Deep View-Aware Metric Learning for Person Re-Identification

Pu Chen, Xinyi Xu and **Cheng Deng***
School of Electronic Engineering, Xidian University, Xi'an 710071, China
puchen@stu.xidian.edu.cn, xyxu.xd@gmail.com, chdeng@mail.xidian.edu.cn

Abstract

Person re-identification remains a challenging issue due to the dramatic changes in visual appearance caused by the variations in camera views, human pose, and background clutter. In this paper, we propose a deep view-aware metric learning (DVAML) model, where image pairs with similar and dissimilar views are projected into different feature subspaces, which can discover the intrinsic relevance between image pairs from different aspects. Additionally, we employ multiple metrics to jointly learn feature subspaces on which the relevance between image pairs are explicitly captured and thus greatly promoting the retrieval accuracy. Extensive experiment results on datasets CUHK01, CUHK03, and PRID2011 demonstrate the superiority of our method compared with state-of-the-art approaches.

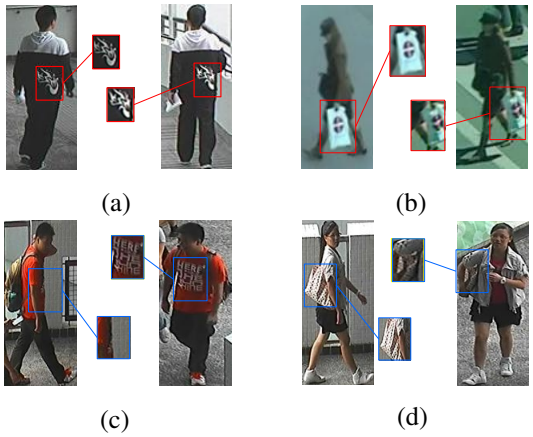

Figure 1: Examples in datasets CUHK03, CUHK01, and PRID2011. (a) and (b) show the image pairs of the same pedestrian in similar views. (c) and (d) show the image pairs in dissimilar views.

1 Introduction

Person re-identification attracts increasing attentions due to its importance in many applications, *e.g.*, video surveillance, pedestrian retrieval, and human-computer interaction. Although significant progress has been made in these years, there are still some challenging problems existing in person re-identification: 1) dramatic changes in visual appearance; 2) dissimilarity between two images of the same pedestrian; 3) similarity between two images of different pedestrians.

To address these issues, typical methods usually focus on extracting robust features or finding a similarity measure. For instance, semantic features from different body regions are captured through a multi-stage ROI pooling pipeline [Zhao *et al.*, 2017]. In [Zhang *et al.*, 2016], a semantics-aware image representation is learned to capture the intrinsic structure information of persons. In [Cheng *et al.*, 2016], an enhanced triplet loss function is proposed to learn a distance measure between two pedestrian images.

However, these approaches are incapable to efficiently discover the intrinsic relevance between image pairs. Fortunately, we find that the intrinsic relevance between image pairs in similar views is different with that in dissimilar views. As shown in Figure 1(a) and (b), the image pairs of the same pedestrian captured from similar views can be connected by some details, such as the patterns of clothing or the carried stuff. As shown in Figure 1(c) and (d), the appearances of pedestrians change dramatically and some details are missed when the image pairs of the same pedestrian are captured from dissimilar views. Even so, the image pairs can also be connected by some view-robust features. Therefore, we can exploit the views as the supplementary information to capture the intrinsic relevance between image pairs.

In this paper, we propose a novel approach called deep view-aware metric learning (DVAML) model, where the image pairs are projected into different feature subspaces according to their view information, which can discover the intrinsic relevance between image pairs from different aspects. Multiple metrics are exploited to supervise the learning of the different feature subspaces. Extensive experiments conducted on three datasets including CUHK01, CUHK03, and PRID2011 show the effectiveness of our method compared with state-of-the-art approaches.

2 Related Work

Typical person re-identification methods usually focus on feature extraction or metric learning. Widely-used features include histograms [Li and Wang, 2013; Khamis *et al.*, 2014;

*Corresponding author

Zhao *et al.*, 2013a], local binary patterns (LBP) [Li and Wang, 2013; Zhao *et al.*, 2013a; Khamis *et al.*, 2014], Gabor features [Li and Wang, 2013] and other cues [Zhang *et al.*, 2014]. However, the changes of appearance lead to instability of these features. The basic idea of metric learning is to find a mapping function from feature space to distance space with certain merits [Cheng *et al.*, 2016], such as Mahalanobis metric learning (KISSME) [Davis *et al.*, 2007], Information-theoretic metric learning (ITML) [Davis *et al.*, 2007], and large margin nearest neighbour (LMNN) [Weinberger and Saul, 2009]. Both lines of the methods regard feature extraction and metric learning processes as two separate steps, which limit the performances significantly.

Inspired by the great success of deep learning networks in computer vision and pattern recognition tasks [Yang *et al.*, 2017b; Deng *et al.*, 2018; Yang *et al.*, 2018; Li *et al.*, 2018; Yang *et al.*, 2017a; Liu *et al.*, 2016; Yang *et al.*, 2017c], many researchers consider the feature and metric learning jointly in an integrated deep architecture, where feature representation can be learned under supervision of the distance metric loss. In [Li *et al.*, 2014], deep learning is exploited to automatically learn features for the re-identification task. And the deep framework has been improved in [Ahmed *et al.*, 2015] by incorporating neighbouring locations of other images. In [Ding *et al.*, 2015], the relative distance between two images are captured by a triplet loss. To further constrain the distances of pairs, [Cheng *et al.*, 2016] propose an enhance triplet loss function. A multi-task deep network (MTDnet) [Chen *et al.*, 2017] are presented to consider the classification loss and the ranking loss simultaneously and takes advantages both of them during training. In order to jointly extract single-image and cross-image feature representations, [Wang *et al.*, 2016] propose a unified triplet and siamese deep architecture.

