

# Deep Metric Learning for Practical Person Re-Identification

Dong Yi, Zhen Lei, *Member, IEEE* Stan Z. Li, *Fellow, IEEE*

**Abstract**—Various hand-crafted features and metric learning methods prevail in the field of person re-identification. Compared to these methods, this paper proposes a more general way that can learn a similarity metric from image pixels directly. By using a “siamese” deep neural network, the proposed method can jointly learn the color feature, texture feature and metric in a unified framework. The network has a symmetry structure with two sub-networks which are connected by Cosine function. To deal with the big variations of person images, binomial deviance is used to evaluate the cost between similarities and labels, which is proved to be robust to outliers.

Compared to existing researches, a more practical setting is studied in the experiments that is training and test on different datasets (cross dataset person re-identification). Both in “intra dataset” and “cross dataset” settings, the superiorities of the proposed method are illustrated on VIPeR and PRID.

**Index Terms**—Person Re-Identification, Deep Metric Learning, Convolutional Network, Cross Dataset

## I. INTRODUCTION

The task of person re-identification is to judge whether two person images belong to the same subject or not. In practical applications, the two images are usually captured by two cameras with disjoint views. The performance of person re-identification is closely related to many other applications, such as cross camera tracking, behaviour analysis, object retrieval and so on. The algorithms proposed in this field are also overlapped with other fields in pattern recognition. In recent years, the performance of person re-identification has increased continuously and will increase further.

The essence of person re-identification is very similar to biometric recognition problems, such as face recognition. The core of them is to find a good representation and a good metric to evaluate the similarities between samples. Compared to biometric problems, person re-identification is more challenging due to the low quality and high variety of person images. Person re-identification usually need to match the person images captured by surveillance cameras working in wide-angle mode. Therefore, the resolution of person images are very low (e.g., around  $48 \times 128$  pixels) and the lighting conditions are unstable too. Furthermore, the direction of cameras and the pose of persons are arbitrary. These factors cause the person images under surveillance scenarios have two distinctive properties: large variations in intra class, and ambiguities between inter classes. In summary, the challenges of person re-identification come from the following aspects:

camera view change, pose variation, non-rigid deformation, unstable illumination and low resolution.

However, another challenge is less studied in existing work, which is “cross dataset person re-identification”. In practical systems, we usually collect some large datasets first and train a model on them. Then the trained model is applied to other datasets or videos for person re-identification. We call the training datasets as source domain and test datasets as target domain. The source and target datasets are totally different, because they are usually captured by different cameras under different environments, *i.e.*, have different probability distribution. A practical person re-identification algorithm should have good generalization with respect to the dataset changes. Therefore, cross dataset person re-identification is an important rule to evaluate the performance of algorithms in practice.

Since the pixels of person images are unstable, effective representations are important and needed for person re-identification. To this end, existing methods borrow many sophisticated features from other fields, such as HSV histogram, Gabor, HOG and so on. Based on the features, direct matching or discriminative learning are then used to evaluate the similarity. Existing methods mainly focus on the second step that is how to learn a metric to discriminate the persons. Many good metric learning methods have been proposed in this context, such as KISSME [1], RDC [2] and so on.

The majority of existing methods include two separate steps: feature extraction and metric learning. The features usually come from two separate sources: color and texture, some of which are designed by hand, some of which are learned, and they are finally connected or fused by simple strategies. On the contrary, this paper proposes a new method to combine the separate modules together, learning the color feature, texture feature and metric in a unified framework, which is called as “Deep Metric Learning” (DML).

The main idea of DML is inspired by a “siamese” neural network [3], which is originally proposed for signature verification. Given two person images  $\mathbf{x}$  and  $\mathbf{y}$ , we want to use a siamese deep neural network to assess their similarity  $s = DML(\mathbf{x}, \mathbf{y})$ . Being more specific than the original work [3], our DML first abstracts the siamese network to two sub-networks, a connection function and a cost function (see Figure 1), and then carefully design the architecture for person images. In this way, DML can adapt well to person re-identification. Denoting the connection function as  $S$ , the similarity equation of DML can be written as  $s = DML(\mathbf{x}, \mathbf{y}) = S(B_1(\mathbf{x}), B_2(\mathbf{y}))$ , where  $B_1$  and  $B_2$  denote the two sub-networks of DML. Depending on specific applications,  $B_1$  and  $B_2$  need to share or not to share their

Dong Yi is with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, 100190 e-mail: dyi@cbsr.ia.ac.cn

Manuscript received June 1, 2014; revised June 1, 2014.

parameters.

Compared with existing person re-identification methods, DML has the following advantages:

- 1) DML can learn a similarity metric from image pixels directly. All layers in DML are optimized by a common objective function, which are more effective than the hand-crafted features in traditional methods;
- 2) The multi-channel filters learned in DML can capture the color and texture information simultaneously, which are more reasonable than the simple fusion strategies in traditional methods, *e.g.*, feature concatenation, sum rule;
- 3) The structure of DML is flexible that can switch between view specific and general person re-identification tasks by whether sharing the parameters of sub-networks.

DML is tested on two popular person re-identification datasets, VIPeR [4] and PRID 2011 [5], using the common evaluation protocols. The results show that DML outperforms or on a par with the state-of-the-art methods, such as [6], and [7]. To appeal the practical requirements and evaluate the generalization of DML, we also conduct more challenging cross dataset experiments, which are training on i-LIDS [8], CUHK Campus [6] and testing on VIPeR, PRID. The results of the cross dataset experiments are significantly better than existing methods [9] under similar experimental settings. To our knowledge, this is the first work to conduct strict cross dataset experiment in the field of person re-identification. Finally, we fuse the features learned from multiple datasets to improve the performance on VIPeR and PRID further.

## II. RELATED WORK

This work uses deep learning to learn a metric for person re-identification. Related works in four aspects are reviewed in this section: feature representation, metric learning for person re-identification, siamese convolutional neural network and cross dataset related methods.

Early papers mainly focus on how to construct effective feature representation. From 2005, numerous features are used or proposed for person re-identification [10]. The features can be divided into two categories: color based and texture based features. The most popular features include HSV color histogram [11], [6], LAB color histogram [12], SIFT [12], LBP histogram [6], Gabor features [6] and their fusion. Among these features, color has the most contribution to the final results. The recent advance in this aspect is the color invariant signature [13]. By combining color histogram, covariance feature and segmentation information, [13] achieved rank1=24% on VIPeR.

On the other hand, [11] has proved that using the silhouette and symmetry structure of person can improve the performance significantly, therefore the color and texture features are usually extracted in a predefined grid or finely localized parts. [14] proposed a MCMC based method to do part localization and person re-identification simultaneously, which obtained rank1=23% on VIPeR. The above work all prove that the performance can be improved significantly when know the geometric configuration of person explicitly or implicitly.

Compared to part based method, salience based methods [12], [7] proposed by Zhao *et al.* relaxed the spatial constraint further and could deal with larger pose variations. Similar to the history of face recognition [15], the future direction of feature representation must be based on precise body parts segmentation, person alignment and pose normalization.

