# Person Re-Identification via Distance Metric Learning With Latent Variables

Chong Sun, Dong Wang, and Huchuan Lu

***Abstract*— In this paper, we propose an effective person re-identification method with latent variables, which represents a pedestrian as the mixture of a holistic model and a number of flexible models. Three types of latent variables are introduced to model uncertain factors in the re-identification problem, including vertical misalignments, horizontal misalignments and leg posture variations. The distance between two pedestrians can be determined by minimizing a given distance function with respect to latent variables, and then be used to conduct the re-identification task. In addition, we develop a latent metric learning method for learning the effective metric matrix, which can be solved via an iterative manner: once latent information is specified, the metric matrix can be obtained based on some typical metric learning methods; with the computed metric matrix, the latent variables can be determined by searching the state space exhaustively. Finally, extensive experiments are conducted on seven databases to evaluate the proposed method. The experimental results demonstrate that our method achieves better performance than other competing algorithms.**

***Index Terms*— Person re-identification, latent variables, metric learning, spatial misalignments.**

## I. Introduction

PERSON re-identification is a fundamental task in visual surveillance, which aims to match pedestrians across several non-overlapping camera views. It has drawn increasing attentions in the past decades for its wide applications, and many comprehensive datasets [1]–[4] are proposed. Though many progresses have been achieved, person re-identification is still a very challenging task due to its intrinsic difficulties, including viewpoint and pose variations, occlusions, cluttered background, to name a few. Broadly speaking, approaches on this task can be mainly divided into two categories: (1) feature design and selection [5]–[10]; and (2) metric learning [9], [11]–[14].

As a fundamental issue in person re-identification, feature design and selection have been extensively studied. Gray and Tao [5] exploit a boosting algorithm to learn an accurate representation for a pedestrian instead of designing a specific feature by hand. Farenzena *et al.* [6] utilize the symmetry and asymmetry nature of a pedestrian, and propose three complementary representations for a pedestrian, which are weighted color histograms, maximally stable color regions and recurrent high-structured patches. Ma *et al.* [7] introduce the biologically inspired features (BIF), and propose a feature descriptor called BiCov by computing the similarity of the BIF features at neighboring scales. Layne *et al.* [10] divide the semantic attributes for the person re-identification problem into 15 categories, and learn the attribute features for each pedestrian via support vector machines. Shi *et al.* [15] compute the semantic descriptors by integrating a Markov random field with the Indian Buffet Process (IBP) model, and transfer the model learned from the fashion photography datasets to person re-identification. Recently, Zheng *et al*. [16] propose a novel method to fuse different kinds of features based on the shape of the sorted score curve. Be superior over other feature fusion methods, this paper has been demonstrated to have good resistance to bad features.

With respect to the former issues, metric learning methods have been proved very effective in person re-identification. Zheng *et al.* [11] try to optimize the probability that intra-class distances are smaller than inter-class distances, which overcomes the overfitting problem with sparse training data. Li *et al.* [13] exploit a transfer learning framework which learns a maximum margin generic metric and transfer the generic metric to candidate specific metrics. Hirzer *et al.* [17] relax the hard constraints exploited in previous metric learning methods, and achieve the state-of-the-art performance with less computation load. Köstinger *et al.* [18] start with the prospect of statistical inference, and learn a metric from equivalence constraints. A closed-form solution can be found in this method, which makes it faster than other metric learning methods with comparable results. Based on the observation that most metric learning algorithms do not perform well with sparse pairwise constraints, Mignon and Jurie [14] learn a projection which maps the original data into a low dimensional space. This method works well for limited training data whose dimension is high. Li *et al.* [9] propose a method (named as LADF) that simultaneously learns the metric matrix and the adaptive threshold term in a unified formula. With more model parameters, this method has been demonstrated to have lower verification error rate than methods with fixed thresholds.

Although the above mentioned metric learning methods have shown great success in various fields, all of them do not take person re-identification as a specific issue for

Manuscript received December 12, 2015; revised August 4, 2016; accepted October 5, 2016. Date of publication October 19, 2016; date of current version November 15, 2016. The work of C. Sun, D. Wang, and H. Lu was supported by the Natural Science Foundation of China under Grant 61502070, Grant 61528101, and Grant 61472060. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Shiguang Shan.

The authors are with the School of Information and Communication Engineering, Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian 116024, China (e-mail: waynecool@mail.dlut.edu.cn; wdice@dlut.edu.cn; lhchuan@dlut.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2016.2619261


Fig. 1. Some examples for the VIPeR, GRID, PRID450S and iLIDS datasets. Images in each column are from the same pedestrian with different camera views. In this figure, all the images are resized to the same size for presentation.

consideration. As is shown in the example images in Figure 1, there exist considerable spatial misalignments and posture variations between images from the same pedestrian in different camera views. The above mentioned metric learning methods fail to achieve satisfactory performance in such cases.

To handle spatial misalignments, Zhao *et al*. [19]–[21] propose several methods which exploit patches for distance computation. In [19], patch saliency is learned in an unsupervised way, and the saliency score is used to assign a weight for each patch. In contrast with the unsupervised saliency based method [19], paper [20] integrates saliency matching into a structural RankSVM framework with a partial order constraint where the matching terms are more properly evaluated than [19]. In [21], patches are divided into three categories, which are general patches, rare patches and effective patches. This paper computes the distance between two pedestrians based on the effective patches, and achieves a good balance between discriminative power and generalization ability. Xu *et al.* [22] represent the human body as an articulated structure, and propose a part matching algorithm with cluster sampling. This method has a good performance at the cost of high computation load. Overall, the above mentioned methods perform well in addressing spatial alignments. However, they do not fully exploit the advantages of the metric learning method, and still need to be improved.

Facing the similar challenges, fields like object detection and visual tracking usually address such challenges by seeking the help of latent variables. In recent years, latent variables have been successfully integrated with the support vector machines (SVM), which leads to a new method called latent support vector machines (LSVM). Yu and Joachims [23] integrate latent variables with the structural SVM framework, and learn model parameters with the concave-convex procedure (CCCP). Analogous to the EM algorithm, the concave-convex procedure alternates between two steps: (1) estimate the latent variables using the computed model parameters; (2) optimize the model parameters with the estimated latent variables. Based on [23], Vedaldi and Zisserman [24] exploit the latent variables to consider information like spatial alignments and the truncation of instances. Their method shows superior performance over previous svm-based methods in object detection. Still in object detection, Zhu *et al.* [25] propose a latent hierarchical structural learning method. An incremental concave-convex procedure (iCCCP) is proposed to avoid the complex training process in a hierarchical structure. Yao *et al.* [26] extend the LSVM method with an online manner, and introduce it to the field of visual tracking. Though latent variables have been proved to be very effective combined with SVM, it is not well studied in the metric learning methods.

Inspired by the great success of patch based methods and latent variables, we try to integrate the strength of them. In this work, three kinds of flexible patch based models are proposed to address spatial misalignments and leg posture variations, where each model defines a distance between patches of two pedestrians. Corresponding to the flexible models, three kinds of variables are exploited to indicate the states (positions or angles) of the patches. These variables are not labeled in the training set, and are regarded as latent variables in the training process. We propose a novel latent metric learning (LML) method to learn the patch based models with latent variables, through which model learning and spatial alignments are iteratively performed. Extensive experiments on seven public available datasets validate the effectiveness of the proposed method.

The rest of this paper is organized as follows. In Section II, we first introduce the formula of metric learning method with latent variables and present the proposed optimization algorithm. In Section III, we introduce the specific distances with latent variables to model vertical misalignments, horizontal misalignments and the variations of leg postures, based on which the overall distance for the re-identification problem is effectively defined. Then, Section IV evaluates and analyzes the proposed method with extensive experiments and comparisons with many state-of-the-art algorithms. Finally, Section V concludes this paper.

## II. Proposed Algorithm

In this section, we first introduce the proposed metric learning method with latent variables based on the mahalanobis distance metric, and then extend it by introducing the distance defined by the locally-adaptive decision function (LADF) [9].

Given a set of training samples $(\mathbf{x}_\phi, y_\phi)_{\phi=1}^N$, where $y_\phi$ is the class label (in our case, each pedestrian is regarded as one class), $\mathbf{x}_\phi$ denotes the image for the $\phi$-th sample. The task of the metric learning method is to enhance the similarities among samples of the same class and the differences among samples of different classes in the meanwhile. For arbitrary

samples $\mathbf{x}_\phi$ and $\mathbf{x}_\varphi$, we compute their distance based on patches, and exploit two latent variables $m^\phi_{\phi,\varphi} \in \mathcal{M}_\phi$ and $m^\varphi_{\phi,\varphi} \in \mathcal{M}_\varphi$ to denote the patch states (positions or angles) of the two samples. For convenience, we exploit the symbol $m$ to represent the three kinds of latent variables, which will be detailed later.

The distance between samples $\mathbf{x}_\phi$ and $\mathbf{x}_\varphi$ can be defined as

$$d_{\phi,\varphi} = \min_{m^\phi_{\phi,\varphi}, m^\varphi_{\phi,\varphi}} (\mathbf{x}_{\phi,m^\phi_{\phi,\varphi}} - \mathbf{x}_{\varphi,m^\varphi_{\phi,\varphi}})^\top \mathbf{M} (\mathbf{x}_{\phi,m^\phi_{\phi,\varphi}} - \mathbf{x}_{\varphi,m^\varphi_{\phi,\varphi}}), \tag{1}$$

where $\mathbf{x}_{\phi,m^\phi_{\phi,\varphi}}$ is the feature extracted in the patch specified by the variable $m^\phi_{\phi,\varphi}$ for the $\phi$-th sample, $\mathbf{M}$ is the metric matrix to be learned. We note that the exact values of $m^\phi_{\phi,\varphi}$ and $m^\varphi_{\phi,\varphi}$ are unknown in the training process.

In this work, we learn the model parameter $\mathbf{M}$ by solving the following optimization problem,

$$\begin{aligned}\hat{\mathbf{M}} = \arg\min_{\mathbf{M}} & \frac{1}{N_1} \sum_{\phi=1}^{N} \sum_{\substack{\varphi=1,\\ \phi\neq\varphi,\\ y_\phi = y_\varphi}}^{N} \min_{m^\phi_{\phi,\varphi}, m^\varphi_{\phi,\varphi}} D_1(\mathbf{x}_{\phi,m^\phi_{\phi,\varphi}}, \mathbf{x}_{\varphi,m^\varphi_{\phi,\varphi}}) \\ & + \frac{1}{N_2} \sum_{\phi=1}^{N} \sum_{\substack{\varphi=1,\\ y_\phi \neq y_\varphi}}^{N} D_2(\mathbf{x}_\phi, \mathbf{x}_\varphi) + \frac{\lambda}{2} \|\mathbf{M}\|_F^2 \\ & s.t.\ \mathbf{M} \succeq 0 \end{aligned} \tag{2}$$

where $N_1$ denotes the number of positive training pairs, $N_2$ is the number of negative training pairs.