3 Our Architecture

Fig. 3 shows the framework of the proposed method. The input images of the network are first embedded by the convolution neural network(CNN). Then the output feature maps are respectively projected to different feature subspaces through two full connected layers that do not share parameters. Finally, multiple metric loss functions are used to guide network optimization.

3.1 View-Aware Feature Embedding

Given a training set $\{\mathbf{x}_i, \mathbf{y}_i^1, \mathbf{y}_i^2\}_{i=1,2,...,n}$, where $\mathbf{x}_i$ denotes the ith pedestrian image, $\mathbf{y}_i^1$ and $\mathbf{y}_i^2$ are the identity label and the view label of $\mathbf{x}_i$ respectively. The view label $\mathbf{y}_i^2$ is set as 0 when the pedestrian is in front or back views, while as 1 when the pedestrian is in side views. Furthermore, we denote the feature maps of input images extracted by CNN as $\mathbf{F} = \{f(\mathbf{x}_i)\}_{i=1,2,...,n}$ and combine them into four kinds of pairs as shown on Figure 2:

$$\begin{aligned}
\mathbf{PS} &= \{(f(\mathbf{x}_i), f(\mathbf{x}_j)) \mid \mathbf{y}_i^1 = \mathbf{y}_j^1,\ \mathbf{y}_i^2 = \mathbf{y}_j^2\} \\
\mathbf{PD} &= \{(f(\mathbf{x}_i), f(\mathbf{x}_j)) \mid \mathbf{y}_i^1 = \mathbf{y}_j^1,\ \mathbf{y}_i^2 \neq \mathbf{y}_j^2\} \\
\mathbf{NS} &= \{(f(\mathbf{x}_i), f(\mathbf{x}_j)) \mid \mathbf{y}_i^1 \neq \mathbf{y}_j^1,\ \mathbf{y}_i^2 = \mathbf{y}_j^2\} \\
\mathbf{ND} &= \{(f(\mathbf{x}_i), f(\mathbf{x}_j)) \mid \mathbf{y}_i^1 \neq \mathbf{y}_j^1,\ \mathbf{y}_i^2 \neq \mathbf{y}_j^2\}
\end{aligned} \tag{1}$$

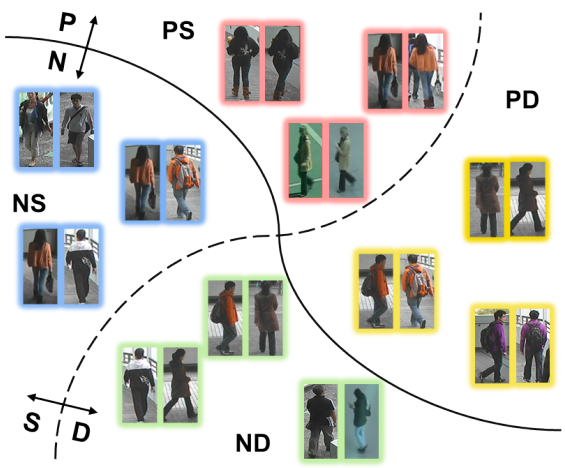

Figure 2: Four sets of image pairs. The active line is the split line between image pairs of same person and of different person, while the broken line is the split line between image pairs in similar views and dissimilar views.

where $1 \leq i, j \leq n$, the feature maps of image pairs $(f(\mathbf{x}_i), f(\mathbf{x}_j)) \in \{\mathbf{PS},\ \mathbf{NS}\}$ are projected to one feature subspace by the full connection layer FC1, while the $(f(\mathbf{x}_k), f(\mathbf{x}_l)) \in \{\mathbf{PD},\ \mathbf{ND}\}$ are projected to another one by FC2. The outputs of FC1 and FC2 are defined as follows:

$$\begin{aligned}
g_1(f(\mathbf{x}_i)) &= \mathbf{W}_1 * f(\mathbf{x}_i) + \mathbf{b}_1 \\
g_2(f(\mathbf{x}_k)) &= \mathbf{W}_2 * f(\mathbf{x}_k) + \mathbf{b}_2
\end{aligned} \tag{2}$$

where $\mathbf{W}_1$ and $\mathbf{W}_2$ are the weights of FC1 and FC2 while b_1 and b_2 represent the biases. $*$ refers to the convolution operation.

The CNN is built inspired by [Xiao *et al.*, 2016; Simonyan and Zisserman, 2014] and details of the structure are listed in Table 1. Each ReLU layer is followed by a Batch Normalization (BN) layer which accelerates the convergence process and avoids manually tweaking the initialization of weights and biases. For training the CNN from scratch, we randomly dropout 50% neurons of the fc7_1 and fc7_2 layers.

Besides, We learn a view classifier to choose the corresponding feature subspace which the feature maps of images are projected into. The view classifier is optimized by minimizing the sigmoid loss. The first and second layers of the view classifier are convolutional layers with the kernel size 32×7×7 and 32×5×5 respectively. The kernels size of third convolutional layer is 64×3×3 and all the three layers are followed by a pooling layer. Then two fully connected layers that contain 512 neurons receive the output of the third convolutional layer.