Based on the extracted features, naive feature matching or unsupervised learning methods usually got moderate results, and state-of-the-art results were achieved by supervised methods, such as Boosting [16], Rank SVM [17], PLS [17] and Metric learning [2], [1], [6]. Among these methods, metric learning is the main stream due to its flexibility. Compared with standard distance measures, *e.g.*,  $L_1$ ,  $L_2$  norm, the learned metric is more discriminative for the task on hand and more robust to large variations of person images across view. Most papers used a holistic metric to evaluate the similarity of two samples, but [6] first divided the samples into several groups according to their pose and then learned metrics for each group. By using the pose information explicitly, [6] obtains the highest performance on VIPeR. Recently, [18] proposed a novel “human in the loop” style method, which illustrated that the performance of person re-identification can be improved drastically by human intervention. Although the results of this paper is hard to reproduce, it supplies a benchmark to reflect the performance of human (Rank1=71.08% on VIPeR). Closing the gap to human performance is the target of researchers.

Early in 1993, a siamese neural network [3] was proposed to evaluate the similarity of two signature samples. In the same year, a neural network [19] with similar structure was proposed for fingerprint verification. Different from traditional neural networks, the siamese architecture is composed by two sub-networks sharing the same parameters. Each sub-network is a convolutional neural network. Then the siamese neural network was used for face verification [20] by the same research group. The best property of siamese neural network is its unified and clear objective function. Guided by the objective function, the end-to-end neural network can learn a optimal metric towards the target automatically. The responsibility of the last layer of the siamese neural network is to evaluate the similarity of the output of two sub-networks, which can be in any form [20], such as  $L_1$ ,  $L_2$  norm and Cosine. Although good experimental results have been obtained in [3], [19] and [20], their disadvantages are lacking implementation details and lacking comparison with other methods. This paper will design a siamese neural network for person image and apply it in the person re-identification problem. In this paper, the implementation details will be described and extensive comparisons will be reported.

In this young field, cross dataset problem has not attracted much attention. Majority researchers do their best to improve the performance within single dataset, *i.e.*, training on VIPeR and test on VIPeR too. Not long ago, [9] started to concern this issue. In [9], the authors proposed a transfer Rank SVM (DTRSVM) to adapt a model trained on the source domain (i-LIDS or PRID) to target domain (VIPeR). All image pairs in the source domain and the negative image pairs in the target domain were used for training. Different from DTRSVM, in our cross dataset experiments, the proposed network is trained

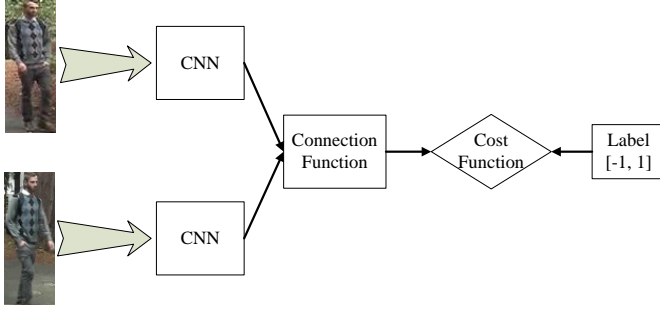


Fig. 1. The structure of the siamese convolutional neural network (SCNN), which is composed by three components: CNN, connection function and cost function.

in the source domain only and its performance is tested in the target domain.

### III. DEEP METRIC LEARNING

Under the joint influence of resolution, illumination and pose changes, the ideal metric for person re-identification may be highly nonlinear. Deep learning is exact one of the most effective tools to learn the nonlinear metric function. This section introduces the architecture, parameters, cost function and implementation details of the proposed convolutional network for deep metric learning.

#### A. Architecture

For most of pattern recognition problems, neural network works in a standalone mode. The input of neural network is a sample and the output is a predicted label. This mode works well for handwritten digit recognition, object recognition and other classification problems when the labels of the training set are the same as the test set. For person re-identification problem, the subjects in the training set are generally different from those in the test set, therefore the “sample  $\rightarrow$  label” style neural network cannot apply to it. To deal with this problem, we construct a siamese neural network, which includes two sub-networks working in a “sample pair  $\rightarrow$  label” mode.

The flowchart of our method is shown in Figure 1. Given two person images, they are sent to siamese convolutional neural network (SCNN). For two images  $x$  and  $y$ , SCNN can predict a label  $l = \pm 1$  to denote whether the image pair comes from the same subject or not. Because many applications need rank the images in the gallery based on their similarities to a probe image, our SCNN outputs a similarity score instead. The structure of the SCNN is shown in Figure 1, which is composed by two convolutional neural networks (CNN). And the two CNNs are connected by a connection function.

Existing siamese neural networks have a constraint that their two sub-networks share the same parameters, *i.e.*, weights and biases. As studied in our previous work [21], this constraint could be removed in some conditions. Without parameters sharing, the network can deal with the view specific matching tasks more naturally. With parameters sharing, the network is more appropriate for general task, *e.g.*, cross dataset person re-identification. We call these two modes as “General” and

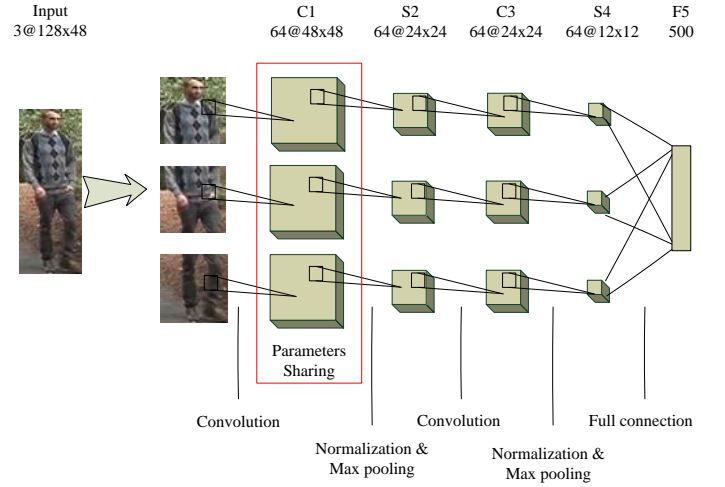


Fig. 2. The structure of the 5-layer CNN used in our method.

“View Specific” SCNN. Because cross dataset problem is the main concern of this paper, we focus on General SCNN.

#### B. Convolutional Neural Network

The CNN in this paper (see Figure 2) is composed by 2 convolutional layers, 2 max pooling layers and a full connected layer. As shown in Figure 2, the number of channels of convolutional and pooling layers are both 64. The output of the CNN is a vector of 500 dimensions. Every pooling layer includes a cross-channel normalization unit. Before convolution the input data are padded by zero values, therefore the output has the same size with input. The filter size of C1 layer is  $7 \times 7$  and the filter size of C2 layer is  $5 \times 5$ . ReLU neuron [22] is used as activation function for each layer.

To capture the different statistical properties of body parts, we train the CNN in part based way. In our previous work [21], person images are cropped into three overlapped parts and three networks are trained independently. Differently, we use a faster scheme in this paper that the three parts are trained jointly. First, the three parts share C1 layer. Second, each part has its own C3 layer, which can help to learn part-specific filters. Third, the high level features of all parts are fused at F5 layer by sum rule. Then, the similarity of fused features are evaluated by the connection function. Driven by a common cost function, the three parts can contribute to the training process jointly.

Overall, there are two main differences between the proposed network in Figure 2 and the network in [21]: 1) C1 layer are shared by three parts or not; 2) the contribution of the three parts are fused in feature level or score level. Parameter sharing in low level can reduce the complexity of the network. Fusion in feature level make the three parts can train jointly, which will improve the performance slightly. Moreover, training and test a single network is more convenient and efficient than using three independent networks.