This optimization problem tries to minimize the distances between the samples in the same class and maximize the distances between the samples of different classes in the meanwhile. The first term aims to describe the losses for the distances of intra-class samples, and the second term aims the describe the losses for the distances of inter-class samples. The detailed introductions on different terms in equation (2) are as follows.

$D_1(\mathbf{x}_{\phi,m^\phi_{\phi,\varphi}}, \mathbf{x}_{\varphi,m^\varphi_{\phi,\varphi}})$ is defined as

$$\begin{aligned} D_1(\mathbf{x}_{\phi,m^\phi_{\phi,\varphi}}, \mathbf{x}_{\varphi,m^\varphi_{\phi,\varphi}}) = l_\varepsilon \Big( & (\mathbf{x}_{\phi,m^\phi_{\phi,\varphi}} - \mathbf{x}_{\varphi,m^\varphi_{\phi,\varphi}})^\top \\ & \times \mathbf{M} (\mathbf{x}_{\phi,m^\phi_{\phi,\varphi}} - \mathbf{x}_{\varphi,m^\varphi_{\phi,\varphi}}) - 1 \Big) \end{aligned} \tag{3}$$

This term aims to compute the cost that the distance between intra-class features $\mathbf{x}_{\phi,m^\phi_{\phi,\varphi}}$ and $\mathbf{x}_{\varphi,m^\varphi_{\phi,\varphi}}$ is larger than 1. $l_\varepsilon(x)$ is the generalized logistic loss function which is defined as $l_\varepsilon(x) = \frac{1}{\varepsilon}\log(1 + e^{\varepsilon x})$ ($\varepsilon = 1$ in the implementation). It is an approximation of the hinge loss function.

$D_2(\mathbf{x}_\phi, \mathbf{x}_\varphi)$ is defined as

$$\begin{aligned} D_2(\mathbf{x}_\phi, \mathbf{x}_\varphi) = \sum_{\substack{m^\phi_{\phi,\varphi} \in \mathcal{M}_\phi,\\ m^\varphi_{\phi,\varphi} \in \mathcal{M}_\varphi}} l_\varepsilon \Big( & -(\mathbf{x}_{\phi,m^\phi_{\phi,\varphi}} - \mathbf{x}_{\varphi,m^\varphi_{\phi,\varphi}})^\top \\ & \times \mathbf{M} (\mathbf{x}_{\phi,m^\phi_{\phi,\varphi}} - \mathbf{x}_{\varphi,m^\varphi_{\phi,\varphi}}) + 1 \Big) \end{aligned} \tag{4}$$

This term aims to compute the cost that the distance between inter-class features $\mathbf{x}_{\phi,m^\phi_{\phi,\varphi}}$ and $\mathbf{x}_{\varphi,m^\varphi_{\phi,\varphi}}$ is smaller than 1. Note that we sum the losses of different latent states for samples $\mathbf{x}_\phi$ and $\mathbf{x}_\varphi$ if they do not belong to the same class.

To the best of our knowledge, there exists no closed-form solution for the optimization problem (2). Thus, we present an iterative algorithm to solve it via the following two steps: **I:** With the computed $\hat{\mathbf{M}}$, we obtain $(\hat{m}^\phi_{\phi,\varphi}, \hat{m}^\varphi_{\phi,\varphi})_{y_\phi = y_\varphi}$, $\phi = 1, \ldots, N, \varphi = 1, \ldots, N$ by exhaustively searching the latent spaces:

$$(\hat{m}^\phi_{\phi,\varphi}, \hat{m}^\varphi_{\phi,\varphi}) = \arg \min_{m^\phi_{\phi,\varphi}, m^\varphi_{\phi,\varphi}} D_1(\mathbf{x}_{\phi,m^\phi_{\phi,\varphi}}, \mathbf{x}_{\varphi,m^\varphi_{\phi,\varphi}}), \tag{5}$$

where we compute the latent variable pairs $(\hat{m}^\phi_{\phi,\varphi}, \hat{m}^\varphi_{\phi,\varphi})$ only when $y_\phi = y_\varphi$. **II:** With latent states $(\hat{m}^\phi_{\phi,\varphi}, \hat{m}^\varphi_{\phi,\varphi})_{y_\phi = y_\varphi}$, $\phi = 1, \ldots, N, \varphi = 1, \ldots, N$, we obtain the model parameter $\hat{\mathbf{M}}$ by solving the following optimization problem:

$$\begin{aligned} \hat{\mathbf{M}} = \arg\min_{\mathbf{M}} & \frac{1}{N_1} \sum_{\phi=1}^{N} \sum_{\substack{\varphi=1, \phi\neq\varphi,\\ y_\phi = y_\varphi}} D_1(\mathbf{x}_{\phi,\hat{m}^\phi_{\phi,\varphi}}, \mathbf{x}_{\varphi,\hat{m}^\varphi_{\phi,\varphi}}) \\ & + \frac{1}{N_2} \sum_{\phi=1}^{N} \sum_{\varphi=1, y_\phi \neq y_\varphi}^{N} D_2(\mathbf{x}_\phi, \mathbf{x}_\varphi) + \frac{\lambda}{2} \|\mathbf{M}\|_F^2 \\ & s.t.\ \mathbf{M} \succeq 0 \end{aligned} \tag{6}$$

where $\lambda$ is the regularization parameter, and is set as $10^{-5}$ in the implementation.

The optimization can be effectively computed via the gradient descent method, and the gradient for matrix $\mathbf{M}$ can be computed as

$$\begin{aligned} \dot{\mathbf{M}} = & \frac{1}{N_1} \sum_{\phi=1}^{N} \sum_{\substack{\varphi=1,\\ \phi\neq\varphi,\\ y_\phi = y_\varphi}}^{N} \frac{d D_1^{\phi,\varphi}(\cdot)}{d\mathbf{M}} \\ & + \frac{1}{N_2} \sum_{\phi=1}^{N} \sum_{\substack{\varphi=1,\\ y_\phi \neq y_\varphi}}^{N} \frac{d D_2^{\phi,\varphi}(\cdot)}{d\mathbf{M}} + \lambda \mathbf{M} \\ = & \frac{1}{N_1} \sum_{\phi=1}^{N} \sum_{\substack{\varphi=1,\\ \phi\neq\varphi,\\ y_\phi = y_\varphi}}^{N} \mathbf{W}_1^+(\mathbf{x}_{\phi,\hat{m}^\phi_{\phi,\varphi}}, \mathbf{x}_{\varphi,\hat{m}^\varphi_{\phi,\varphi}}) + \lambda \mathbf{M} \\ & - \frac{1}{N_2} \sum_{\phi=1}^{N} \sum_{\substack{\varphi=1,\\ y_\phi \neq y_\varphi}}^{N} \sum_{\substack{m^\phi_{\phi,\varphi} \in \mathcal{M}_\phi,\\ m^\varphi_{\phi,\varphi} \in \mathcal{M}_\varphi}} \mathbf{W}_1^-(\mathbf{x}_{\phi,m^\phi_{\phi,\varphi}}, \mathbf{x}_{\varphi,m^\varphi_{\phi,\varphi}}) \end{aligned} \tag{7}$$

$\dot{\mathbf{M}}$ denotes the gradient of matrix $\mathbf{M}$, $D_1^{\phi,\varphi}(\cdot)$ and $D_2^{\phi,\varphi}(\cdot)$ are the shorthands for $D_1(\mathbf{x}_{\phi,\hat{m}^\phi_{\phi,\varphi}}, \mathbf{x}_{\varphi,\hat{m}^\varphi_{\phi,\varphi}})$ and $D_2(\mathbf{x}_\phi, \mathbf{x}_\varphi)$, $\mathbf{W}_1^+(\cdot)$ and $\mathbf{W}_1^-(\cdot)$ are the abbreviations for

$$\begin{aligned} \boldsymbol{W}_1^+(\boldsymbol{\alpha}, \boldsymbol{\beta}) &= \frac{1}{1 + \exp(-f_1(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{M}) + 1)} (\boldsymbol{\alpha} - \boldsymbol{\beta})(\boldsymbol{\alpha} - \boldsymbol{\beta})^\top \\ \boldsymbol{W}_2^-(\boldsymbol{\alpha}, \boldsymbol{\beta}) &= \frac{1}{1 + \exp(f_1(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{M}) - 1)} (\boldsymbol{\alpha} - \boldsymbol{\beta})(\boldsymbol{\alpha} - \boldsymbol{\beta})^\top \end{aligned} \tag{8}$$

where $f_1(\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{M}) = (\boldsymbol{\alpha} - \boldsymbol{\beta})^\top \mathbf{M} (\boldsymbol{\alpha} - \boldsymbol{\beta})$.

In our implementation, we project the update of matrix $\mathbf{M}$ onto the positive semidefinite cone after each step. A similar derivation for the distance defined by the LADF method can be found in Appendix A.

The main computation load for the training process lies in computing the objective value (equation (2)) and the gradient (equation (7)), which can be rewritten via matrix manipulation (like [9] does in the implementation). Firstly, suppose we have $K$ pedestrian samples and each sample has $H$ features (each feature corresponds to one latent state). We concatenate these features together as $\mathbf{X} \in \mathbb{R}^{d,H\times K}$, and then we compute the distance matrix of the concatenated features as

$$\mathbf{D} = \mathbf{K} + \mathbf{K}^\top - 2\mathbf{X}^\top\mathbf{M}\mathbf{X} - 1, \tag{9}$$

where $\mathbf{K} = (\mathbf{M}\mathbf{X} \odot \mathbf{X})^\top \mathbf{1}_{d,1}\mathbf{1}_{1,H\times K}$. With the computed distance matrix, we obtain the value of our objective function as

$$\begin{aligned} obj = {} & \mathbf{1}_{1,H\times K}\left[\log(1+\exp(\mathbf{D}\odot\mathbf{Y}))\odot\mathbf{L}\right]1_{H\times K,1} \\ & + \frac{1}{2}\lambda\left\|\mathbf{M}\right\|_F^2, \end{aligned} \tag{10}$$

where the $(i, j)$-th element in $\mathbf{Y}$ indicates whether the $(i, j)$-th feature pair is negative ($\mathbf{Y}_{i,j} = -1$) or positive ($\mathbf{Y}_{i,j} = 1$), $\mathbf{L}$ is the weight matrix indicating the weight for each feature pair, $\odot$ is the hadmard product.