3.2 Multiple Metric Loss

Different with other methods, we utilize two metric loss functions to constrain the distances between image pairs in similar views and in dissimilar views separately. Figure 4 illustrates the failure case of other methods that constrain the distances between all image pairs with one loss. In Figure 4(a), the loss decreases the distances between $(\mathbf{x}_1, \mathbf{x}_2)$ and between

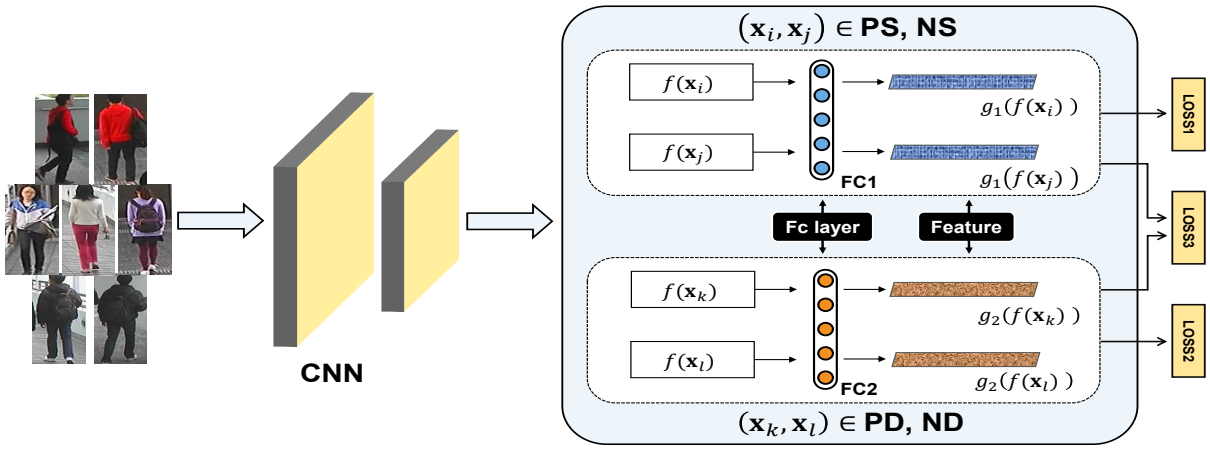

Figure 3: Architecture of the proposed deep view-aware metric learning (DVAML) model. The input of the network is a set of images. Then the feature maps output by the CNN are respectively projected to different feature subspaces through two full connected layers. Finally, three loss functions are used to guide network optimization.

$(\mathbf{x}_1, \mathbf{x}_3)$ by giving $\mathbf{x}_1$ a upwards and a downwards pulling force respectively. Simultaneously, $\mathbf{x}_1$ is given a left and a right pushing force to increase the distances between $(\mathbf{x}_1, \mathbf{x}_4)$ and between $(\mathbf{x}_1, \mathbf{x}_5)$. Therefore, $\mathbf{x}_1$ will still in place. In Figure 4(b), image pairs in similar views and in dissimilar views are first projected into two different feature subspaces as introduced in 3.1 section. Then L^1 and L^2 are used to supervise the learning of the different feature subspaces respectively. L^1 decreases the distance between $(\mathbf{x}_1, \mathbf{x}_2)$ by pulling $\mathbf{x}_1$ upwards and increases the distance between $(\mathbf{x}_1, \mathbf{x}_5)$ by pushing $\mathbf{x}_1$ right, while L^2 decreases the distance between $(\mathbf{x}_1, \mathbf{x}_3)$ by pulling $\mathbf{x}_1$ downwards and increases the distance between $(\mathbf{x}_1, \mathbf{x}_4)$ by pushing $\mathbf{x}_1$ left. $\mathbf{x}_1$ moves finally towards the correct direction in both feature subspaces.

The loss function L^1 based on the lifted structured feature embedding[Song *et al.*, 2016] is defined as:

$$L^1_{i,j} = \log\Bigg[\sum_{(i,k)\in\mathbf{NS}} \exp\{\beta_1 - D_{i,k}\} + \sum_{(j,l)\in\mathbf{NS}} \exp\{\beta_1 - D_{j,l}\}\Bigg] + D_{i,j} \tag{3}$$

$$L^1 = \frac{1}{2|\mathbf{PS}|} \sum_{(i,j)\in\mathbf{PS}} \max(0, L^1_{i,j})^2 \tag{4}$$

where $L^1_{i,j}$ is used to limit the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ in $\mathbf{PS}$ less than all pairs in $\mathbf{NS}$ which contain $\mathbf{x}_i$ or $\mathbf{x}_j$ with a margin β_1.

Correspondingly, the loss function L^2 is defined as:

$$L^2_{i,j} = \log\Bigg[\sum_{(i,k)\in\mathbf{ND}} \exp\{\beta_2 - D_{i,k}\} + \sum_{(j,l)\in\mathbf{ND}} \exp\{\beta_2 - D_{j,l}\}\Bigg] + D_{i,j} \tag{5}$$

$$L^2 = \frac{1}{2|\mathbf{PD}|} \sum_{(i,j)\in\mathbf{PD}} \max(0, L^2_{i,j})^2 \tag{6}$$

where $L^2_{i,j}$ is used to limit the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ in $\mathbf{PD}$ less than all pairs in $\mathbf{ND}$ which contain $\mathbf{x}_i$ or $\mathbf{x}_j$ with a margin β_2. The value $|\mathbf{PS}|$ and $|\mathbf{PD}|$ are respectively equal to the number of image pairs belong to $\mathbf{PS}$ and $\mathbf{PD}$ in each batch.