#### C. Cost Function and Learning

Before learning the parameters of SCNN, we revisit its structure again. As shown in Figure 1, the structure of SCNN

can be abstracted into three basic components: two sub-networks, a connection function and a cost function. Connection function is used to evaluate the relationship between two samples and cost function is used to convert the relationship into a cost. How to choose the connection function and cost function is closely related to the performance of SCNN.

There are many distance, similarity, or other functions can be used as candidates to connect two vectors, such as: Euclidian distance, Cosine similarity, absolute difference, vector concatenate and so on. Their formulas are

$$S_{euc}(\mathbf{x}, \mathbf{y}) = - \sum_i (\mathbf{x}_i - \mathbf{y}_i)^2, \quad (1)$$

$$S_{cos}(\mathbf{x}, \mathbf{y}) = \frac{\sum_i \mathbf{x}_i \mathbf{y}_i}{\sqrt{\sum_i \mathbf{x}_i \mathbf{x}_i \sum_i \mathbf{y}_i \mathbf{y}_i}}, \quad (2)$$

$$S_{abs}(\mathbf{x}, \mathbf{y}) = - \sum_i |\mathbf{x}_i - \mathbf{y}_i|, \quad (3)$$

$$S_{con}(\mathbf{x}, \mathbf{y}) = \sum_i w_i [\mathbf{x}; \mathbf{y}]_i. \quad (4)$$

In the above equations, we negate the distance functions to make them consistent to similarity. The advantage of Euclidian distance is that its derivation has simple form, but its output is unbounded which could make the training process unstable. Absolute difference is non-derivable at some points. Cosine function is bounded to  $[-1, 1]$  and invariant to the magnitude of samples. Because of the good property of Cosine function and it has been used widely in many pattern recognition problems [3], [23], [24], we choose it as the connection function.

For the cost function, [21] has given some analysis about Square loss, Exponential loss, and Binomial deviance [25] and chose Binomial deviance as the final cost function. Here we discuss another popular cost function in pattern recognition: Fisher criterion. Given a training set  $X = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]$ , its corresponding similarity matrix  $S$  and mask matrix  $M$ , Binomial deviance and Fisher criterion are formulated as

$$J_{dev} = \sum_{i,j} W \circ \ln(e^{-\alpha(S-\beta) \circ M} + 1), \quad (5)$$

$$J_{fisher} = - \frac{(\sum_{i,j} P \circ S)^2}{\sum_{i,j} (S - \bar{S})^2}, \quad (6)$$

where  $\circ$  is element-wise matrix product.

$$S = [S_{ij}]_{n \times n}, \quad S_{ij} = S(\mathbf{x}_i, \mathbf{x}_j), \quad (7)$$

$$M = [M_{ij}]_{n \times n}, \quad M_{ij} = \begin{cases} 1, \text{positive pair} \\ -1, \text{negative pair} \\ 0, \text{neglected pair} \end{cases}, \quad (8)$$

$$W = [W_{ij}]_{n \times n}, \quad W_{ij} = \begin{cases} \frac{1}{n_1}, \text{positive pair} \\ \frac{1}{n_2}, \text{negative pair} \\ 0, \text{neglected pair} \end{cases}, \quad (9)$$

$$P = [P_{ij}]_{n \times n}, \quad P_{ij} = \begin{cases} \frac{1}{n_1}, \text{positive pair} \\ -\frac{1}{n_2}, \text{negative pair} \\ 0, \text{neglected pair} \end{cases}. \quad (10)$$

$S_{ij}$  denotes the similarity of sample  $\mathbf{x}_i$  and  $\mathbf{x}_j$ .  $M_{ij}$  denotes whether  $\mathbf{x}_i$  and  $\mathbf{x}_j$  come from the same subject or not.  $n_1$  is the count of positive pairs.  $n_2$  is the count of negative

pairs.  $\bar{S}$  is the mean of  $S$ .  $\alpha$  and  $\beta$  are hyper-parameters of Binomial deviance. The numerator of Eqn. (6) is the between class divergence of similarity matrix  $S$  and the denominator is the total variance. By minimizing Eqn. (5) or Eqn. (6), we can learn a network separating the positive and negative pairs as far as possible.

By comparing Eqn. (5) and Eqn. (6), we can see that Binomial deviance cost more focus on the false classified samples (or the samples near the boundary), but Fisher criterion focus on every elements of the similarity matrix  $S$  equally. In our intuition, Binomial deviance cost can make the network be trained mainly on the hard samples and more likely to get a good model, which will be verified in the experiments. In the following sections, we fix the connection function as Cosine and the cost function as Binomial deviance.

After the connection and cost functions are determined, Back-Propagation (BP) [26] is used to learn the parameters of SCNN. By plugging Eqn. (2) into Eqn. (5), we can get the forward propagation function to calculate the cost of training set.

$$J_{dev}(X) = \sum_{i,j} W \circ \ln(e^{-\alpha(S-\beta) \circ M} + 1), \quad (11)$$

where  $X$  is the output of CNN.  $S$  is the output of connection function.

$$S = [S_{ij}]_{n \times n}, S_{ij} = \frac{\mathbf{x}_i^T \mathbf{x}_j}{\sqrt{\mathbf{x}_i^T \mathbf{x}_i \mathbf{x}_j^T \mathbf{x}_j}}. \quad (12)$$

Differentiating the cost function with respect to  $X$ , we can get

$$\frac{\partial J_{dev}}{\partial X} = X(AB + (AB)^T) - X(\widetilde{AC} + \widehat{AD}), \quad (13)$$

where

$$A = -\alpha W \circ M \circ \frac{e^{-\alpha(S-\beta) \circ M}}{e^{-\alpha(S-\beta) \circ M} + 1}, \quad (14)$$

$$B = [B_{ij}]_{n \times n}, B_{ij} = \frac{1}{\sqrt{\mathbf{x}_i^T \mathbf{x}_i \mathbf{x}_j^T \mathbf{x}_j}}, \quad (15)$$

$$C = [C_{ij}]_{n \times n}, C_{ij} = B_{ij} \frac{\mathbf{x}_i^T \mathbf{x}_j}{\mathbf{x}_i^T \mathbf{x}_i}, \quad (16)$$

$$D = [D_{ij}]_{n \times n}, D_{ij} = B_{ij} \frac{\mathbf{x}_i^T \mathbf{x}_j}{\mathbf{x}_j^T \mathbf{x}_j}, \quad (17)$$

$$\widetilde{AC} = \text{repmat}\left(\left(\sum_j A_{ij} C_{ij}\right)^T, d, 1\right), \quad (18)$$

$$\widehat{AD} = \text{repmat}\left(\sum_i A_{ij} D_{ij}, d, 1\right). \quad (19)$$

where “repmat” is a function to create a matrix by tiling vector many times, as same as the function in Matlab. One can refer Appendix A for the detail derivation of Eqn. (13).

While [21] trains the network in a pairwise way, this paper formulates the cost and gradient in totally matrix form. When the network is trained by Stochastic Gradient Descent (SGD) with mini-batch, the new formulation can process more sample pairs in one batch. When the size of batch is set to 128, in [21] a batch includes 64 positive and 64 negative sample pairs, but in this paper a batch can generate  $128 \times (128 - 1)/2 = 8128$

pairs, which makes the network scans the training data faster and saves the training time.