In computing the gradient with respect to $\mathbf{M}$, we first compute matrix $\mathbf{W}$ as

$$\mathbf{W} = \frac{1}{1+\exp(-\mathbf{D}\odot\mathbf{Y})}\odot\mathbf{L}\odot\mathbf{Y}, \tag{11}$$

and then we compute the gradient as

$$\begin{aligned} d\mathbf{M} = {} & \lambda\mathbf{M} + \Big[\mathbf{1}_{d,1}(\mathbf{1}_{1,H\times K}\mathbf{W}^\top)\odot\mathbf{X}\mathbf{X}^\top \\ & + \mathbf{1}_{d,1}(\mathbf{1}_{1,H\times K}\mathbf{W})\odot\mathbf{X}\mathbf{X}^\top\Big] - 2\mathbf{X}\mathbf{W}\mathbf{X}^\top. \end{aligned} \tag{12}$$

From the above equations, we can see that the algorithm complexity in the training process is $O(K^2H^2d^2)$. We exploit the MATLAB GPU Computing support for acceleration in this paper.

As to the algorithm complexity in the testing phase, we aim to compute the distance between every two pedestrians in probe and gallery sets. For convenience, we take the single-shot setting for example. Still, Suppose we have $K_p$ and $K_g$ samples in the probe and gallery sets respectively, with each sample having $H$ latent feature. Then we concatenate these features in probe and gallery set respectively, and obtain $\mathbf{X}_p \in \mathbb{R}^{d,H\times K_p}$ and $\mathbf{X}_g \in \mathbb{R}^{d,H\times K_g}$. The distance matrix between probe and gallery samples is computed as

$$\mathbf{D}_t = \mathbf{K}_1 + \mathbf{K}_2 - 2\mathbf{X}_p^\top\mathbf{M}\mathbf{X}_g, \tag{13}$$

where

$$\begin{aligned} \mathbf{K}_1 &= \mathrm{diag}(\mathbf{X}_p^\top\mathbf{M}\mathbf{X}_p)1_{1,H\times K_g} \\ \mathbf{K}_2 &= 1_{H\times K_p,1}\mathrm{diag}(\mathbf{X}_g^\top\mathbf{M}\mathbf{X}_g)^\top \end{aligned} \tag{14}$$

With the computed distance matrix $\mathbf{D}_t \in \mathbb{R}^{H\times K_p,H\times K_g}$, we can easily reshape it as a $(H^2)\times(K_pK_g)$ matrix (the same operation as they do in the convolution layer in Caffe with kernel stride and size $H \times H$ ). Then we can obtain the minimum value for each column, and obtain a row vector with size $1 \times K_pK_g$. This row vector is then reshaped to final distance matrix with size $K_p \times K_g$. Therefore, in the test phase, the algorithm complexity is $O(K_pK_gH^2d^2)$.

## III. Person Re-Identification With Latent Variables

In this section, we present how to exploit latent variables for the re-identification problem. The adopted latent variables are used in three patch-based models, which mainly focus on modeling vertical misalignments, horizontal misalignments and leg postures. Here, we use the mahalanobis distance metric to present the proposed models, and we also note that it is not difficult to extend our algorithm by using the locally-adaptive decision function (LADF) method.

### A. Model Misalignments

In realistic scenarios, due to the imperfect performance of the human detector, there inevitably exist considerable vertical misalignments between images of the same pedestrian (*e.g.*, the first and second examples on the PRID450S dataset, the second and third examples on the iLIDS dataset in Figure 1).

To address this issue, in this work, we divide the pedestrian image of camera $a$ into $J_V$ horizontal patches. For each patch $j$ in camera $a$, we match it with a number of candidate patch regions in camera $b$, where a latent variable $v_b^j$ is exploited as the candidate region index. We use $\mathbf{e}_j^a$ to denote the feature extracted in the patch $j$ in camera $a$, and $\mathbf{e}^b_{v_b^j}$ to denote the feature vector of the $v_b^j$-th candidate region. The feature $\mathbf{e}^b_{v_b^j}$ can be calculated as

$$\begin{aligned} \mathbf{e}^b_{v_b^j} &= \Phi(\mathbf{N}_{v_b^j}), \quad v_b^j = 1,\ldots,\mathrm{length}(\mathbf{N}) \\ \mathbf{N} &= [\theta_j - hh,\ldots,\theta_j + hh], \end{aligned} \tag{15}$$

where $\mathbf{N}$ is a vector containing row coordinates, $\mathbf{N}_{v_b^j}$ denotes the $v_b^j$-th element of the vector $\mathbf{N}$, and $\Phi(\mathbf{N}_{v_b^j})$ denotes the feature extracted in the patch centred in the $\mathbf{N}_{v_b^j}$-th row. $\theta_i$ stands for the row centre coordinate of the patch $j$, and $hh$ is the local search range for candidates in camera $b$ ($hh$ is set as 6 pixels in our experiments by default).

Then we define the distance between pedestrians $I_a$ and $I_b$ for the local patch $j$ as

$$d_V^j(I_a, I_b) = \min_{v_b^j}\left[(\mathbf{e}_j^a - \mathbf{e}^b_{v_b^j})^\top\mathbf{M}_V^j(\mathbf{e}_j^a - \mathbf{e}^b_{v_b^j}) + \left|\theta_j - \mathbf{N}_{v_b^j}\right|\right], \tag{16}$$

where the matrix $\mathbf{M}_V^j$ is the learned model parameter and the second term provides a spatial constraint for encouraging the patch $j$ to be matched with a spatially adjacent candidate.

Finally, for the vertical latent variable model, the distance $d_V(I_a, I_b)$ can be determined by,

$$d_V(I_a, I_b) = \sum_{j=1}^{J_V} d_V^j(I_a, I_b), \tag{17}$$

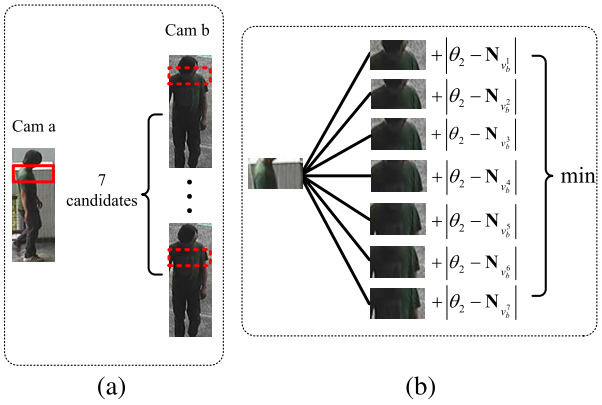



Fig. 2. An example on how distance is computed between two pedestrians considering vertical misalignments. (a) We use patch $e_j^a$ to match a number of candidate patches. (b) We compute the distance between two pedestrians by simultaneously considering the distance between patches and spatial constraint.


Fig. 3. An example showing that features extracted in the rectangular blocks (red rectangular blocks for the probe images and blue rectangular blocks for the gallery images) are consistent across camera views.

where $J_V$ is the number of vertical patches, and is set as 6 in our experiment. The matching process is illustrated in Figure 2.

Besides vertical misalignments, horizontal misalignments are usually caused by various orientations of the pedestrians to be matched, which makes the extracted features of two cameras less consistent. We note that this problem can be better handled if features are extracted in the aligned sub-regions of two images. Figure 3 provides some examples where features extracted in the rectangular blocks are more consistent across camera views.

For pedestrians $I_a$ and $I_b$, we divide the images into $J_H$ horizontal patches, and obtain $J_H$ patch pairs. For each patch pair, our aim is to compute the distance between the well-matched sub-regions defined on the patches. Specially, we adopt the latent variables $h_a^j$ and $h_b^j$ to denote the regions for feature extraction in the $j$-th patch pair for both cameras, and denote the extracted features as $\mathbf{o}^a_{h_a^j}$ and $\mathbf{o}^b_{h_b^j}$.[1] The distance between $I_a$ and $I_b$ in the $j$-th patch pair is calculated as

$$d_H^j(I_a, I_b) = \min_{h_a^j, h_b^j} (\mathbf{o}^a_{h_a^j} - \mathbf{o}^b_{h_b^j})^\top \mathbf{M}_H^j (\mathbf{o}^a_{h_a^j} - \mathbf{o}^b_{h_b^j}). \quad (18)$$

[1]By default, we sample 5 candidate regions for each patch and the width of each candidate region is $\frac{5}{8}$ of the entire patch width.

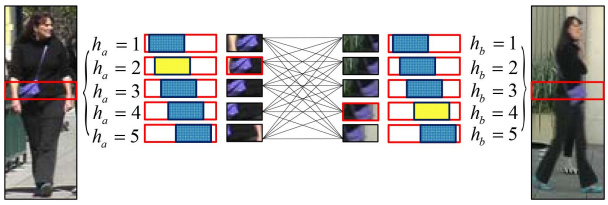



Fig. 4. An example on how distance is computed between two pedestrians considering horizontal misalignments. In this example, the second and fourth candidate patches in the two cameras are used for distance computation, as they have the most similar appearance.


Fig. 5. An illustration on how rotating rectangle blocks in the leg regions are defined.

Figure 4 presents an example to illustrate how this distance is calculated considering horizontal misalignments.

Finally, the distance in the horizontal latent variable model can be defined as

$$d_H(I_a, I_b) = \sum_{j=1}^{J_H} d_H^j(I_a, I_b), \quad (19)$$

where $J_H$ is also set to 6 in this work.

### *B. Model Leg Posture*

We note that the lower bodies of the detected pedestrian examples usually have very large variations even for the same person, where the variation of leg postures is the most important issue. Thus, in this subsection, we calculate the distance between lower bodies of two pedestrians by implicitly estimating the leg postures. To be specific, we adopt a rotating rectangle block to roughly capture the leg posture for each leg, and sample 9 candidate regions for each leg with 9 angles (between the rectangle block and the vertical axis of the image) and one pivoting point. The candidate angles for the left and right legs are respectively $(-30°, -25°, -20°, -15°, -10°, -5°, 0°, 5°, 10°)$ and $(-10°, -5°, 0°, 5°, 10°, 15°, 20°, 25°, 30°)$. The distance between the pivoting points of two legs is 1/6 of the image width. Figure 5 intuitively demonstrates how the rectangle blocks are defined in this work.