Noted that the distances between images pairs in similar and dissimilar views are respectively denoted as:

$$D_{i,j} = \begin{cases} ||g_1(f(\mathbf{x}_i)) - g_1(f(\mathbf{x}_j))||_2 & (i,j) \in \mathbf{S} \\ ||g_2(f(\mathbf{x}_i)) - g_2(f(\mathbf{x}_j))||_2 & (i,j) \in \mathbf{D} \end{cases} \tag{7}$$

in which $\mathbf{S}$ denotes the set of image pairs in similar views including $\mathbf{PS}$ and $\mathbf{NS}$, while $\mathbf{D}$ denotes the set of image pairs with dissimilar views consisting of $\mathbf{PD}$ and $\mathbf{ND}$.

Considering that the loss function L^1 and L^2 only constrains the distances between image pairs in similar views and dissimilar views separately. So we use L^3 constrain the distances between image pairs of same person and between image pairs of different person overall. The loss function L^3 is defined as:

$$L^3_{i,j} = \log\Bigg[\sum_{(i,k)\in\mathbf{N}} \exp\{\beta_3 - D_{i,k}\} + \sum_{(j,l)\in\mathbf{N}} \exp\{\beta_3 - D_{j,l}\}\Bigg] + D_{i,j} \tag{8}$$

$$L^3 = \frac{1}{2|\mathbf{PS}| + |\mathbf{PD}|} \sum_{(i,j)\in\mathbf{P}} \max(0, L^3_{i,j})^2 \tag{9}$$

in which $\mathbf{P}$ denotes the set of the image pairs of same identity consist of $\mathbf{PS}$ and $\mathbf{PD}$. $\mathbf{N}$ constrains all the image pairs in $\mathbf{NS}$ and $\mathbf{ND}$.

We finally unite the three loss functions to a joint loss function:

$$L = \alpha_1 L^1 + \alpha_2 L^2 + \alpha_3 L^3 \tag{10}$$

where the three hyper-parameters α_1, α_2 and α_3 are used to balance the three loss functions.

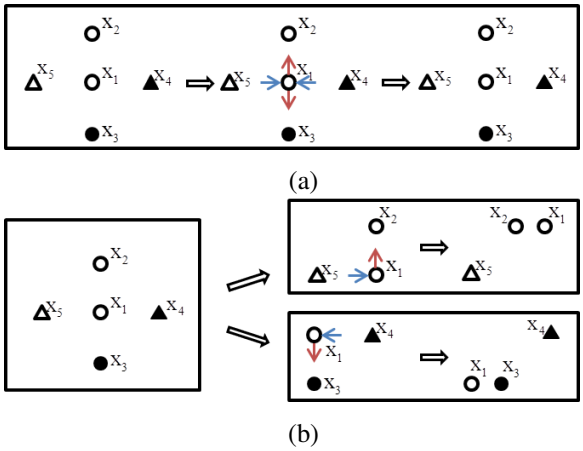

Figure 4: The rounds $\mathbf{x}_1$, $\mathbf{x}_2$ and $\mathbf{x}_3$ are images of person A, while the triangles $\mathbf{x}_4$ and $\mathbf{x}_5$ are images of person B. The solid and hollow denote the two different views of the person respectively. The red arrowhead means the pulling force and the blue arrowhead means the push force.

3.3 Optimization

The network is jointly supervised and optimized by the joint loss. The back propagation gradient with respect to the distance $D_{i,j}$ is:

$$\frac{\partial L}{\partial D_{i,j}} = \frac{\partial L^1}{\partial D_{i,j}}\mathbf{1}[(i,j) \in \mathbf{PS}] + \frac{\partial L^2}{\partial D_{i,j}}\mathbf{1}[(i,j) \in \mathbf{PD}] + \frac{\partial L^3}{\partial D_{i,j}} \tag{11}$$

where $\mathbf{1}$ is the indicator function which outputs 1 if the expression evaluates to true and outputs 0 otherwise. The gradients of the distances between image pairs in $\mathbf{PS}$ consist of two components coming from L^1 and L^3, while that of image pairs in $\mathbf{PD}$ coming from L^2 and L^3.

$$\frac{\partial L^1}{\partial D_{i,j}} = \frac{1}{|\mathbf{PS}|}L^1_{i,j}\mathbf{1}[L^1_{i,j} > 0] \tag{12}$$

$$\frac{\partial L^2}{\partial D_{i,j}} = \frac{1}{|\mathbf{PD}|}L^2_{i,j}\mathbf{1}[L^2_{i,j} > 0] \tag{13}$$

$$\frac{\partial L^3}{\partial D_{i,j}} = \frac{1}{|\mathbf{PS}| + |\mathbf{PD}|}L^3_{i,j}\mathbf{1}[L^3_{i,j} > 0] \tag{14}$$

The gradient with respect to the distance $D_{i,k}$ is:

$$\frac{\partial L}{\partial D_{i,k}} = \frac{\partial L^1}{\partial D_{i,k}}\mathbf{1}[(i,k) \in \mathbf{NS}] + \frac{\partial L^2}{\partial D_{i,k}}\mathbf{1}[(i,k) \in \mathbf{ND}] + \frac{\partial L^3}{\partial D_{i,k}} \tag{15}$$

in which the gradients of the three loss functions with respect to $D_{i,k}$ are:

$$\frac{\partial L^1}{\partial D_{i,k}} = \frac{L^1_{i,j}\mathbf{1}[L^1_{i,j} > 0]}{|\mathbf{PS}|}\sum_{(i,j)\in\mathbf{PS}}\frac{-\exp\{\beta_1 - D_{i,k}\}}{\exp\{L^1_{i,j} - D_{i,j}\}} \tag{16}$$