Based on Eqn. (11) and Eqn. (13), we can learn the parameters of SCNN by SGD algorithm. For general neural network, the error is backward propagated from top to down through a single path. On the contrary, the error of specific SCNN is backward propagated through two branches by Eqn. (27) and Eqn. (33) respectively as described in [21]. In practice, we also can assign asymmetry cost on the label  $M_{ij}$  to positive and negative pairs to tune the performance of network, *e.g.*, 1 for positive pairs and  $-2$  for negative pairs. The effect of asymmetry cost on the performance will be discussed in the experiments.

#### IV. EXPERIMENTS

There are many popular datasets were built for person re-identification, such as VIPeR [4], PRID 2011 [5], i-LIDS [8], CUHK Campus [6] and so on. Among these datasets, the evaluation protocols of VIPeR and PRID (single-shot version) are the clearest two, therefore we compare our method with other methods on them. The experiments are done in two settings:

- 1) Intra dataset experiments
  - training and test both on VIPeR;
  - training and test both on PRID.
- 2) Cross dataset experiments
  - training on i-LIDS and test on VIPeR;
  - training on i-LIDS and test on PRID;
  - training on CUHK Campus and test on VIPeR;
  - training on CUHK Campus and test on PRID.

The intra dataset experiments are conducted to illustrate the basic performance of the proposed method. The cross dataset experiments are to illustrate the generalization ability.

##### A. Intra Dataset

Except for [9] and [21], other papers all conduct experiments in this setting that is training and test on the same dataset. VIPeR includes 632 subjects, 2 images per subject coming from 2 different camera views (camera A and camera B). We split VIPeR into disjoint training (316 subjects) and testing set (316 subjects) randomly, and repeat the process 11 times. The first split (Dev. view) is used for parameter tuning, such as the number of training epoch, learning rate, weight decay and so on. The other 10 splits (Test view) are used for reporting the results.

As similar to VIPeR, PRID is captured by 2 cameras too (camera A and camera B). Camera A shows 385 subjects and camera B shows 749 subjects. The first 200 subjects appear in both cameras. Follow the testing protocols in [5], [27], we randomly select 100 subjects from the first 200 subjects for training and the remain subjects for testing. The test set is composed by probe and gallery, and their information are as follows.

- Probe set: the remain 100 subjects of the first 200 subjects in camera A except for the 100 training subjects;

- Gallery set: the remain 649 subjects of camera B except for the 100 training subjects.

Again, the whole process is repeated 11 times. The first split is used for parameter tuning and the other 10 splits are used for reporting the results.

In the training stage, all training images from camera A and camera B are merged, randomly shuffled and sent to a generic SCNN. The corresponding mask matrix is generated according to the label of samples. Those pairs from the same subjects are assigned 1, and those from the different subjects are assigned  $-1$ . Because the mask matrix is symmetric, we set the elements in the lower triangular part to 0 to avoid redundant computation. In the testing stage, one image of each subject is used as gallery and the other one is used as probe.

Before evaluate on Test view, we use Dev. view to investigate the most three important factors which affect the performance of network: 1) data augmentation, 2) asymmetric cost for positive and negative sample pairs, and 3) cost function. Besides, we set the parameters of the cost function Eqn. (5) as  $\alpha = 2$  and  $\beta = 0.5$ .

1) *Data Augmentation*: Data augmentation is a widely used trick [28] for training neural network. In the experiments, we will mirror all person images to double the training and test sets. Although this trick has been used in [21], but its effect on the performance was not analyzed. Here, we compare the performance on Dev. view of VIPeR with and without data augmentation. For the training set, the number of images of each subject is increased from 2 to 4. In the testing stage, 2 original images and their mirrored version generate 4 similarity scores and the final score are fused by sum rule.

From the results in Table I, we can see significant improvements brought by data augmentation, especially in the top ranks. This indicates that the scale of dataset is crucial to train good networks. If know the geometry information of the person in the image, *i.e.*, the 3D pose of the person, we can augment the datasets guided by the 3D pose and generate more virtual images to improve the performance. In this sense, human 3D pose estimation will be an important direction for person re-identification in the following years.

2) *Asymmetric Cost*: As described in Section III-C, while generating mask matrix from the labels of training samples, the number of negative sample pairs is far more than positive pairs. However, in practice we usually split the training samples into many batches first, and then the positive and negative sample pairs are just generated within each batch. Therefore, the negative sample pairs between batches can not be covered by the training process. This may cause the negative pairs prone to under-fitting.

To balance the weight of positive and negative sample pairs, we can assign asymmetric costs to them. While fixing the cost of positive pair to 1, we tune the cost of negative pair  $c$  from 1 to 3. The asymmetric cost can apply easily on Eqn. (11) and Eqn. (13) by setting

$$M_{ij} = \begin{cases} 1, & \text{positive pair} \\ -c, & \text{negative pair} \\ 0, & \text{neglected pair} \end{cases} \quad (20)$$

Table II shows the relationship between the negative cost

TABLE I  
THE RANK-1 TO RANK-50 RECOGNITION RATES ON DEV. VIEW OF VIPeR WITH OR WITHOUT DATA AUGMENTATION.

Rank	1	5	10	15	20	25	30	50
With Aug.	34.49%	60.13%	74.37%	80.7%	84.18%	88.61%	91.14%	96.84%
Without Aug.	18.4%	49.37%	67.09%	74.05%	80.06%	84.18%	87.04%	96.2%

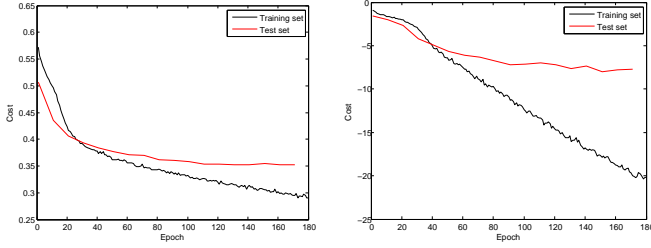


Fig. 3. The epoch-cost curves of the training and test set (Dev. view of VIPeR) when using Binomial deviance (left) or Fisher criterion (right) as cost function.

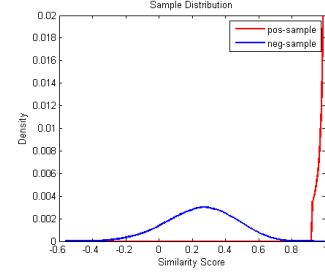


Fig. 5. The similarity score distribution of VIPeR computed based on the 21 human labelled attributes [29]. The 21 attributes are binary and the similarity scores are evaluated by Hamming distance.

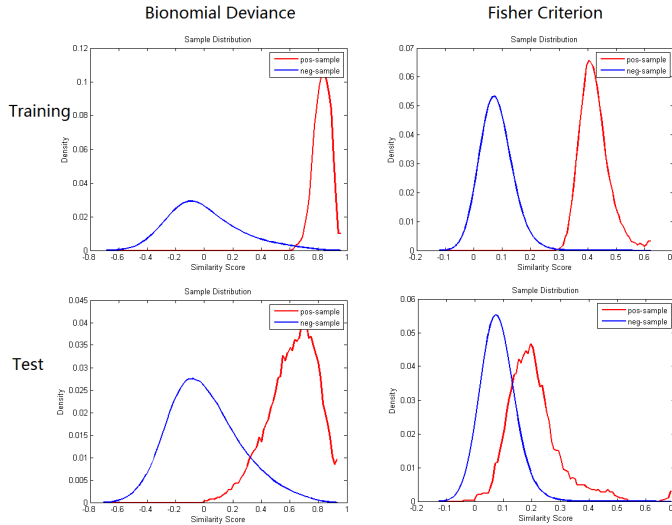


Fig. 4. The similarity score distributions of the training and test set (Dev. view of VIPeR) when using Binomial deviance (left) or Fisher criterion (right) as cost function.