For distance computation, we use latent variables $\mathbf{p}_a = [p_a^{\text{left}}, p_a^{\text{right}}]$ and $\mathbf{p}_b = [p_b^{\text{left}}, p_b^{\text{right}}]$ to denote the angles for both left and right legs in cameras $a$ and $b$, and use variables $\mathbf{g}^a_{p_a^{\text{left}}}$, $\mathbf{g}^a_{p_a^{\text{right}}}$, $\mathbf{g}^b_{p_b^{\text{left}}}$ and $\mathbf{g}^b_{p_b^{\text{right}}}$ to denote features extracted in regions specified by these latent variables. Then the distance

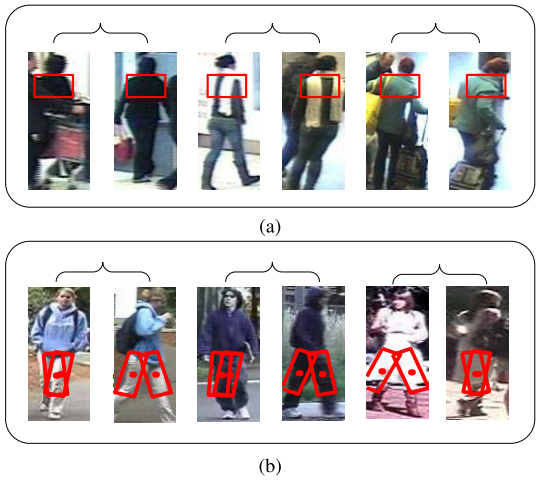


Fig. 6. Some examples of the estimated consistent regions across cameras. (a) The specified sub-regions by the latent variables $h_a^2$ and $h_b^2$. (b) The specified sub-regions by the latent variables $p_a^{\text{left}}$, $p_a^{\text{right}}$, $p_b^{\text{left}}$ and $p_b^{\text{right}}$.

between lower bodies of $I_a$ and $I_b$ can be defined as

$$d_P(I_a, I_b) = \min_{p_a^{\text{left}}, p_b^{\text{left}}} (\mathbf{g}^a_{p_a^{\text{left}}}, \mathbf{g}^b_{p_b^{\text{left}}})^\top \mathbf{M}_P^{\text{left}} (\mathbf{g}^a_{p_a^{\text{left}}}, \mathbf{g}^b_{p_b^{\text{left}}}) + \min_{p_a^{\text{right}}, p_b^{\text{right}}} (\mathbf{g}^a_{p_a^{\text{right}}}, \mathbf{g}^b_{p_b^{\text{right}}})^\top \mathbf{M}_P^{\text{right}} (\mathbf{g}^a_{p_a^{\text{right}}}, \mathbf{g}^b_{p_b^{\text{right}}}) \tag{20}$$

where we compute the matching distance for left and right legs respectively, and sum up these two distances for further processing. In most cases, regions specified by variables $p_a^{\text{left}}$ (or $p_a^{\text{right}}$, $p_b^{\text{left}}$, $p_b^{\text{right}}$) are leg regions. The underlying reason is that leg regions usually experience smaller variations than the backgrounds across camera views. In Figure 6, we show that consistent features can be obtained in the estimated sub-regions.

### C. The Overall Distance for the Re-Identification Problem

Based on the above-mentioned discussions, we define our final distance for the re-identification problem as

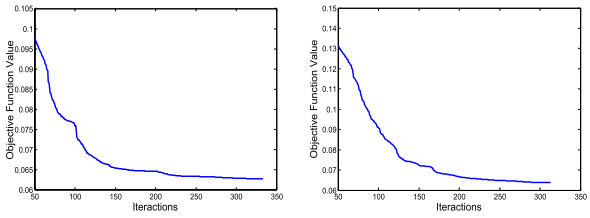
$$d(I_a, I_b) = \gamma_0 d_0(I_a, I_b) + \gamma_V d_V(I_a, I_b) + \gamma_H d_H(I_a, I_b) + \gamma_P d_P(I_a, I_b), \tag{21}$$

where $\gamma_0$, $\gamma_V$, $\gamma_H$ and $\gamma_P$ are the trade-off parameters for different distances, which are set as 1, 1, 0.5, 0.2 in the experiments. $d_0(\cdot)$ denotes the distance computed between features in the holistic target (it can be viewed as a baseline method in this work), $d_V(\cdot)$, $d_H(\cdot)$ and $d_P(\cdot)$ stand for the vertical, horizontal and posture latent variable models. We incorporate the holistic model in our method as it has good complementary with the local models. The effectiveness of the holistic model can be found in the experiment section (Table X).

The holistic distance $d_0(I_a, I_b)$ can be directly computed as

$$d_0(I_a, I_b) = (\mathbf{x}_0^a - \mathbf{x}_0^b)^\top \mathbf{M}_0 (\mathbf{x}_0^a - \mathbf{x}_0^b), \tag{22}$$

where $\mathbf{x}_0^a$ and $\mathbf{x}_0^b$ denote the holistic features for $I_a$ and $I_b$, and $\mathbf{M}_0$ is the metric matrix. The computations of $d_V(I_a, I_b)$, $d_H(I_a, I_b)$ and $d_P(I_a, I_b)$ have been detailed in the previous sections.




(a) (b)

Fig. 7. The objective function values with increasing iterations in the training process for left (a) and right legs (b).

### D. Model Learning

In this subsection, we introduce the model learning process based on the mahalanobis distance metric. The learning process is not difficult to be extended into the LADF method. We note that the parameters to be optimized include $\{\mathbf{M}_0\}$, $\{\mathbf{M}_V^j\}_{j=1}^{J_V}$, $\{\mathbf{M}_H^j\}_{j=1}^{J_H}$, and $\{\mathbf{M}_P^{\text{left(right)}}\}$.

- **Learning $\{\mathbf{M}_0\}$.** The learning process for this parameter is straightforward. We compute the features of two pedestrians and learn the parameters merely with equation (6) as the holistic features have no relations with latent variables.

- **Learning $\{\mathbf{M}_V^j\}_{j=1}^{J_V}$, $\{\mathbf{M}_H^j\}_{j=1}^{J_H}$ and $\{\mathbf{M}_P^{\text{left(right)}}\}$.** We exploit the formula of equation (2) for model learning, and solve it with the iterative optimization manner as in equation (5) and (6). Figure 7 provides an example to demonstrate the variations of the objective function value in the iteration process, which also illustrates the convergence of the proposed metric learning algorithm.

## IV. Experiments

*Pedestrian Representation:* In this work, we extract the same kinds of features for our holistic target and the flexible patches. Specially, we use five kinds of features: 8-bin histogram for the RGB color space, 8-bin histogram for the HSV color space, 32-bin histogram for the LAB color space, 16-bin histogram for YCbCr color space, and the Scale Invariant Local Ternary Pattern (SILTP) descriptor computed via [27] (we use $8 \times 16$ overlapping blocks with stride $8 \times 8$). For the local patch, we directly compute the histograms of different color (texture) models in it. In the holistic target, we concatenate the color features of six equal sized horizontal patches as in [28]. Descriptors from both the foreground image and the holistic image are used as is done in [29]. We perform the feature dimension reduction via the PCA method, and retain 100 dimensions. In addition to the above mentioned low-level features, we also report performance of our method based on a recently proposed feature called LOMO [30]. For the LOMO feature, we perform feature dimension reduction for the holistic target following [30], while for the patches, we exploit the PCA method for dimension compressing.

*Evaluation Protocol:* In this work, we follow the evaluation protocol in [5]. For different datasets, we evenly partition the training and testing samples. Images in camera A are regarded

as probe images, and images in camera B are regarded as gallery ones. We match each probe image to one of the gallery images. Experiments on each dataset are repeated 10 times (20 times for the CUHK03 dataset), and then the average performance is reported to overcome the influence of randomness. For easier comparison with the published results (some algorithms only have reported results at some discrete ranks), we only report the cumulated matching accuracy at rank 1, 5, 10 and 20, and do not report the entire CMC curve [31].

### A. Overall Performance

In this work, we exploit seven publicly available datasets (including VIPeR [5], iLIDS [32], QUML GRID [33], PRID450S [34], PRID2011 [35], CUHK01 [1] and CUHK03 [2])[2] to validate the effectiveness of the proposed method. The detailed introductions and evaluations are as follows.

*VIPeR Dataset:*[3] The VIPeR dataset contains 632 pedestrians, one of which has one image (with size 128×48 pixels) for each camera view. This dataset is captured outdoor, and thus has complex illumination variations. Apart from illumination variations, it also contains considerable viewpoint variations, where the viewpoint changes between two cameras are usually larger than 90 degrees (Figure 1(a)). On this dataset, we compare the performance of the proposed method against several state-of-the-art algorithms, including eBiCov [7], PCCA [14], KISSME [18], LADF [9], eSDC [19], SalMatch [20], SDALF [6], PRDC [11], aPRDC [8], LF [36], Mid-Filter [21] and ECM [29], LOMO+XQDA [30], MetricEnsemble [37]. The results are demonstrated in Table I.

In the experiment, we test the effectiveness of our method by exploiting the mahalanobis distance metric and the LADF metric respectively. Note that as the LOMO+XQDA and MetricEnsemble methods exploit very strong features (*e.g.*, LOMO and CNN features), we also provide the experimental results of our method exploiting the same LOMO features for fair comparison. The experimental results are reported in two groups: (1) we make comparisons among algorithms with weak features; (2) we make comparisons between our method (with mahalanobis distance metric) exploiting LOMO features, the LOMO+XQDA method and the MetricEnsemble method. As Table I shows, the proposed method equipped with the LADF metric achieves a 41.2% rank-1 performance, and 86.0% rank-10 performance, which improves the second best one by 3.0% and 7.7% among algorithms with weak features. In group 2, it seems that the EnsembleMetric has the best performance. However, this method combines 6 kinds of features and several metrics with different parameters. We show that our method has comparable (or better) performance, when we combine our results with the LADF method [9].

[2]For the CUHK01 and CUHK03 datasets, we exploit three candidates for each patch model considering the memory of our graphics card (we use Nvidia Titan X for acceleration).