Algorithm 1 BackPropagation gradient

Input: $D, \alpha_1, \alpha_2, \alpha_3, \beta_1, \beta_2, \beta_3$
Output: $\frac{\partial L}{\partial f(\mathbf{x}_i)}, \quad \forall i \in [1, n]$

Initialize $\frac{\partial L}{\partial f(\mathbf{x}_i)} = 0, \quad \forall i \in [1, n]$
for $i = 1, .., n$ **do**
  for $j = i + 1, .., n, s.t.(i, j) \in \mathbf{P}$ **do**
    $\frac{\partial L}{\partial f(\mathbf{x}_i)} \leftarrow \frac{\partial L}{\partial f(\mathbf{x}_i)} + \frac{\partial L}{\partial D_{i,j}}\frac{\partial D_{i,j}}{\partial f(\mathbf{x}_i)}$;
    $\frac{\partial L}{\partial f(\mathbf{x}_j)} \leftarrow \frac{\partial L}{\partial f(\mathbf{x}_j)} + \frac{\partial L}{\partial D_{i,j}}\frac{\partial D_{i,j}}{\partial f(\mathbf{x}_j)}$;
  end for
  for $k = 1, .., n, s.t.(i, k) \in \mathbf{N}$ **do**
    $\frac{\partial L}{\partial f(\mathbf{x}_i)} \leftarrow \frac{\partial L}{\partial f(\mathbf{x}_i)} + \frac{\partial L}{\partial D_{i,k}}\frac{\partial D_{i,k}}{\partial f(\mathbf{x}_i)}$;
    $\frac{\partial L}{\partial f(\mathbf{x}_k)} \leftarrow \frac{\partial L}{\partial f(\mathbf{x}_k)} + \frac{\partial L}{\partial D_{i,k}}\frac{\partial D_{i,k}}{\partial f(\mathbf{x}_k)}$;
  end for
end for

$$\frac{\partial L^2}{\partial D_{i,k}} = \frac{L^2_{i,j}\mathbf{1}[L^2_{i,j} > 0]}{|\mathbf{PD}|}\sum_{(i,j)\in\mathbf{PD}}\frac{-\exp\{\beta_2 - D_{i,k}\}}{\exp\{L^2_{i,j} - D_{i,j}\}} \tag{17}$$

$$\frac{\partial L^3}{\partial D_{i,k}} = \frac{L^3_{i,j}\mathbf{1}[L^3_{i,j} > 0]}{|\mathbf{PS}| + |\mathbf{PD}|}\sum_{(i,j)\in\mathbf{P}}\frac{-\exp\{\beta_3 - D_{i,k}\}}{\exp\{L^3_{i,j} - D_{i,j}\}} \tag{18}$$

The gradient with respect to $D_{j,l}$ is:

$$\frac{\partial L}{\partial D_{j,l}} = \frac{\partial L^1}{\partial D_{j,l}}\mathbf{1}[(j,l) \in \mathbf{NS}] + \frac{\partial L^2}{\partial D_{j,l}}\mathbf{1}[(j,l) \in \mathbf{ND}] + \frac{\partial L^3}{\partial D_{j,l}} \tag{19}$$

The gradients of the distances between image pairs in similar views with respect to the feature maps of them are:

$$\frac{\partial D_{i,j}}{\partial f(\mathbf{x}_i)} = \frac{1}{D_{i,j}}(g_1(f(\mathbf{x}_i)) - g_1(f(\mathbf{x}_j)))\frac{\partial g_1(f(\mathbf{x}_i))}{\partial f(\mathbf{x}_i)} \tag{20}$$

$$\frac{\partial D_{i,j}}{\partial f(\mathbf{x}_j)} = \frac{1}{D_{i,j}}(g_1(f(\mathbf{x}_j)) - g_1(f(\mathbf{x}_i)))\frac{\partial g_1(f(\mathbf{x}_j))}{\partial f(\mathbf{x}_j)} \tag{21}$$

Correspondingly, the gradients of the distances between image pairs in dissimilar views with respect to the feature maps of them are:

$$\frac{\partial D_{i,j}}{\partial f(\mathbf{x}_i)} = \frac{1}{D_{i,j}}(g_2(f(\mathbf{x}_i)) - g_2(f(\mathbf{x}_j)))\frac{\partial g_2(f(\mathbf{x}_i))}{\partial f(\mathbf{x}_i)} \tag{22}$$

$$\frac{\partial D_{i,j}}{\partial f(\mathbf{x}_j)} = \frac{1}{D_{i,j}}(g_2(f(\mathbf{x}_j)) - g_2(f(\mathbf{x}_i)))\frac{\partial g_2(f(\mathbf{x}_j))}{\partial f(\mathbf{x}_j)} \tag{23}$$

4 Experiments

4.1 Datasets and Settings

Datasets: We evaluate our method on three datasets, CUHK03 [Li *et al.*, 2014], CUHK01 [Li *et al.*, 2012] and PRID2011 [Hirzer *et al.*, 2011]. CUHK03 dataset contains 13164 images from 1360 persons. We select 1160 persons