$c$  and recognition rate. On Dev. view of VIPeR, the highest overall performance is achieved at  $c = 2$ . This illustrates that the negative pairs should be paid more attention in each training batch.

3) *Binomial Deviance vs. Fisher Criterion*: Next, we compare two cost functions discussed in Section III-C: Binomial deviance (Eqn. (5)) and Fisher criterion (Eqn. (6)). Like the two prior two experiments, the comparison is conducted on Dev. view of VIPeR too. In the comparison experiments, we mirror the images to double the dataset and set the cost for negative pairs to  $c = 2$ . The connection function is fixed to Cosine. The differences between Binomial deviance and Fisher criterion are evaluated in three aspects: 1) epoch-cost curve, 2) similarity score distribution, and 3) recognition rate.

Figure 3 shows the epoch-cost curves on Dev. view of VIPeR. Low cost reflects high performance approximately, although there is no explicit relationship between the cost

and the recognition rate. No matter using which cost function, the cost of the training set always drop continually. On the contrary, the cost of the test set drops significantly just at the beginning of training process, and then it gradually becomes converged after tens of epochs. From the figure we can see that the cost gap between the training and test set is small when using Binomial deviance. But for Fisher criterion, the gap is obviously bigger than Binomial deviance, which reflects that Fisher criterion is easily overfitting to the training set. By inspecting the epoch-cost curves, we set epoch= 180 based on our experience in the following experiments.

As shown in Figure 4, the two-class similarity distributions of the two cost functions are very different. And a distribution generated from the attributes [29] of VIPeR is also given in Figure 5 for reference. The experiment in [29] has shown that only using the 21 attributes and Hamming distance can achieve very high recognition rate, *i.e.*, rank1  $\approx$  76%. Because the attributes are labelled by human, they can be seen as a baseline of human performance, and the distribution in Figure 5 can be seen as the ideal two-class distribution of VIPeR. For Binomial deviance, the distribution of negative similarity scores are significantly wider than that of positive scores, which is very coincide with the ideal distribution generated from the 21 attributes. For Fisher criterion, the distributions of positive and negative scores are standard Gaussian with the same variance.

Although nearly perfect results can be obtained on the training set using any cost function, their performance on the test set are different (see Table III). From the performance, we can see that Binomial deviance is more suitable for person re-identification problem because it mainly focus on the samples near the boundary and less affected by the distributions of positive and negative samples. For Fisher criterion, the ideal distributions are highly heteroscedastic, which is conflicted with the assumption of Fisher. Maybe some modifications can be made on Fisher criterion to solve this problem, but we leave the work to future.

TABLE II  
THE RECOGNITION RATE ON DEV. VIEW OF VIPeR AT DIFFERENT NEGATIVE COSTS.

Rank	1	5	10	15	20	25	30	50
c = 1	26.9%	53.48%	65.82%	74.05%	78.48%	85.76%	87.34%	94.94%
c = 1.5	31.01%	58.54%	70.89%	78.8%	84.49%	87.66%	90.82%	97.78%
c = 2	34.49%	60.13%	74.37%	80.7%	84.18%	88.61%	91.14%	96.84%
c = 2.5	29.75%	59.18%	71.84%	80.06%	85.13%	89.24%	91.77%	96.2%
c = 3	27.85%	56.65%	70.57%	78.8%	85.44%	87.97%	91.46%	94.94%

TABLE III  
THE RANK-1 TO RANK-50 RECOGNITION RATES ON DEV. VIEW OF VIPeR USING DIFFERENT COST FUNCTIONS.

Rank	1	5	10	15	20	25	30	50
Binomial Deviance	34.49%	60.13%	74.37%	80.7%	84.18%	88.61%	91.14%	96.84%
Fisher Criterion	14.24%	35.13%	47.15%	56.96%	62.66%	67.41%	71.84%	80.06%

4) *Results:* After the above analysis and parameter tuning, we test the performance of our network on Test View of VIPeR and PRID with the following configuration:

- Augment the training and set set by mirror the samples and fused the score by sum;
- Set the negative cost  $c = 2$ ; set the number of epoch = 180;
- Use Cosine as connection function and Binomial Deviance as cost function.

The experiments are repeated 10 times and the mean recognition rates are list in Table IV and Table V.

To distinguish the modified DML from the original one in [21], we rename the proposed method in this paper as “Improved DML”. Compared to the pairwise version in [21], the improved DML has  $5\times$  training speed and significant performance improvement. Notably, the rank-1 recognition rate increases by 6%, from 28.23% to 34.4%. The results of other compared methods are copied from the original papers. If the results are unavailable, they are leaved as “-”. From the table we can see that the proposed method outperforms most of compared methods on VIPeR including the current state-of-the-art methods [6], [7]. From rank-1 to rank-50, our method outperforms [6], [7] remarkably, except that the rank-50 recognition rate of [6] is higher than ours slightly. Among all methods, our method is nearly the most simple and elegant one, which doesn’t need any pre-processing, pose information or segmentation. From the bottom to top layers in the network, every building blocks contribute to a common objective function and are optimized by BP algorithm simultaneously.

On PRID, we get similar results to VIPeR that the proposed method outperforms the state-of-the art method, RPML [27]. However, the superiority of our method on PRID (1% to 3%) is smaller than that on VIPeR (4% to 10%), the reason is that the scale of PRID dataset is too small to train a good network. Just using 100 sample pairs, our network outperforms the compared methods, which can illustrate the power of deep metric learning. Compared to VIPeR, the quality of PRID is poorer and the size of its gallery is bigger, therefore the recognition rates on PRID are overall lower than VIPeR.

#### B. Cross Dataset

In this section, we conduct many experiments in the cross dataset setting which is more coincide with practical applica-

TABLE VI  
CROSS DATASET EXPERIMENT: COMPARISON OF THE PROPOSED METHOD AND DTRSVM [9] ON VIPeR.

Methods	Tr. Set	1	10	20	30
DML [21]	CUHK	16.17%	45.82%	57.56%	64.24%
DTRSVM [9]	PRID	10.9%	28.2%	37.69%	44.87%
DTRSVM [9]	i-LIDS	8.26%	31.39%	<b>44.83%</b>	<b>53.88%</b>
Improved DML	i-LIDS	<b>11.61%</b>	<b>34.43%</b>	44.08%	52.69%
Improved DML	CUHK	<b>16.27%</b>	<b>46.27%</b>	<b>59.94%</b>	<b>70.13%</b>
Improved DML	i-LIDS + CUHK	<b>17.72%</b>	<b>48.8%</b>	<b>63.35%</b>	<b>72.85%</b>

tions. In the following, i-LIDS, CUHK Campus are used as training set, and VIPeR, PRID are used as test set. i-LIDS is captured at a airport in indoor environment; CUHK Campus is captured in a campus; VIPeR and PRID are captured on the street. Due to totally different capture environments and devices, cross dataset experiments are more challenging than the previous experiments. Different from [9], we don’t use any samples in VIPeR and PRID to adapt classifiers to the target domains.