[3]https://vision.soe.ucsc.edu/node/178

TABLE I

THE TOP MATCHING RATES (IN PERCENTAGE) OF DIFFERENT METHODS ON THE VIPeR DATASET. THE BEST PERFORMANCE IS HIGHLIGHTED IN BOLD FONTS. NOTE THAT, WHEN OUR METHOD IS COMBINED WITH THE LADF METHOD, THE RANK-1 MATCHING RATE CAN BE 50.4%

| Method | Rank 1 | Rank 5 | Rank 10 | Rank 20 |
|---|---|---|---|---|
| eBicov | 20.7 | 42.0 | 56.2 | 68.0 |
| PCCA | 19.3 | 48.9 | 64.9 | 80.3 |
| KISSME | 19.6 | 48.0 | 62.2 | 77.0 |
| LADF | 29.3 | 61.0 | 76.0 | 86.2 |
| eSDC | 26.3 | 46.6 | 58.9 | 72.8 |
| SalMatch | 30.2 | 52.3 | 65.5 | 79.1 |
| SDALF | 19.9 | 38.9 | 49.4 | 65.7 |
| PRDC | 15.7 | 38.4 | 53.9 | 70.1 |
| aPRDC | 16.1 | 37.7 | 51.0 | 66.0 |
| LF | 24.2 | 52.0 | 67.1 | 82.0 |
| Mid-Filter | 29.1 | 52.3 | 66.0 | 79.9 |
| ECM | 38.2 | 67.2 | 78.3 | 87.9 |
| $\text{Ours}_{\text{LADF}}$ | **41.2** | **74.4** | **86.0** | **94.1** |
| $\text{Ours}_{\text{M}}$ | 39.6 | 72.0 | 84.5 | 92.5 |
| LOMO+XQDA | 40.0 | 68.1 | 80.5 | 91.1 |
| MetricEnsemble | 45.9 | 77.5 | **88.9** | **95.8** |
| $\text{Ours}_{\text{LOMO}}$ | 44.3 | 74.4 | 84.7 | 92.7 |
| $\text{Ours}_{\text{LOMO}}$ + LADF | **50.4** | **80.5** | 88.7 | 95.0 |

TABLE II

THE TOP MATCHING RATES (IN PERCENTAGE) OF DIFFERENT METHODS ON THE iLIDS DATASET. THE BEST PERFORMANCE IS HIGHLIGHTED IN BOLD FONTS

| Method | Rank 1 | Rank 5 | Rank 10 | Rank 20 |
|---|---|---|---|---|
| PCCA | 20.6 | 48.1 | 64.8 | 82.0 |
| LFDA | 32.3 | 59.0 | 72.3 | 85.8 |
| MFA | 31.2 | 55.4 | 68.3 | 81.8 |
| LADF | 21.1 | 50.7 | 66.7 | 83.3 |
| kLFDA | 37.6 | 66.4 | 78.6 | 90.1 |
| KISSME | 29.9 | 56.2 | 69.1 | 82.6 |
| $\text{Ours}_{\text{LADF}}$ | 39.0 | 68.3 | 81.7 | 90.3 |
| $\text{Ours}_{\text{M}}$ | **41.7** | **69.5** | **82.2** | **91.2** |
| LOMO+XQDA | 43.0 | 66.8 | 78.2 | 88.2 |
| MetricEnsemble | **50.3** | – | – | – |
| $\text{Ours}_{\text{LOMO}}$ | 46.2 | **70.2** | **80.7** | **91.3** |

*iLIDS Dataset:*[4] The iLIDS dataset contains 119 pedestrians captured in 2 to 8 cameras in the airport. This dataset is challenging for considerable spatial misalignments (Figure 1(d)). On this dataset, we resize all the images to 128 × 48 pixels for normalization, and compare our method against 8 state-of-the-art algorithms including PCCA [14], LFDA [36], MFA [38], LADF [9], kLFDA [39], KISSME [18], LOMO+XQDA [30] and MetricEnsemble [37]. The performance comparison between our method and the 8 state-of-the-art methods are demonstrated in Table II.

It can be seen from Table II that our method with LADF metric achieves a 39.0% rank-1 performance, which outperforms the second best one by 1.4% among algorithms with weak features. By exploiting the mahalanobis distance metric, the proposed method improves the second best algorithm by 4.1% at rank-1 matching rate. In group 2, we make comparisons with the LOMO+XQDA and MetricEnsemble methods, among which our method has comparable performance.

[4]https://www.gov.uk/guidance/imagery-library-for-intelligent-detection-systems

TABLE III
THE TOP MATCHING RATES (IN PERCENTAGE) OF DIFFERENT METHODS ON THE GRID DATASET. THE BEST PERFORMANCE IS HIGHLIGHTED IN BOLD FONTS

| Method | Rank 1 | Rank 5 | Rank 10 | Rank 20 |
|---|---|---|---|---|
| PRDC | 9.7 | 22.0 | 33.0 | 44.3 |
| RankSVM | 10.2 | 24.6 | 33.3 | 43.7 |
| MRank-PRDC | 11.1 | 26.1 | 35.8 | 46.6 |
| MRank-RankSVM | 12.2 | 27.8 | 27.8 | 46.6 |
| PolyMap | 16.3 | 35.8 | 46.0 | 57.6 |
| $\text{Ours}_{\text{LADF}}$ | 20.5 | 39.4 | 50.2 | 66.0 |
| $\text{Ours}_{\text{M}}$ | **26.8** | **47.9** | **58.2** | **69.7** |
| LOMO+XQDA | 16.6 | 33.8 | 41.8 | 52.4 |
| $\text{Ours}_{\text{LOMO}}$ | **19.8** | **35.6** | **46.5** | **58.1** |

*GRID Dataset:*[5]The QMUL underGround Re-IDentification (GRID) dataset is also a very challenging dataset, which is captured in 8 disjoint cameras in the subway station. This dataset contains large pose and illumination variations, and images are usually in low resolution (Figure 1(b)). In addition to the 225 pedestrian pairs, 775 interference pedestrians are included in the gallery set, which makes the performance of different algorithms on this dataset lower than that on other datasets. Several algorithms including PRDC [11], RankSVM [28], LCRML [40], MRank-RankSVM [41], MtMCML [42], PolyMap [43] and LOMO+XQDA [30] report their results on this dataset. In this experiment, we evaluate our methods in comparison with these state-of-the-art methods, and report the performance in Table III.

With the traditional features, our algorithm (with the LADF metric) has very good performance in terms of matching rates at rank 1, 10 and 20, which outperforms the second best one by 4.2%, 4.2% and 8.4% respectively. By exploiting the mahalanobis distance metric, the performance of our method can be further improved, which improves the state-of-the-art performance by 10.5% at rank-1 matching rate. We also exploit the LOMO features in the implementation, and our algorithm achieves a 19.8% rank-1 matching rate, which is 3.2% higher than LOMO+XQDA.

*PRID450S Dataset:*[6] The PRID450S dataset has 450 pedestrians captured in two cameras, which results in 900 images in total. This dataset is challenging for the large changes in viewpoint, pose and illumination, and persons in two cameras are significantly different. As a newly constructed dataset, only a few of methods report their results on it, which are respectively KISSME [18], EIML [44], SCNCN [45], ECM [29] and LOMO+XQDA [30]. We compare our method with these algorithms, and summarize the performance of different algorithms in Table IV.

Among the algorithms with traditional features, our method (equipped with the mahalanobis or LADF metric) achieves 47.8% at rank 1 and 82.8% at rank 10 with significant improvement over the compared methods. When we exploit the LOMO features, our method can be further enhanced. We improve the LOMO+XQDA method by 3.3% at rank-1 matching rate.

[5]http://www.eecs.qmul.ac.uk/~ccloy/downloads_qmul_underground_reid.html

[6]http://lrs.icg.tugraz.at/download.php

TABLE IV
THE TOP MATCHING RATES (IN PERCENTAGE) OF DIFFERENT METHODS ON THE PRID450S DATASET. THE BEST PERFORMANCE IS HIGHLIGHTED IN BOLD FONTS

| Method | Rank 1 | Rank 5 | Rank 10 | Rank 20 |
|---|---|---|---|---|
| KISSME | 33.0 | – | 71.0 | 79.0 |
| EIML | 35.0 | – | 68.0 | 77.0 |
| ECM | 41.9 | 66.3 | 76.9 | 84.9 |
| SCNCD | 41.6 | 68.9 | 79.4 | 87.8 |
| $\text{Ours}_{\text{LADF}}$ | **47.8** | **74.7** | **82.8** | **90.9** |
| $\text{Ours}_{\text{M}}$ | **47.8** | 73.7 | **82.8** | 90.1 |
| LOMO+XQDA | 61.2 | 84.8 | 90.9 | 95.1 |
| $\text{Ours}_{\text{LOMO}}$ | **64.5** | **85.7** | **92.1** | **96.0** |

TABLE V
THE TOP MATCHING RATES (IN PERCENTAGE) OF DIFFERENT METHODS ON THE PRID2011 DATASET. THE BEST PERFORMANCE IS HIGHLIGHTED IN BOLD FONTS

| Method | Rank 1 | Rank 5 | Rank 10 | Rank 20 |
|---|---|---|---|---|
| RPLM | 15.0 | 32.0 | 42.0 | 54.0 |
| $\text{Ours}_{\text{LADF}}$ | **16.2** | 34.0 | 44.4 | 59.5 |
| $\text{Ours}_{\text{M}}$ | 15.2 | **36.1** | **48.3** | **60.4** |
| LOMO+XQDA | 26.7 | **49.9** | **61.9** | **73.8** |
| MetricEnsemble | 17.9 | 39.0 | 50.0 | 62.0 |
| $\text{Ours}_{\text{LOMO}}$ | **27.8** | 48.4 | 59.5 | 72.7 |

TABLE VI
THE TOP MATCHING RATES (IN PERCENTAGE) OF DIFFERENT METHODS ON THE CUHK01 DATASET. THE BEST PERFORMANCE IS HIGHLIGHTED IN BOLD FONTS

| Method | Rank 1 | Rank 5 | Rank 10 | Rank 20 |
|---|---|---|---|---|
| ITML | 16.0 | 35.22 | 45.6 | 59.8 |
| GenericMetric | 20.0 | 43.6 | 56.0 | 69.3 |
| SalMatch | 28.5 | 45.9 | 55.7 | 68.0 |
| MidFilter | 34.3 | 55.1 | 65.0 | 74.9 |
| $\text{Ours}_{\text{LADF}}$ | **58.0** | **83.7** | **90.5** | **94.9** |
| $\text{Ours}_{\text{M}}$ | 56.3 | 82.0 | 89.3 | 94.3 |
| LOMO+XQDA | 63.2 | 83.9 | 90.0 | 94.4 |
| MetricEnsemble | 53.4 | 76.4 | 84.4 | 90.5 |
| $\text{Ours}_{\text{LOMO}}$ | **65.0** | **85.6** | **91.1** | **95.1** |

*PRID2011 Dataset:* The PRID2011 dataset is similar with the PRID450S dataset with both single-shot and multiple-shot experimental settings, while we exploit the single-shot setting for performance report. In this dataset, camera *a* contains 385 pedestrians and camera *b* contains 749 pedestrians. This dataset is very challenging as only 200 pedestrians appear in both cameras, and the other pedestrians can be regarded as distractions. As few algorithms report their performance on this dataset, we only compare the proposed method with RPLM [17], LOMO+XQDA [30] and MetricEnsemble [37]. We summarize the results of these methods in Table V.