Type	patch size/ stride	output size	# 1 × 1	# 3 ×3 reduce	#1 × 1	double # 3 × 3 reduce	double # 3 × 3	pool + proj
input		3 × 128 × 64						
conv1-conv2	3 × 3/2	32 × 128 × 64						
conv3	3 × 3/2	64 × 128 × 64						
pool3	2 × 2/2	64 × 64 × 32						
inception (4a)		256 × 64 × 32	64	64	64	64	64	avg + 64
inception (4b)		384 × 32 × 16	64	64	64	64	64	max + pass through
inception (5a)		512 × 32 × 16	128	128	128	128	128	avg + 128
inception (5b)		768 × 16 × 8	128	128	128	128	128	max + pass through
inception (6a)		1024 × 16 × 8	256	256	256	256	256	avg + 256
inception (6b)		1536 × 8 × 4	256	256	256	256	256	max + pass through
avg pool	8 × 4/1	1536 × 1 × 1						
fc7_1		512 × 1 × 1						
fc7_2		512 × 1 × 1						

Table 1: The network details of CNN in the proposed method for person re-identification

Method	Top1	Top5	Top10	Top20
KISSME	14.17	37.46	52.20	69.32
ITML	5.53	18.89	28.96	44.33
LMNN	21.17	49.49	61.12	78.32
eSDC	8.76	24.07	39.28	50.91
FPNN	20.65	51.00	67.00	83.31
kLFDA	48.20	69.01	79.63	89.18
IDLA	54.74	86.50	94.00	98.00
GatedSiamese	68.10	88.10	94.60	-
MTDnet	74.68	95.99	97.47	-
Ours	**76.11**	**96.01**	**98.90**	**99.87**

Table 2: Matching rates (%) on CUHK03 dataset

Method	Top1	Top5	Top10	Top20
KISSME	29.40	59.34	71.45	88.12
ITML	17.10	42.31	55.07	71.65
LMNN	21.17	49.49	61.12	78.32
eSDC	22.84	43.89	57.67	69.84
FPNN	27.87	58.20	73.46	86.31
kLFDA	42.76	69.01	79.63	89.18
IDLA	65.00	89.33	93.00	96.51
ImpTrpLoss	53.70	84.30	91.00	96.30
P2S	77.34	93.51	96.73	98.53
MTDnet-cross	78.50	**96.50**	97.50	-
Ours	**80.54**	95.00	**97.99**	**99.38**

Table 3: Matching rates (%) on CUHK01 dataset

Method	Top1	Top5	Top10	Top20
KISSME	28.54	59.78	72.13	83.26
LMNN	14.38	38.09	50.22	67.19
kLFDA	22.10	46.60	58.10	-
LADF	8.20	20.45	29.89	42.25
ImpTrpLoss	22.00	-	47.00	57.00
MDTnet-cross	31.00	54.00	61.00	-
Ours	**33.00**	**65.00**	**75.00**	**83.00**

Table 4: Matching rates (%) on PRID-2011 dataset

for training, 100 for validation and 100 for testing following the same setting as [Li *et al.*, 2014] and [Ahmed *et al.*, 2015]. CUHK01 [Li *et al.*, 2012] dataset contains 971 persons captured from two camera views in a campus. We utilize the same protocol used in [Zhou *et al.*, 2017] and [Wang *et al.*, 2016], where 871 person images are used for training and the left for testing. Following the setting in [Chen *et al.*, 2017], 100 persons in PRID2011 dataset are used for training, specially, 100 for probe and 649 (the remaining persons from camera B except the 100 for training) for gallery in test set.

For training the view classifier, we manually annotate the coarse view labels for 200 persons in CUHK03 and CUHK01. Considering the size and complexity of PRID2011 dataset, 50 persons in it are annotated. Since the view classification is treated as a simple binary classification problem, the view classifier achieves a high accuracy rate even though training samples are limited.

Considering that multiple cameras in varied views may be retrieved for a given query image in practical application, we fuse the three datasets two by two into three new datasets: Syn1 (CUHK03 and CUHK01), Syn2 (CUHK03 and PRID2011) and Syn3 (CUHK01 and PRID2011). Then we train the model and retrieve probe of the one on the two datasets for each fused. For example, the same 1260 persons in CUHK03 and 871 persons in CUHK01 are chosen as Syn1's training set. For the probe of CUHK03, there are another 100 persons from CUHK01 in the CUHK03's gallery besides the original 100. Equally, there are 200 persons in the CUHK01's gallery. Noted that 749 persons are used as the gallery of PRID2011 in Syn2 and Syn3, while only 200 persons as the CUHK01's and CUHK03's gallery.

Parameters Implementation. All the images are resized to $128 * 64$ before being fed to the network and the batchsize of the input is 64. For identity classifier, we first pretrain a model on CUHK03 and the learning rate is set to 0.001, then finetune this model with learning rate 0.002. We picked a set of optimal loss weights $\alpha_1 = 0.4$, $\alpha_2 = 0.4$, $\alpha_3 = 0.2$ experimentally. And all the margin parameters β_1, β_2, β_3 are set to 1 [Song *et al.*, 2016].