In i-LIDS dataset, there are 119 people with total 476 images captured by multiple non-overlapping cameras with an average of 4 images for each person. Many of these images have large illumination changes and are subject to occlusions. The scale of i-LIDS is smaller than VIPeR and PRID and the resolution of images in i-LIDS are varied. CUHK Campus is a large scale dataset captured by two cameras in 5 sessions, includes 1816 subjects and 7264 images. Each subject has 4 images from 2 camera views. The resolution of CUHK Campus is  $60 \times 160$ . The cross dataset networks are trained with the same parameters as the previous experiments.

Before training, we resize all images in i-LIDS and CUHK Campus to  $40 \times 128$  for convenience. For the test set, we use the same setting with the intra dataset experiments. For VIPeR, A half of subjects and images are randomly selected to construct the test set, which includes 316 subjects and 632 images. For PRID, the test set includes 649 subjects and 749 images. The test process is repeated 10 times too, and the average recognition rate is reported.

1) *Results:* First, we use i-LIDS and CUHK Campus as training set respectively, and test the performance of the trained networks on VIPeR. The recognition rates on VIPeR are shown in Table VI and the results of DTRSVM [9] are



TABLE IV  
INTRA DATASET EXPERIMENT: COMPARISON OF THE PROPOSED METHOD AND OTHER STATE-OF-THE-ART METHODS ON VIPeR.

Method \ Rank	1	5	10	15	20	25	30	50
ELF [16]	12%	31%	41%	-	58%	-	-	-
RDC [2]	15.66%	38.42%	53.86%	-	70.09%	-	-	-
PPCA [30]	19.27%	48.89%	64.91%	-	80.28%	-	-	-
Saliency [12]	26.74%	50.7%	62.37%	-	76.36%	-	-	-
RPML [27]	27%	-	69%	-	83%	-	-	95%
LAF [6]	29.6%	-	69.31%	-	-	88.7%	-	<b>96.8%</b>
Saliency1 [7]	30.16%	52.3%	-	-	-	-	-	-
DML [21]	28.23%	59.27%	73.45%	81.2%	86.39%	89.53%	92.28%	96.68%
Improved DML	<b>34.4%</b>	<b>62.15%</b>	<b>75.89%</b>	<b>82.56%</b>	<b>87.22%</b>	<b>89.65%</b>	<b>92.28%</b>	96.52%

TABLE V  
INTRA DATASET EXPERIMENT: COMPARISON OF THE PROPOSED METHOD AND OTHER STATE-OF-THE-ART METHODS ON PRID 2011.

Method \ Rank	1	5	10	15	20	25	30	50
Descr. Model [5]	4%	-	24%	-	37%	-	-	56%
RPML [27]	15%	-	42%	-	54%	-	-	70%
Improved DML	<b>17.9%</b>	<b>37.5%</b>	<b>45.9%</b>	<b>50.7%</b>	<b>55.4%</b>	<b>59.3%</b>	<b>63.1%</b>	<b>71.4%</b>

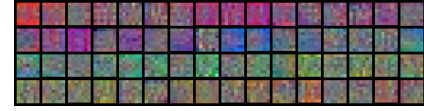
TABLE VII  
CROSS DATASET EXPERIMENT: COMPARISON OF THE PROPOSED METHOD AND DTRSVM [9] ON PRID 2011.

Methods	Tr. Set	1	10	20	30
DTRSVM [9]	VIPeR	4.6%	17.25%	22.9%	28.1%
DTRSVM [9]	i-LIDS	3.95%	18.85%	26.6%	33.2%
Improved DML	i-LIDS	<b>8%</b>	<b>25.5%</b>	<b>38.9%</b>	<b>45.6%</b>
Improved DML	CUHK	<b>7.6%</b>	<b>23.4%</b>	<b>30.9%</b>	<b>36.1%</b>
Improved DML	i-LIDS + CUHK	<b>13.8%</b>	<b>35.4%</b>	<b>45%</b>	<b>51.3%</b>

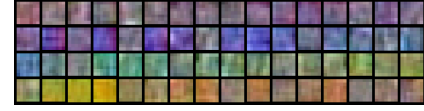
listed for comparison. From the results we can see that our method outperforms the original DML in [21] slightly. When training on i-LIDS, our method is on a par with DTRSVM, and the rank-1 and rank-10 recognition rates are better than DTRSVM slightly. When training on CUHK Campus, our performance improved significantly. A possible reason is that compared to i-LIDS the quality and aspect ratio of images in CUHK Campus is more similar to VIPeR. By combining the similarity scores of the two networks by sum, the performance is improved further and even approaches the performance of some methods in intra dataset setting, such as ELF [16] and RDC [2].

Then we test the two trained networks on PRID, and give the results in Table VII. When using i-LIDS as training set, our method is better than DTRSVM significantly. But different from the results on VIPeR, training on CUHK Campus decreases the recognition rates remarkably. The performance decline is caused by the big difference between CUHK Campus and PRID datasets. Fusing the similarity scores of i-LIDS and CHUK Campus increases the performance too, especially the rank-1 recognition raises from 8% to 13.8%.

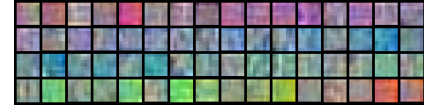
Compared to the intra dataset experiments, the performance of cross dataset experiments decline sharply. For rank-30 recognition rate, the number of VIPeR and PRID both drop more than 20%, which indicates the trained models are hard to generalize across datasets due to the distinctive properties of each dataset. Figure 6 shows some filters learned from



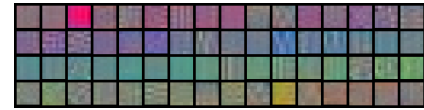
(a) VIPeR



(b) PRID



(c) i-LIDS



(d) CUHK

Fig. 6. The filters in the first layer of CNN, from top to down, which are learned on VIPeR, PRID, i-LIDS, CUHK Campus respectively. The size of filters are  $7 \times 7$ , and their order are sorted by Hue component for best view.

different datasets, from which we can see the distributions of color and texture on these datasets are very diverse. Besides of the experimental results, the diverse filters also can prove that the model learned on a dataset can hardly adapt to another one. How to transfer a model to target domain or how to fully use the multiple heterogeneous datasets at hand to improve the performance in target domain are important research topics in the future.

## V. CONCLUSIONS

This paper proposed a deep metric learning method by using siamese convolutional neural network. The structure of the network and the training process were described in



detail. Extensive intra dataset and cross dataset person re-identification experiments were conducted to illustrate the superiorities of the proposed method. This is the first work to apply deep learning in the person re-identification problem and is also the first work to study the person re-identification problem in fully cross dataset setting. The experimental results illustrated that the network can deal with the cross view and cross dataset person re-identification problems efficiently and outperformed the state-of-the-art methods significantly. In the future, we will apply DML to other applications; explore the way to pre-train the network; and investigate how to embed geometry information into the network to improve the robustness to pose variations. Moreover, we will continue to research how to train a general person matching engine with good generalization across view and dataset.

