ma. A deep learning-based approach to progressive vehicle re-identification for urban surveillance. In *ECCV*, pages 869–884, 2016.

[32] X. Liu, S. Zhang, Q. Huang, and W. Gao. Ram: a region-aware deep model for vehicle re-identification. In *ICME*, pages 1–6, 2018.

[33] J. Lu, J. Hu, and Y.-P. Tan. Discriminative deep metric learning for face and kinship verification. *TIP*, 26(9):4269–4282, 2017.

[34] 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.

[35] T. Munkhdalai and H. Yu. Meta networks. In *ICML*, 2017.

[36] H. Oh Song, Y. Xiang, S. Jegelka, and S. Savarese. Deep metric learning via lifted structured feature embedding. In *CVPR*, pages 4004–4012, 2016.

[37] 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.

[38] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *ECCVW*, 2016.

[39] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *ECCV*, pages 17–35, 2016.

[40] A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, and T. Lillicrap. Meta-learning with memory-augmented neural networks. In *ICML*, pages 1842–1850, 2016.

[41] V. G. Satorras and J. B. Estrach. Few-shot learning with graph neural networks. In *NeurIPS*, 2018.

[42] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, pages 815–823, 2015.

[43] Y. Shen, T. Xiao, H. Li, S. Yi, and X. Wang. Learning deep neural networks for vehicle re-id with visual-spatio-temporal path proposals. In *ICCV*, pages 1918–1927, 2017.

[44] J. Snell, K. Swersky, and R. Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, pages 4077–4087, 2017.

[45] K. Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *NeurIPS*, pages 1857–1865, 2016.

[46] C. Song, Y. Huang, W. Ouyang, and L. Wang. Mask-guided contrastive attention model for person re-identification. In *CVPR*, pages 1179–1188, 2018.

[47] H. O. Song, S. Jegelka, V. Rathod, and K. Murphy. Deep metric learning via facility location. In *CVPR*, volume 8, 2017.

[48] Y. Sun, X. Wang, and X. Tang. Deep learning face representation from predicting 10,000 classes. In *CVPR*, pages 1891–1898, 2014.

[49] Y. Sun, L. Zheng, W. Deng, and S. Wang. Svdnet for pedestrian retrieval. In *ICCV*, pages 3820–3828, 2017.

[50] Y. Sun, L. Zheng, Y. Yang, Q. Tian, and S. Wang. Beyond part models: Person retrieval with refined part pooling. In *ECCV*, 2018.

[51] L. Torresani and K.-c. Lee. Large margin component analysis. In *NeurIPS*, pages 1385–1392, 2007.

[52] E. Ustinova and V. Lempitsky. Learning deep embeddings with histogram loss. In *NeurIPS*, pages 4170–4178, 2016.

[53] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra, et al. Matching networks for one shot learning. In *NeurIPS*, pages 3630–3638, 2016.

[54] F. Wang, J. Cheng, W. Liu, and H. Liu. Additive margin softmax for face verification. *IEEE Signal Processing Letters*, 25(7):926–930, 2018.

[55] F. Wang, W. Zuo, L. Lin, D. Zhang, and L. Zhang. Joint learning of single-image and cross-image representations for person re-identification. In *CVPR*, pages 1288–1296, 2016.

[56] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu. Learning fine-grained image similarity with deep ranking. In *CVPR*, pages 1386–1393, 2014.

[57] J. Wang, F. Zhou, S. Wen, X. Liu, and Y. Lin. Deep metric learning with angular loss. In *ICCV*, pages 2612–2620, 2017.

[58] Z. Wang, L. Tang, X. Liu, Z. Yao, S. Yi, J. Shao, J. Yan, S. Wang, H. Li, and X. Wang. Orientation invariant feature embedding and spatial temporal regularization for vehicle re-identification. In *CVPR*, pages 379–387, 2017.

[59] K. Q. Weinberger and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. *JMLR*, 10(Feb):207–244, 2009.

[60] Y. Wen, K. Zhang, Z. Li, and Y. Qiao. A discriminative feature learning approach for deep face recognition. In *ECCV*, pages 499–515, 2016.

[61] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *CVPR*, pages 529–534, 2011.

[62] C.-W. Wu, C.-T. Liu, C.-E. Chiang, W.-C. Tu, S.-Y. Chien, and N. I. Center. Vehicle re-identification with the space-time prior. In *CVPRW*, 2018.

[63] F. Xiong, M. Gou, O. Camps, and M. Sznaier. Person re-identification using kernel-based metric learning methods. In *ECCV*, pages 1–16, 2014.

[64] X. Zhang, H. Luo, X. Fan, W. Xiang, Y. Sun, Q. Xiao, W. Jiang, C. Zhang, and J. Sun. Alignedreid: Surpassing human-level performance in person re-identification. *arXiv preprint arXiv:1711.08184*, 2017.

[65] X. Zhang, F. Zhou, Y. Lin, and S. Zhang. Embedding label structures for fine-grained feature representation. In *CVPR*, pages 1114–1123, 2016.

[66] Y. Zhang, T. Xiang, T. M. Hospedales, and H. Lu. Deep mutual learning. In *CVPR*, 2018.

[67] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. In *ICCV*, 2015.

[68] Z. Zheng, L. Zheng, and Y. Yang. A discriminatively learned cnn embedding for person reidentification. *TOMM*, 14(1):13, 2017.

[69] Z. Zheng, L. Zheng, and Y. Yang. Unlabeled samples generated by gan improve the person re-identification baseline in vitro. In *ICCV*, pages 3774–3782. IEEE, 2017.

[70] Z. Zheng, L. Zheng, and Y. Yang. Pedestrian alignment network for large-scale person re-identification. *TCSVT*, 2018.

[71] Z. Zhong, L. Zheng, D. Cao, and S. Li. Re-ranking person re-identification with k-reciprocal encoding. In *CVPR*, pages 3652–3661, 2017.

[72] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang. Random erasing data augmentation. *arXiv preprint arXiv:1708.04896*, 2017.

[73] Z. Zhong, L. Zheng, Z. Zheng, S. Li, and Y. Yang. Camera style adaptation for person re-identification. In *CVPR*, pages 5157–5166, 2018.

[74] Y. Zhou and L. Shao. Aware attentive multi-view inference for vehicle re-identification. In *CVPR*, pages 6489–6498, 2018.