Evaluation Protocol: We report the single-shot results on

Method	CUHK03			CUHK01		
	Top1	Top5	Top10	Top1	Top5	Top10
KISSME	11.03	32.21	49.82	21.32	52.17	68.44
LMNN	18.33	46.31	57.52	17.52	45.32	56.31
kLFDA	43.17	67.32	72.63	37.65	65.32	74.33
MTDnet	67.00	68.00	92.00	70.00	91.00	96.00
Ours	**72.77**	**94.83**	**98.00**	**73.00**	**93.00**	**97.00**

Table 5: Matching rates (%) on Syn1 dataset

Method	CUHK03			PRID2011		
	Top1	Top5	Top10	Top1	Top5	Top10
KISSME	13.56	32.66	48.63	24.31	48.41	68.13
LMNN	19.54	47.26	60.21	12.98	35.67	46.85
kLFDA	45.31	65.33	77.39	19.69	43.74	56.31
MTDnet	72.00	91.00	94.00	21.00	41.00	52.00
Ours	**75.02**	**94.51**	**96.36**	**35.00**	**61.00**	**70.00**

Table 6: Matching rates (%) on Syn2 dataset

Method	CUHK01			PRID2011		
	Top1	Top5	Top10	Top1	Top5	Top10
KISSME	14.57	35.61	52.63	26.31	51.41	65.13
LMNN	20.61	45.26	62.33	14.92	37.62	44.80
kLFDA	47.21	68.35	74.32	18.61	45.72	55.31
MTDnet	73.00	91.00	93.00	24.00	39.00	47.00
Ours	**76.00**	**91.27**	**94.00**	**31.00**	**65.00**	**77.00**

Table 7: Matching rates (%) on Syn3 dataset

the three datasets and the synthetic dataset of them. The datasets are separated into the training set and the testing set, and the testing set is further divided into probe set and gallery set that contain different images of the same person. The result is evaluated by cumulative matching characteristic (CMC) curve, which is an estimation of finding the corrected match in the top n match.

4.2 Results and Analysis

We compare our method with 7 common methods including KISSME, ITML, LMNN, eSDC [Zhao *et al.*, 2013b], kLFDA [Xiong *et al.*, 2014], FPNN [Li *et al.*, 2014], IDLA [Ahmed *et al.*, 2015]. And Table 2-7 show the evaluation results of these method.

Results on CUHK03. In Table 2, we also compare our method with two deep methods GatedSiamese [Varior *et al.*, 2016] and MTDnet [Chen *et al.*, 2017] except the 7 methods. From Table1, most of deep learning methods except FPNN obviously outperform the traditional methods. Ours model has achieved the top performances by 76.11% on Top1 accuracy (vs. 74.68% by the next best method).

Results on CUHK01. Table 3 extra lists the results of the P2S [Zhou *et al.*, 2017] and ImpTrpLoss [Cheng *et al.*, 2016] methods. From Table 3, we can see that our method outperforms the state-of-the-art methods by more than 80% on Top1 accuracy (80.54% vs. 78.50% by MTDnet-cross). Note that the Top 5 recognition rate of our method reaches 95.00%, meaning that the trained model has high probability of finding the correct person from other cameras.

Results on PRID2011. We can see from Table 4 that our method achieves 33% on Top 1 matching rate and outperforms the previous best accuracy on PRID2011 even though the training samples are limited for deep neural networks. It is noted that the Top 5 matching rate achieves 65% far beyond the accuracy of other methods.

Results on Syn1, Syn2 and Syn3. Table 5-7 illustrates the performance of ours method and the previous methods on synthetic datasets. And it's obvious that our method has achieved the top performance on all the three datasets. From Table 5, we can see that the matching rate of our method on CUHK03 and CUHK01 reaches 72.77% and 73.00%. It's worth noting that the accuracy on Syn1 declines contrast to each single dataset due to the gallery images from the other dataset can interfere the retrieval of the probe. The results on Syn2 are listed in Table 6 and our method's accuracy achieves 75.02% and 35.00% on CUHK03 and PRID2011 respectively. The accuracy of PRID2011 when trained on Syn2 dataset improves 2% compared with on single dataset, since the added training samples may improve the performance of model on small dataset. The accuracy of CUHK01 and PRID2011 listed in Table 7 reaches 76.00% and 31.00%. It is noted that we improve the performance on PRID2011 by more than 20% on the rank-5 accuracy in contrast to the previous methods.

Effect of deep view-aware metric. We employ an view-aware method to project image pairs into different feature subspaces. The method has positive effect on person re-identification since it captures the intrinsic relevance between image pairs. Besides, the proposed network is constrained by three metric loss functions, of which the two supervise separately the learning of the different feature subspaces and the last supervises the whole network. And our model wins over 3% compared to other methods on the three synthetic datasets, which further proves the effectiveness of it.

5 Conclusion

In this paper, we propose a deep view-aware metric learning (DVAML) model for person re-identification. We project image pairs in similar and dissimilar views into different feature subspaces, respectively. Then multiple loss functions are used to supervise the learning of the different feature subspaces. In addition, we also consider the actual situation that a given probe image can be retrieved from multiple cameras by synthesizing different datasets. Experiment results on CUHK03, CUHK01, PRID2011 and three synthetic datasets illustrate that our method outperforms the state-of-the-art approaches.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 61572388 and Grant 61703327, and in part by the Key R&D Program. The Key Industry Innovation Chain of Shaanxi under Grant 2017ZDCXL-GY-05-04-02 and Grant 2017ZDCXL-GY-05-04-02.

References

[Ahmed *et al.*, 2015] Ejaz Ahmed, Michael Jones, and Tim K. Marks. An improved deep learning architecture for person re-identification. In *CVPR*, 2015.

[Chen *et al.*, 2017] Weihua Chen, Xiaotang Chen, Jianguo Zhang, and Kaiqi Huang. A multi-task deep network for person re-identification. In *AAAI*, 2017.

[Cheng *et al.*, 2016] De Cheng, Yihong Gong, Sanping Zhou, Jinjun Wang, and Nanning Zheng. Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In *CVPR*, 2016.

[Davis *et al.*, 2007] Jason V. Davis, Brian Kulis, Prateek Jain, Suvrit Sra, and Inderjit S. Dhillon. Information-theoretic metric learning. In *ICML*, 2007.