#### APPENDIX A GRADIENTS OF GENERAL SCNN

For general SCNN, the two sub-networks share their parameters, therefore they have the same output for a sample. Input a training set, we denote the output of CNN by  $X = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]$ , where  $n$  is the number of samples. The cost produced by the training set  $X$  can be calculated by Eqn. (11). The gradient of Eqn. (11) with respect to  $X$  can be derived as follows.

$$\begin{aligned} \frac{\partial J_{dev}}{\partial X} &= \sum_{ij} \frac{\partial J_{ij}}{\partial X} \\ &= \sum_{ij} -\alpha W \circ M \circ \frac{e^{-\alpha(S-\beta) \circ M}}{e^{-\alpha(S-\beta) \circ M} + 1} \frac{\partial S_{ij}}{\partial X} \\ &= \sum_{ij} A_{ij} \frac{\partial S_{ij}}{\partial X}, \end{aligned} \quad (21)$$

where  $A_{ij}$  can be seen as a weight for each sample pair. And  $\frac{\partial S_{ij}}{\partial X}$  is the derivative of the similarity  $S_{ij}$  with respect to  $X$ .

Expand the Cosine similarity  $S_{ij}$  by Eqn. (2), we can get

$$\begin{aligned} \frac{\partial S_{ij}}{\partial X} &= \frac{\sqrt{\mathbf{x}_i^T \mathbf{x}_i \mathbf{x}_j^T \mathbf{x}_j} \frac{\partial \mathbf{x}_i^T \mathbf{x}_j}{\partial X} - \mathbf{x}_i^T \mathbf{x}_j \frac{\partial \sqrt{\mathbf{x}_i^T \mathbf{x}_i \mathbf{x}_j^T \mathbf{x}_j}}{\partial X}}{\mathbf{x}_i^T \mathbf{x}_i \mathbf{x}_j^T \mathbf{x}_j} \\ &= \frac{1}{\sqrt{\mathbf{x}_i^T \mathbf{x}_i \mathbf{x}_j^T \mathbf{x}_j}} \\ &\quad \left( \frac{\partial \mathbf{x}_i^T \mathbf{x}_j}{\partial X} - \frac{\mathbf{x}_i^T \mathbf{x}_j}{2 \mathbf{x}_i^T \mathbf{x}_i} \frac{\partial \mathbf{x}_i^T \mathbf{x}_i}{\partial X} - \frac{\mathbf{x}_i^T \mathbf{x}_j}{2 \mathbf{x}_j^T \mathbf{x}_j} \frac{\partial \mathbf{x}_j^T \mathbf{x}_j}{\partial X} \right) \\ &= B_{ij} \frac{\partial \mathbf{x}_i^T \mathbf{x}_j}{\partial X} - \frac{C_{ij}}{2} \frac{\partial \mathbf{x}_i^T \mathbf{x}_i}{\partial X} - \frac{D_{ij}}{2} \frac{\partial \mathbf{x}_j^T \mathbf{x}_j}{\partial X}, \end{aligned} \quad (22)$$

where  $B_{ij}$ ,  $C_{ij}$  and  $D_{ij}$  are defined in Eqn. (15), Eqn. (16) and Eqn. (17) respectively. The derivatives of  $\mathbf{x}_i^T \mathbf{x}_j$ ,  $\mathbf{x}_i^T \mathbf{x}_i$  and  $\mathbf{x}_j^T \mathbf{x}_j$  with respect to  $X$  are as follows.

$$\frac{\partial \mathbf{x}_i^T \mathbf{x}_j}{\partial X} = \begin{pmatrix} 0 & x_j & 0 & x_i & 0 \end{pmatrix} \quad (23)$$

$$\frac{\partial \mathbf{x}_i^T \mathbf{x}_i}{\partial X} = 2 \begin{pmatrix} 0 & x_i & 0 \end{pmatrix} \quad (24)$$

$$\frac{\partial \mathbf{x}_j^T \mathbf{x}_j}{\partial X} = 2 \begin{pmatrix} 0 & x_j & 0 \end{pmatrix} \quad (25)$$

By substituting Eqn. (23), Eqn. (24) and Eqn. (25) into Eqn. (22) and substituting Eqn. (22) into Eqn. (21), we can get the final formulation of  $\frac{\partial J_{dev}}{\partial X}$ .

$$\begin{aligned} \frac{\partial J_{dev}}{\partial X} &= \sum_{ij} A_{ij} B_{ij} \begin{pmatrix} 0 & x_j & 0 & x_i & 0 \end{pmatrix} \\ &\quad - \sum_{ij} A_{ij} C_{ij} \begin{pmatrix} 0 & x_i & 0 \end{pmatrix} \\ &\quad - \sum_{ij} A_{ij} D_{ij} \begin{pmatrix} 0 & x_j & 0 \end{pmatrix} \\ &= X(AB + (AB)^T) - X \circ (\widehat{AC} + \widehat{AD}), \end{aligned} \quad (26)$$

where  $\widehat{AC}$  is defined in Eqn. (18) and  $\widehat{AD}$  is defined in Eqn. (19).

#### APPENDIX B GRADIENTS OF VIEW SPECIFIC SCNN

Although this paper doesn't use specific (asymmetric) SCNN for experiment, we give the gradients of cost function for reference. We denote the output of two sub-networks by  $X = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]$  and  $Y = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_m]$ , where  $n$  is the number of samples in view1 and  $m$  is the number of samples in view2. The dimensions of their corresponding similarity matrix  $S$  and mask matrix  $M$  are  $n \times m$ .

Following the derivation process in Appendix A, the gradients of  $J_{dev}$  with respect to  $X$  and  $Y$  are as follows.

$$\frac{\partial J_{dev}}{\partial X} = Y(EF)^T - X \circ \widehat{EG}, \quad (27)$$

where

$$E = -\alpha W \circ M \circ \frac{e^{-\alpha(S-\beta) \circ M}}{e^{-\alpha(S-\beta) \circ M} + 1}, \quad (28)$$

$$F = [F_{ij}]_{n \times n}, F_{ij} = \frac{1}{\sqrt{\mathbf{x}_i^T \mathbf{x}_i \mathbf{y}_j^T \mathbf{y}_j}}, \quad (29)$$

$$G = [G_{ij}]_{n \times n}, G_{ij} = F_{ij} \frac{\mathbf{x}_i^T \mathbf{y}_j}{\mathbf{x}_i^T \mathbf{x}_i}, \quad (30)$$

$$\widehat{EG} = \text{repmat}\left(\left(\sum_j E_{ij} G_{ij}\right)^T, d, 1\right). \quad (31)$$

$$(32)$$

And

$$\frac{\partial J_{dev}}{\partial Y} = X(EF) - Y \circ \widehat{EH}, \quad (33)$$

where

$$H = [H_{ij}]_{n \times n}, H_{ij} = F_{ij} \frac{\mathbf{x}_i^T \mathbf{y}_j}{\mathbf{y}_j^T \mathbf{y}_j}, \quad (34)$$

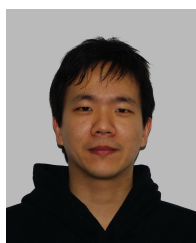
$$\widehat{EH} = \text{repmat}\left(\sum_i E_{ij} H_{ij}, d, 1\right). \quad (35)$$

## ACKNOWLEDGMENT

This work was supported by the Chinese National Natural Science Foundation Projects #61105023, #61103156, #61105037, #61203267, #61375037, National Science and Technology Support Program Project #2013BAK02B01, Chinese Academy of Sciences Project No. KGZD-EW-102-2, Jiangsu Science and Technology Support Program Project #BE2012627, and AuthenMetric R&D Funds.