Overall, our method has comparable performance with the LOMO+XQDA method, and has far better performance than the MetricEnsemble method. This dataset also shows the effectivenss of the proposed method.

*CUHK01 Dataset:* The CUHK01 dataset has 3884 images for 971 pedestrians from two cameras, and thus it has two images for each pedestrian per camera. In this dataset, we compare the proposed method with ITML [46], GenericMetric [13], MidFilter [21], SalMatch [20], LOMO+XQDA [30], and MetricEnsemble [37]. The results can be seen in Table VI.

(a) (b)

Fig. 8. Sampled re-identification results for the VIPeR and iLIDS datasets. For each dataset, we present the successful examples in the first two rows and present the failure cases in the last row. We mark the probe image and corrected matched images with brown and red rectangle boxes respectively. (a) VIPeR dataset. (b) iLIDS dataset.

TABLE VII
RUNNING COST OF THE PROPOSED METHOD

| | VIPeR | iLIDS | PRID450S | GRID | PRID2011 | CUHK01 | CUHK03 |
|---|---|---|---|---|---|---|---|
| Phase | Time (seconds) | Time (seconds) | Time (seconds) | Time (seconds) | Time (seconds) | Time (seconds) | Time (seconds) |
| Training | 635 | 21 | 264 | 208 | 62 | 5335 | 15634 |
| Testing | 0.4 | 0.02 | 0.2 | 0.6 | 0.3 | 3.0 | 0.04 |

Even though with weak features, our method equipped with LADF metric achieves comparative performance against the MetrcEnsemble method, which has exploited the strong CNN features. In addition, by exploiting the LOMO feature, we achieve a 65% rank-1 matching rate, which is 1.8% higher than the LOMO+XQDA method with the same feature.

*CUHK03 Dataset:* The CUHK03 dataset is an extension of the CUHK01 dataset, which contains 13164 images from 1360 pedestrians. This dataset provides both manually labelled bounding boxes and bounding boxes detected by an automatic detector [47]. In this dataset, we only conduct experiments based on the automatically detected bounding boxes, which is more challenging than experiments based on the manually labelled bounding boxes. We follow the experimental settings of [2], and run the experiments for 20 random splits for stable performance analysis. We compare the proposed methods with DeepReID [2], KISSME [18], LDML [48], eSDC [19], LMNN [49], ITML [46], LOMO+XQDA [30]. The performance comparison is demonstrated in Table VIII.

The comparisons in Table VIII show that the proposed method[7] has better performance than the LOMO+XQDA method with the same feature. In addition, exploiting the traditional color and texture features, the proposed method also has very good performance compared to the previous methods.

In Figure 8, we show the ranked observation results given the query images for two example datasets.

*Running Cost:* The running cost of the proposed method on the reported datasets is presented in Table VII (based on the mahalanobis metric). It can be seen that our method is very efficient in the testing process.

[7]Different from other datasets, we exploit the LADF metric here as it contains more model parameters and is more suitable for the large scale dataset.

TABLE VIII
THE TOP RANK-1 MATCHING RATES (IN PERCENTAGE) OF DIFFERENT METHODS ON THE CUHK03 DATASET. THE BEST PERFORMANCE IS HIGHLIGHTED IN BOLD FONTS

| Method | Matching Rate (%) |
|---|---|
| ITML | 5.14 |
| DeepReID | 19.9 |
| KISSME | 11.7 |
| LDML | 10.9 |
| eSDC | 7.7 |
| LMNN | 6.3 |
| ITML | 5.1 |
| $\text{Ours}_{\text{LADF}}$ | **39.1** |
| $\text{Ours}_{\text{M}}$ | 36.1 |
| LOMO+XQDA | 46.3 |
| $\text{Ours}_{\text{LOMO}}$ | **48.1** |

## B. Analysis of the Proposed Method

In this subsection, we use the method based on the LADF metric for analysis by default, and the conclusions are applicable to the method based on mahalanobis distance metric.

*1) Evaluation on the Parameter Sensitivity:* In this paper, there are four major parameters, which are respectively $\gamma_0$, $\gamma_V$, $\gamma_H$ and $\gamma_P$. We test their sensitivity by varying their values and observe the influence they have. Without loss of generality, we take the iLIDS dataset for analysis. The rank 1, 5, 10 and 20 matching rates with the varying parameter settings are presented in Figure 9. It can be seen that in a relatively large value range, the parameter settings do not drastically influence the re-identification performance.

*2) Evaluation on the Latent Variables:* In this subsection, we conduct experiments to test the effectiveness of latent variables, where both the LADF metric and the mahalanobis metric are exploited. In the reported two datasets, we use "LADF" and "Mahalanobis" to represent the methods that do not consider latent variables (*i.e.*, each patch is

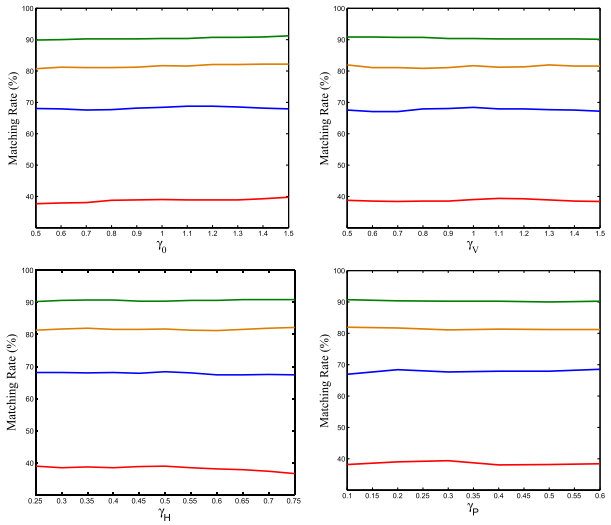



Fig. 9. Re-identification performance with changing parameter settings. In this figure, we show the rank 1, 5, 10 and 20 matching rates in red, blue, brown and green lines respectively.

TABLE IX
COMPARISONS BETWEEN THE METHODS WITH AND WITHOUT LATENT VARIABLES

| PRID2011 dataset | | | | |
|---|---|---|---|---|
| Method | Rank 1 | Rank 5 | Rank 10 | Rank 20 |
| Ours_no_metric | 9.9 | 25.9 | 36.4 | 43.4 |
| LADF | 14.0 | 33.8 | 44.1 | 56.5 |
| LADF_latent | 16.2 | 34.0 | 44.4 | 59.5 |
| Mahalanobis | 12.3 | 28.9 | 39.0 | 54.3 |
| Mahalanobis_latent | 15.2 | 36.1 | 48.3 | 60.4 |
| GRID dataset | | | | |
| Method | Rank 1 | Rank 5 | Rank 10 | Rank 20 |
| Ours_no_metric | 16.2 | 33.8 | 40.6 | 50.2 |
| LADF | 18.4 | 36.9 | 47.4 | 61.1 |
| LADF_latent | 20.5 | 39.4 | 50.2 | 66.0 |
| Mahalanobis | 21.0 | 43.4 | 53.7 | 65.6 |
| Mahalanobis_latent | 26.8 | 47.9 | 58.2 | 69.7 |

assigned to a fixed state), and exploit "LADF_latent" and "Mahalanobis_latent" to represent the methods wherein latent variables are considered. The results are presented in Table IX. With the LADF metric, our method "LADF_latent" improves the baseline method "LADF" by 2.2% and 2.1% at rank 1 matching rates in two datasets, while our method "Mahalanobis latent" improves the baseline method "Mahalanobis" by 2.9% and 5.8% at rank 1 matching rates.

*3) Evaluation on the Flexible Patches:* In this paper, altogether three kinds of flexible patches are proposed to address vertical misalignments, horizontal misalignments and posture variations, which are respectively named as "vPatch", "hPatch" and "pPatch" for simplicity. We choose the algorithm that only considers the holistic features as the baseline method, and then verify the effectiveness of the flexible patches by comparing the following six variant algorithms: (1) "baseline" (*i.e.*, $d_0$), (2) "baseline+vPatch" (*i.e.*, $d_0 + d_V$), (3) "baseline+hPatch" (*i.e.*, $d_0 + d_H$), (4) "baseline+hPatch+vPatch" (*i.e.*, $d_0 + d_V + d_H$), (5) "hPatch+vPatch+pPatch" (*i.e.*, $d_V + d_H + d_P$), and (6) "Ours" (*i.e.*, "baseline+hPatch+vPatch+pPatch"). In Table X, we show the performance of these variant algorithms on the iLIDS and PRID450S datasets.