[Deng *et al.*, 2018] Cheng Deng, Zhaojia Chen, Xianglong Liu, Xinbo Gao, and Dacheng Tao. Triplet-based deep hashing network for cross-modal retrieval. *IEEE Transactions on Image Processing*, 2018.

[Ding *et al.*, 2015] Shengyong Ding, Liang Lin, Guangrun Wang, and Hongyang Chao. Deep feature learning with relative distance comparison for person re-identification. *Pattern Recognition*, 2015.

[Hirzer *et al.*, 2011] Martin Hirzer, Csaba Beleznai, Peter M Roth, and Horst Bischof. Person re-identification by descriptive and discriminative classification. In *Scandinavian conference on Image analysis*. Springer, 2011.

[Khamis *et al.*, 2014] Sameh Khamis, Cheng-Hao Kuo, Vivek K Singh, Vinay D Shet, and Larry S Davis. Joint learning for attribute-consistent person re-identification. In *ECCV*, 2014.

[Li and Wang, 2013] Wei Li and Xiaogang Wang. Locally aligned feature transforms across views. In *CVPR*, 2013.

[Li *et al.*, 2012] Wei Li, Rui Zhao, and Xiaogang Wang. Human reidentification with transferred metric learning. In *ACCV*, 2012.

[Li *et al.*, 2014] Wei Li, Rui Zhao, Tong Xiao, and Xiaogang Wang. Deepreid: Deep filter pairing neural network for person re-identification. In *CVPR*, 2014.

[Li *et al.*, 2018] Chao Li, Cheng Deng, Ning Li, Wei Liu, Xinbo Gao, and Dacheng Tao. Self-supervised adversarial hashing networks for cross-modal retrieval. *arXiv preprint arXiv:1804.01223*, 2018.

[Liu *et al.*, 2016] Xianglong Liu, Cheng Deng, Bo Lang, Dacheng Tao, and Xuelong Li. Query-adaptive reciprocal hash tables for nearest neighbor search. *IEEE Transactions on Image Processing*, 2016.

[Simonyan and Zisserman, 2014] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[Song *et al.*, 2016] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *CVPR*, 2016.

[Varior *et al.*, 2016] Rahul Rama Varior, Mrinal Haloi, and Gang Wang. Gated siamese convolutional neural network architecture for human re-identification. In *ECCV*, 2016.

[Wang *et al.*, 2016] Faqiang Wang, Wangmeng Zuo, Liang Lin, David Zhang, and Lei Zhang. Joint learning of single-image and cross-image representations for person re-identification. In *CVPR*, 2016.

[Weinberger and Saul, 2009] Kilian Q Weinberger and Lawrence K Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 2009.

[Xiao *et al.*, 2016] Tong Xiao, Hongsheng Li, Wanli Ouyang, and Xiaogang Wang. Learning deep feature representations with domain guided dropout for person re-identification. In *CVPR*, 2016.

[Xiong *et al.*, 2014] Fei Xiong, Mengran Gou, Octavia Camps, and Mario Sznaier. Person re-identification using kernel-based metric learning methods. In *ECCV*, 2014.

[Yang *et al.*, 2017a] Erkun Yang, Cheng Deng, Wei Liu, Xianglong Liu, Dacheng Tao, and Xinbo Gao. Pairwise relationship guided deep hashing for cross-modal retrieval. In *AAAI*, 2017.

[Yang *et al.*, 2017b] Yanhua Yang, Cheng Deng, Shangqian Gao, Wei Liu, Dapeng Tao, and Xinbo Gao. Discriminative multi-instance multitask learning for 3d action recognition. *IEEE Transactions on Multimedia*, 2017.

[Yang *et al.*, 2017c] Yanhua Yang, Cheng Deng, Dapeng Tao, Shaoting Zhang, Wei Liu, and Xinbo Gao. Latent max-margin multitask learning with skelets for 3-d action recognition. *IEEE Transactions on Cybernetics*, 2017.

[Yang *et al.*, 2018] Erkun Yang, Cheng Deng, Chao Li, Wei Liu, Jie Li, and Dacheng Tao. Shared predictive cross-modal deep quantization. *IEEE TNNLS*, 2018.

[Zhang *et al.*, 2014] Ziming Zhang, Yuting Chen, and Venkatesh Saligrama. A novel visual word co-occurrence model for person re-identification. In *ECCV*, 2014.

[Zhang *et al.*, 2016] Yaqing Zhang, Xi Li, Liming Zhao, and Zhongfei Zhang. Semantics-aware deep correspondence structure learning for robust person re-identification. In *IJCAI*, 2016.

[Zhao *et al.*, 2013a] Rui Zhao, Wanli Ouyang, and Xiaogang Wang. Person re-identification by salience matching. In *ICCV*, 2013.

[Zhao *et al.*, 2013b] Rui Zhao, Wanli Ouyang, and Xiaogang Wang. Unsupervised salience learning for person re-identification. In *CVPR*, 2013.

[Zhao *et al.*, 2017] Haiyu Zhao, Maoqing Tian, Shuyang Sun, Jing Shao, Junjie Yan, Shuai Yi, Xiaogang Wang, and Xiaoou Tang. Spindle net person re-identification with human body region guided feature decomposition and fusion. In *CVPR*, 2017.

[Zhou *et al.*, 2017] Sanping Zhou, Jinjun Wang, Jiayun Wang, Yihong Gong, and Nanning Zheng. Point to set similarity based deep feature learning for person re-identification. In *CVPR*, 2017.