## REFERENCES

- [1] M. Kostinger, M. Hirzer, P. Wohlhart, P. Roth, and H. Bischof, "Large scale metric learning from equivalence constraints," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2012, pp. 2288–2295.
- [2] W.-S. Zheng, S. Gong, and T. Xiang, "Reidentification by relative distance comparison," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 3, pp. 653–668, 2013.
- [3] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a siamese time delay neural network," in *NIPS*, 1993, pp. 737–744.
- [4] D. Gray, S. Brennan, and H. Tao, "Evaluating appearance models for recognition, reacquisition, and tracking," in *In IEEE International Workshop on Performance Evaluation for Tracking and Surveillance, Rio de Janeiro*, 2007.
- [5] M. Hirzer, C. Belezni, P. Roth, and H. Bischof, "Person re-identification by descriptive and discriminative classification," in *Image Analysis*, ser. Lecture Notes in Computer Science, A. Heyden and F. Kahl, Eds. Springer Berlin Heidelberg, 2011, vol. 6688, pp. 91–102.
- [6] W. Li and X. Wang, "Locally aligned feature transforms across views," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, 2013, pp. 3594–3601.
- [7] R. Zhao, W. Ouyang, and X. Wang, "Person re-identification by salience matching," in *Proceedings of IEEE International Conference on Computer Vision*, 2013, pp. 2528–2535.
- [8] W.-S. Zheng, S. Gong, and T. Xiang, "Person re-identification by probabilistic relative distance comparison," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, 2011, pp. 649–656.
- [9] A. J. Ma, P. C. Yuen, and J. Li, "Domain transfer support vector ranking for person re-identification without target camera label information," in *Proceedings of IEEE International Conference on Computer Vision*, 2013, pp. 3567–3574.
- [10] O. Javed, K. Shafique, and M. Shah, "Appearance modeling for tracking in multiple non-overlapping cameras," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2, 2005, pp. 26–33 vol. 2.
- [11] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani, "Person re-identification by symmetry-driven accumulation of local features," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 2010, pp. 2360–2367.
- [12] R. Zhao, W. Ouyang, and X. Wang, "Unsupervised salience learning for person re-identification," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, 2013, pp. 3586–3593.
- [13] I. Kviatkovsky, A. Adam, and E. Rivlin, "Color invariants for person reidentification," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 7, pp. 1622–1634, 2013.
- [14] Y. Xu, L. Lin, W.-S. Zheng, and X. Liu, "Human re-identification by matching compositional template with cluster sampling," *ICCV*, 2013.
- [15] X. Zhang and Y. Gao, "Face recognition across pose: A review," *Pattern Recognition*, vol. 42, no. 11, pp. 2876–2896, 2009.
- [16] D. Gray and H. Tao, "Viewpoint invariant pedestrian recognition with an ensemble of localized features," in *Computer Vision C ECCV 2008*, ser. Lecture Notes in Computer Science, D. Forsyth, P. Torr, and A. Zisserman, Eds. Springer Berlin Heidelberg, 2008, vol. 5302, pp. 262–275.
- [17] B. Prosser, W.-S. Zheng, S. Gong, and T. Xiang, "Person re-identification by support vector ranking," in *BMVC*, 2010, pp. 1–11.
- [18] C. Liu, C. C. Loy, S. Gong, and G. Wang, "Pop: Person re-identification post-rank optimisation," in *International Conference on Computer Vision*, 2013.
- [19] P. Baldi and Y. Chauvin, "Neural networks for fingerprint recognition," *Neural Computation*, vol. 5, no. 3, pp. 402–418, 1993.
- [20] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *CVPR (1)*, 2005, pp. 539–546.
- [21] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Deep metric learning for person re-identification," in *Proceedings of International Conference on Pattern Recognition*, 2014, pp. 2666–2672.
- [22] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012, pp. 1106–1114.
- [23] T. Qin, X.-D. Zhang, M.-F. Tsai, D.-S. Wang, T.-Y. Liu, and H. Li, "Query-level loss functions for information retrieval," *Information Processing & Management*, vol. 44, no. 2, pp. 838–855, 2008.
- [24] H. Nguyen and L. Bai, "Cosine similarity metric learning for face verification," in *Computer Vision C ACCV 2010*, ser. Lecture Notes in Computer Science, R. Kimmel, R. Klette, and A. Sugimoto, Eds. Springer Berlin Heidelberg, 2011, vol. 6493, pp. 709–720.
- [25] J. Friedman, R. Tibshirani, and T. Hastie, *The Elements of Statistical Learning : Data Mining, Inference, and Prediction*, ser. Springer Series in Statistics. New York: Springer-Verlag, 2009.
- [26] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [27] M. Hirzer, P. Roth, M. Köstinger, and H. Bischof, "Relaxed pairwise learned metric for person re-identification," in *Computer Vision C ECCV 2012*, ser. Lecture Notes in Computer Science, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Springer Berlin Heidelberg, 2012, vol. 7577, pp. 780–793.
- [28] D. C. Cireşan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber, "Flexible, high performance convolutional neural networks for image classification," in *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Volume Two*. AAAI Press, 2011, pp. 1237–1242.
- [29] R. Layne, T. M. Hospedales, and S. Gong, "Attributes-based re-identification," in *Person Re-Identification*. Springer, 2014, pp. 93–117.
- [30] A. Mignon and F. Jurie, "Pcca: A new approach for distance learning from sparse pairwise constraints," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2012, pp. 2666–2672.



for the Shenzhen-Hongkong immigration control project and 2008 Beijing Olympic Games.

**Dong Yi** received the B.S. degree in electronic engineering in 2003, the M.S. degree in communication and information system from Wuhan University, Wuhan, China, in 2006, and received the Ph.D. degree in pattern recognition and intelligent systems from CASIA, Beijing, China, in 2009. His research areas are unconstrained face recognition, heterogeneous face recognition, and deep learning. He has authored and acted as a reviewer for tens of articles in international conferences and journals. He developed the face biometric algorithms and systems

PLACE  
PHOTO  
HERE

**Zhen Lei** received the B.S. degree in automation from the University of Science and Technology of China (USTC), in 2005, and the Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences (CASIA), in 2010, where he is now an assistant professor. His research interests are in computer vision, pattern recognition, image processing, and face recognition in particular. He has published over 40 papers in international journals and conferences.

PLACE  
PHOTO  
HERE

**Stan Z. Li** received the B.Eng. degree from Hunan University, Changsha, China, the M.Eng. degree from the National University of Defense Technology, China, and the Ph.D. degree from Surrey University, Surrey, U.K. He is currently a Professor and the Director of Center for Biometrics and Security Research (CBSR), Institute of Automation, Chinese Academy of Sciences (CASIA). He worked at Microsoft Research Asia as a researcher from 2000 to 2004. Prior to that, he was an Associate Professor at Nanyang Technological University, Singapore. His research interest includes pattern recognition and machine learning, image and vision processing, face recognition, biometrics, and intelligent video surveillance. He has published over 200 papers in international journals and conferences, and authored and edited eight books.