TABLE X
THE TOP MATCHING RATES (IN PERCENTAGE) OF DIFFERENT VARIANTS ON THE PRID450S AND iLIDS DATASETS

| iLIDS dataset | | | | |
|---|---|---|---|---|
| Method | Rank 1 | Rank 5 | Rank 10 | Rank 20 |
| $d_0$ (baseline) | 23.0 | 50.0 | 65.7 | 82.2 |
| $d_0 + d_V$ | 33.2 | 62.3 | 76.0 | 88.5 |
| $d_0 + d_H$ | 34.8 | 64.2 | 78.3 | 88.2 |
| $d_0 + d_V + d_H$ | 37.0 | 65.5 | 80.5 | 90.7 |
| $d_V + d_H + d_P$ | 36.5 | 67.7 | 81.0 | 90.0 |
| Ours | 39.0 | 68.3 | 81.7 | 90.3 |
| PRID450S dataset | | | | |
| Method | Rank 1 | Rank 5 | Rank 10 | Rank 20 |
| $d_0$ (baseline) | 32.9 | 62.7 | 74.7 | 85.5 |
| $d_0 + d_V$ | 45.5 | 71.8 | 81.4 | 90.0 |
| $d_0 + d_H$ | 41.5 | 70.3 | 80.3 | 89.0 |
| $d_0 + d_V + d_H$ | 46.9 | 73.1 | 82.6 | 90.6 |
| $d_V + d_H + d_P$ | 43.5 | 68.8 | 79.4 | 88.5 |
| Ours | 47.8 | 73.8 | 82.8 | 90.9 |

By comparing the performance of the first three rows in each sub-table, we show the effectiveness of the proposed "vPatch" and "hPatch". Specially, on the reported datasets, "vPatch" improves the baseline method by 10.2% and 12.6% respectively at rank-1 matching rates, while 'hPatch" improves the baseline method by 11.8% and 8.6% at rank-1 matching rates. By comparing the fourth row and the last row in each sub-table, we also illustrate the effectiveness of the proposed "pPatch", which improves the performance by 2% and 0.9% at rank-1 matching rates. In addition, by comparing the last two rows, we can see that the holistic model also contribute to the re-identification performance.

## V. CONCLUSION

In this paper, we represent a pedestrian as the mixture of a holistic model and a number of flexible models with latent variables. These flexible models are based on patches, and consist of two kinds of model parameters, *i.e.*, metric matrices and latent variables indicating spatial positions or orientations of the patches. Given two pedestrians, we calculate the distance between them by summing the distances of the holistic model and the flexible models. The adopted latent variables could capture the spatial or posture deformations between the pedestrians compared, and thus deformations can be better handled in our method. In addition, we propose a novel latent metric learning method, which effectively achieves metric learning in an iterative manner for each flexible model: once the latent information is specified, the metric matrix can be obtained via some classical metric learning algorithms; with the computed metric matrix, the latent states can be obtained by searching the state space exhaustively. Finally, we conduct extensive experiments on seven publicly available datasets to validate the effectiveness of the proposed method. The experimental results demonstrate that our method performs favourably against the state-of-the-art methods.

## APPENDIX A
## LEARNING LATENT LADF DISTANCE METRIC

In addition to the mahalanobis metric, we attempt to introduce the LADF distance [9] into our framework. According to

the LADF metric, the distance between feature vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ is defined as

$$f_2(\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A}, \mathbf{B}, c) = \frac{1}{2}\boldsymbol{\alpha}^\top \mathbf{A}\boldsymbol{\alpha} + \frac{1}{2}\boldsymbol{\beta}^\top \mathbf{A}\boldsymbol{\beta} - \boldsymbol{\alpha}^\top \mathbf{B}\boldsymbol{\beta} + c \quad (23)$$

where matrices $\mathbf{A}$, $\mathbf{B}$ and variable $c$ are parameters to be learned. Similar with equation (1), the distance between pedestrians $\phi$ and $\varphi$ defined by the LDAF metric is written as

$$d_{\phi,\varphi} = \min_{m^{\phi}_{\phi,\varphi}, m^{\varphi}_{\phi,\varphi}} f_2(\mathbf{x}_{\phi,m^{\phi}_{\phi,\varphi}}, \mathbf{x}_{\varphi,m^{\varphi}_{\phi,\varphi}}, \mathbf{A}, \mathbf{B}, c) \quad (24)$$

Then equation (2) can be rewritten as

$$\begin{aligned}\mathbf{A}, \mathbf{B}, c = \arg\min_{\mathbf{A},\mathbf{B},c} & \frac{1}{N_1}\sum_{\phi=1}^{N}\sum_{\substack{\varphi=1,\\ \phi\neq\varphi,\\ y_\phi=y_\varphi}}^{N} \min_{\substack{m^{\phi}_{\phi,\varphi},\\ m^{\varphi}_{\phi,\varphi}}} D_3(\mathbf{x}_{\phi,m^{\phi}_{\phi,\varphi}}, \mathbf{x}_{\varphi,m^{\varphi}_{\phi,\varphi}}) \\ & + \frac{1}{N_2}\sum_{\phi=1}^{N}\sum_{\substack{\varphi=1,\\ y_\phi\neq y_\varphi}}^{N} D_4(\mathbf{x}_\phi, \mathbf{x}_\varphi) + \frac{\lambda_1}{2}\|\mathbf{A}\|_F^2 + \frac{\lambda_2}{2}\|\mathbf{B}\|_F^2 \\ & s.t.\ \mathbf{A}, \mathbf{B} \in \text{symmetry}. \end{aligned} \quad (25)$$

where $D_3(\cdot)$ and $D_4(\cdot)$ can be rewritten as

$$\begin{aligned} D_3(\mathbf{x}_{\phi,m^{\phi}_{\phi,\varphi}}, \mathbf{x}_{\varphi,m^{\varphi}_{\phi,\varphi}}) &= l_\varepsilon\left(f_2(\mathbf{x}_{\phi,m^{\phi}_{\phi,\varphi}}, \mathbf{x}_{\varphi,m^{\varphi}_{\phi,\varphi}}, \mathbf{A}, \mathbf{B}, c)\right) \\ D_4(\mathbf{x}_\phi, \mathbf{x}_\varphi) &= \sum_{\substack{m^{\phi}_{\phi,\varphi}\in\mathcal{M}_\phi\\ m^{\varphi}_{\phi,\varphi}\in\mathcal{M}_\varphi}} l_\varepsilon\left(-f_2(\mathbf{x}_{\phi,m^{\phi}_{\phi,\varphi}}, \mathbf{x}_{\varphi,m^{\varphi}_{\phi,\varphi}}, \mathbf{A}, \mathbf{B}, c)\right) \end{aligned} \quad (26)$$

For model learning, the similar iterative optimization manner is exploited, and with the optimal latent variable pairs $(\hat{m}^{\phi}_{\phi,\varphi}, \hat{m}^{\varphi}_{\phi,\varphi})_{y_\phi=y_\varphi}$, $\phi = 1, \ldots, N, \varphi = 1, \ldots, N$, we rewrite equation (6) as

$$\begin{aligned}\hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{c} = \arg\min_{\mathbf{A},\mathbf{B},c} & \frac{1}{N_1}\sum_{\phi=1}^{N}\sum_{\substack{\varphi=1,\phi\neq\varphi,\\ y_\phi=y_\varphi}}^{N} D_3(\mathbf{x}_{\phi,\hat{m}^{\phi}_{\phi,\varphi}}, \mathbf{x}_{\varphi,\hat{m}^{\varphi}_{\phi,\varphi}}) \\ & + \frac{1}{N_2}\sum_{\phi=1}^{N}\sum_{\varphi=1,y_\phi\neq y_\varphi}^{N} D_4(\mathbf{x}_\phi, \mathbf{x}_\varphi) + \frac{\lambda_1}{2}\|\mathbf{A}\|_F^2 + \frac{\lambda_2}{2}\|\mathbf{B}\|_F^2 \\ & s.t.\ \mathbf{A}, \mathbf{B} \in \text{symmetry}. \end{aligned} \quad (27)$$

where $\lambda_1$ and $\lambda_2$ are set as $10^{-5}$.

The gradients of matrices $\mathbf{A}$, $\mathbf{B}$ and variable $c$ have similar form as equation (7), we directly provide them as

$$\begin{aligned}\dot{\mathbf{A}} = & \frac{1}{N_1}\sum_{\phi=1}^{N}\sum_{\substack{\varphi=1,\phi\neq\varphi,\\ y_\phi=y_\varphi}}^{N} \mathbf{W}_2^+(\mathbf{x}_{\phi,\hat{m}^{\phi}_{\phi,\varphi}}, \mathbf{x}_{\varphi,\hat{m}^{\varphi}_{\phi,\varphi}}) + \lambda_1\mathbf{A} \\ & - \frac{1}{N_2}\sum_{\phi=1}^{N}\sum_{\varphi=1,y_\phi\neq y_\varphi}^{N}\sum_{\substack{m^{\phi}_{\phi,\varphi}\in\mathcal{M}_\phi,\\ m^{\varphi}_{\phi,\varphi}\in\mathcal{M}_\varphi}} \mathbf{W}_2^-(\mathbf{x}_{\phi,m^{\phi}_{\phi,\varphi}}, \mathbf{x}_{\varphi,m^{\varphi}_{\phi,\varphi}}), \\ \dot{\mathbf{B}} = & \frac{1}{N_1}\sum_{\phi=1}^{N}\sum_{\substack{\varphi=1,\phi\neq\varphi,\\ y_\phi=y_\varphi}}^{N} \mathbf{W}_3^+(\mathbf{x}_{\phi,\hat{m}^{\phi}_{\phi,\varphi}}, \mathbf{x}_{\varphi,\hat{m}^{\varphi}_{\phi,\varphi}}) + \lambda_2\mathbf{B} \\ & - \frac{1}{N_2}\sum_{\phi=1}^{N}\sum_{\varphi=1,y_\phi\neq y_\varphi}^{N}\sum_{\substack{m^{\phi}_{\phi,\varphi}\in\mathcal{M}_\phi,\\ m^{\varphi}_{\phi,\varphi}\in\mathcal{M}_\varphi}} \mathbf{W}_3^-(\mathbf{x}_{\phi,m^{\phi}_{\phi,\varphi}}, \mathbf{x}_{\varphi,m^{\varphi}_{\phi,\varphi}}), \\ \dot{c} = & \frac{1}{N_1}\sum_{\phi=1}^{N}\sum_{\substack{\varphi=1,\phi\neq\varphi,\\ y_\phi=y_\varphi}}^{N} \mathbf{W}_4^+(\mathbf{x}_{\phi,\hat{m}^{\phi}_{\phi,\varphi}}, \mathbf{x}_{\varphi,\hat{m}^{\varphi}_{\phi,\varphi}}) \\ & - \frac{1}{N_2}\sum_{\phi=1}^{N}\sum_{\varphi=1,y_\phi\neq y_\varphi}^{N}\sum_{\substack{m^{\phi}_{\phi,\varphi}\in\mathcal{M}_\phi,\\ m^{\varphi}_{\phi,\varphi}\in\mathcal{M}_\varphi}} \mathbf{W}_4^-(\mathbf{x}_{\phi,m^{\phi}_{\phi,\varphi}}, \mathbf{x}_{\varphi,m^{\varphi}_{\phi,\varphi}}), \end{aligned} \quad (28)$$

where $\mathbf{W}_i^+(\cdot)$, $\mathbf{W}_i^-(\cdot)$ $(i = 2, 3, 4)$ are respectively

$$\begin{aligned} \mathbf{W}_2^+(\boldsymbol{\alpha}, \boldsymbol{\beta}) &= \frac{1}{1+\exp(-f_2(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{A}, \boldsymbol{B}, c))}(\frac{1}{2}\boldsymbol{\alpha}\boldsymbol{\alpha}^\top + \frac{1}{2}\boldsymbol{\beta}\boldsymbol{\beta}^\top) \\ \mathbf{W}_2^-(\boldsymbol{\alpha}, \boldsymbol{\beta}) &= \frac{1}{1+\exp(f_2(\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A}, \mathbf{B}, c))}(\frac{1}{2}\boldsymbol{\alpha}\boldsymbol{\alpha}^\top + \frac{1}{2}\boldsymbol{\beta}\boldsymbol{\beta}^\top) \\ \mathbf{W}_3^+(\boldsymbol{\alpha}, \boldsymbol{\beta}) &= \frac{-1}{1+\exp(-f_2(\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A}, \mathbf{B}, c))}\boldsymbol{\alpha}\boldsymbol{\beta}^\top \\ \mathbf{W}_3^-(\boldsymbol{\alpha}, \boldsymbol{\beta}) &= \frac{-1}{1+\exp(f_2(\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A}, \mathbf{B}, c))}\boldsymbol{\alpha}\boldsymbol{\beta}^\top \\ \mathbf{W}_4^+(\boldsymbol{\alpha}, \boldsymbol{\beta}) &= \frac{1}{1+\exp(-f_2(\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A}, \mathbf{B}, c))} \\ \mathbf{W}_4^-(\boldsymbol{\alpha}, \boldsymbol{\beta}) &= \frac{1}{1+\exp(f_2(\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A}, \mathbf{B}, c))} \end{aligned} \quad (29)$$

## REFERENCES


[1] L. Wei and X. Wang, "Locally aligned feature transforms across views," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3594–3601.

[2] L. Wei, R. Zhao, T. Xiao, and X. Wang, "Deepreid: Deep filter pairing neural network for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 152–159.

[3] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1116–1124.

[4] L. Zheng *et al.*, "Mars: A video benchmark for large-scale person re-identification," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 868–884.

[5] D. Gray and H. Tao, "Viewpoint invariant pedestrian recognition with an ensemble of localized features," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 262–275.

[6] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani, "Person re-identification by symmetry-driven accumulation of local features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2360–2367.

[7] B. Ma, Y. Su, and F. Jurie, "Bicov: A novel image representation for person re-identification and face verification," in *Proc. Brit. Machive Vis. Conf.*, 2012, p. 11.

[8] C. Liu, S. Gong, C. C. Loy, and X. Lin, "Person re-identification: What features are important?" in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 391–401.

[9] Z. Li, S. Chang, F. Liang, T. S. Huang, L. Cao, and J. R. Smith, "Learning locally-adaptive decision functions for person verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3610–3617.

[10] R. Layne, T. M. Hospedales, S. Gong, and Q. Mary, "Person re-identification by attributes," in *Proc. Brit. Machive Vis. Conf.*, 2012.

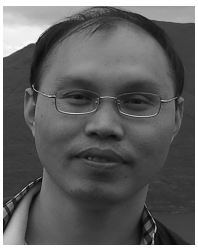
[11] W.-S. Zheng, S. Gong, and T. Xiang, "Person re-identification by probabilistic relative distance comparison," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 649–656.

[12] M. Dikmen, E. Akbas, T. S. Huang, and N. Ahuja, "Pedestrian recognition with a learned metric," in *Proc. Asian Conf. Comput. Vis.*, 2011, pp. 501–512.

[13] W. Li, R. Zhao, and X. Wang, "Human reidentification with transferred metric learning," in *Proc. Asian Conf. Comput. Vis.*, 2012, pp. 31–44.
[14] A. Mignon and F. Jurie, "Pcca: A new approach for distance learning from sparse pairwise constraints," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2666–2672.
[15] Z. Shi, T. M. Hospedales, and T. Xiang, "Transferring a semantic representation for person re-identification and search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 4184–4193.
[16] L. Zheng, S. Wang, L. Tian, F. He, Z. Liu, and Q. Tian, "Query-adaptive late fusion for image search and person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1741–1750.
[17] M. Hirzer, P. M. Roth, M. Köstinger, and H. Bischof, "Relaxed pairwise learned metric for person re-identification," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 780–793.
[18] M. Köstinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof, "Large scale metric learning from equivalence constraints," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2288–2295.
[19] R. Zhao, W. Ouyang, and X. Wang, "Unsupervised salience learning for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3586–3593.
[20] R. Zhao, W. Ouyang, and X. Wang, "Person re-identification by salience matching," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2528–2535.
[21] R. Zhao, W. Ouyang, and X. Wang, "Learning mid-level filters for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 144–151.
[22] Y. Xu, L. Lin, W.-S. Zheng, and X. Liu, "Human re-identification by matching compositional template with cluster sampling," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 3152–3159.
[23] C.-N. J. Yu and T. Joachims, "Learning structural svms with latent variables," in *Proc. ACM Int. Conf. Mach. Learn.*, 2009, pp. 1169–1176.
[24] A. Vedaldi and A. Zisserman, "Structured output regression for detection with partial truncation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 1928–1936.
[25] L. Zhu, Y. Chen, A. Yuille, and W. Freeman, "Latent hierarchical structural learning for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 1062–1069.
[26] R. Yao, Q. Shi, C. Shen, Y. Zhang, and A. van den Hengel, "Part-based visual tracking with online latent structural learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2363–2370.
[27] S. Liao, G. Zhao, V. Kellokumpu, M. Pietikäinen, and S. Z. Li, "Modeling pixel process with scale invariant local patterns for background subtraction in complex scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 1301–1306.
[28] B. Prosser, W.-S. Zheng, S. Gong, T. Xiang, and Q. Mary, "Person re-identification by support vector ranking," in *Proc. Brit. Machive Vis. Conf.*, 2010.
[29] X. Liu, H. Wang, Y. Wu, J. Yang, and M.-H. Yang, "An ensemble color model for human re-identification," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Jan. 2015, pp. 868–875.
[30] S. Liao, Y. Hu, X. Zhu, and S. Z. Li, "Person re-identification by local maximal occurrence representation and metric learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 2197–2206.
[31] X. Wang, G. Doretto, T. Sebastian, J. Rittscher, and P. Tu, "Shape and appearance context modeling," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–8.
[32] W.-S. Zheng, S. Gong, and T. Xiang, "Associating groups of people," in *Proc. Brit. Machive Vis. Conf.*, 2009.
[33] C. C. Loy, T. Xiang, and S. Gong, "Multi-camera activity correlation analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1988–1995.
[34] P. M. Roth, M. Hirzer, M. Köstinger, C. Beleznai, and H. Bischof, "Mahalanobis distance learning for person re-identification," in *Person Re-Identification*. New York, NY, USA: Springer, 2014.
[35] M. Hirzer, C. Beleznai, P. M. Roth, and H. Bischof, "Person re-identification by descriptive and discriminative classification," in *Proc. Scandin. Conf. Image Anal.*, 2011, pp. 91–102.
[36] S. Pedagadi, J. Orwell, S. Velastin, and B. Boghossian, "Local fisher discriminant analysis for pedestrian re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3318–3325.
[37] S. Paisitkriangkrai, C. Shen, and A. V. D. Hengel, "Learning to rank in person re-identification with metric ensembles," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1846–1855.
[38] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 1, pp. 40–51, Jan. 2007.
[39] F. Xiong, M. Gou, O. Camps, and M. Sznaier, "Person re-identification using kernel-based metric learning methods," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 1–16.
[40] J. Chen, Z. Zhang, and Y. Wang, "Relevance metric learning for person re-identification by exploiting global similarities," in *Proc. IEEE Int. Conf. Pattern Recognit.*, Aug. 2014, pp. 1657–1662.
[41] C. C. Loy, C. Liu, and S. Gong, "Person re-identification by manifold ranking," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2013, pp. 3567–3571.
[42] L. Ma, X. Yang, and D. Tao, "Person re-identification over camera networks using multi-task distance metric learning," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3656–3670, Aug. 2014.
[43] D. Chen, Z. Yuan, G. Hua, N. Zheng, and J. Wang, "Similarity learning on an explicit polynomial kernel feature map for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1565–1573.
[44] M. Hirzer, P. M. Roth, and H. Bischof, "Person re-identification by efficient impostor-based metric learning," in *Proc. IEEE Int. Conf. Adv. Video Signal-Based Surveill.*, Sep. 2012, pp. 203–208.
[45] Y. Yang, J. Yang, J. Yan, S. Liao, D. Yi, and S. Z. Li, "Salient color names for person re-identification," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 536–551.
[46] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon, "Information-theoretic metric learning," in *Proc. ACM Int. Conf. Mach. Learn.*, 2007, pp. 209–216.
[47] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.
[48] M. Guillaumin, J. Verbeek, and C. Schmid, "Is that you? Metric learning approaches for face identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Sep./Oct. 2009, pp. 498–505.
[49] K. Q. Weinberger, J. Blitzer, and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," in *Proc. Adv. Neural Inf. Process. Syst.*, 2006, pp. 1473–1480.

**Chong Sun** received the B.E. degree in electronic information engineering, Dalian University of Technology (DUT), Dalian, China, in 2012. He is currently pursuing the Ph.D. degree with the School of Information and Communication Engineering, DUT. His research interests are in the object tracking, person re-identification, and object segmentation.

**Dong Wang** received the B.E. degree in electronic information engineering and the Ph.D. degree in signal and information processing from the Dalian University of Technology (DUT), Dalian, China, in 2008 and 2013, respectively. He is currently a Faculty Member with the School of Information and Communication Engineering, DUT. His current research interests include face recognition, interactive image segmentation, and object tracking.



**Huchuan Lu** (SM'12) received the M.Sc. degree in signal and information processing and the Ph.D. degree in system engineering from the Dalian University of Technology (DUT), China, in 1998 and 2008, respectively. He has been a Faculty Member since 1998 and a Professor since 2012 with the School of Information and Communication Engineering, DUT. His research interests are in the areas of computer vision and pattern recognition. In recent years, he focuses on visual tracking and segmentation. He serves as an Associate Editor of the IEEE TRANSACTIONS ON CYBERNETICS.