# Person Re-Identification using Kernel-based Metric Learning Methods

Fei Xiong, Mengran Gou, Octavia Camps and Mario Sznaier

Dept. of Electrical and Computer Engineering,
Northeastern University, Boston, MA 02115
{fxiong,mengran,camps,msznaier}@coe.neu.edu
http://robustsystems.coe.neu.edu

**Abstract.** Re-identification of individuals across camera networks with limited or no overlapping fields of view remains challenging in spite of significant research efforts. In this paper, we propose the use, and extensively evaluate the performance, of four alternatives for re-ID classification: regularized Pairwise Constrained Component Analysis, kernel Local Fisher Discriminant Analysis, Marginal Fisher Analysis and a ranking ensemble voting scheme, used in conjunction with different sizes of sets of histogram-based features and linear, $\chi^2$ and RBF-$\chi^2$ kernels. Comparisons against the state-of-art show significant improvements in performance measured both in terms of Cumulative Match Characteristic curves (CMC) and Proportion of Uncertainty Removed (PUR) scores on the challenging VIPeR, iLIDS, CAVIAR and 3DPeS datasets.

## 1 Introduction

Surveillance systems for large public spaces (i.e. airport terminals, train stations, etc.) use networks of cameras to maximize their coverage area. However, due to economical and infrastructural reasons, these cameras often have very little or no overlapping field of view. Thus, recognizing individuals across cameras is a critical component when tracking in the network.

The task of re-identification (re-ID) can be formalized as the problem of matching a given *probe* image against a *gallery* of candidate images. As illustrated in Figure 1(a), this is a very challenging task since images of the same individual can be very different due to variations in pose, viewpoint, and illumination. Moreover, due to the (relatively low) resolution and the placement of the cameras, different individuals may appear very similar and with little or none visible faces, preventing the use of biometric and soft-biometric approaches [9, 24].

A good overview of existing re-ID methods can be found in [7, 10, 13, 23, 29] and references therein. The three most important aspects in re-ID are i) the features used, ii) the matching procedure, and iii) the performance evaluation. Most re-ID approaches use appearance-based features that are viewpoint quasi-invariant [2, 3, 5, 11, 12, 14, 25] such as color and texture descriptors. However, the number and support of features used varies greatly across approaches making it difficult to compare their impact on performance. Using standard metrics such

**Fig. 1.** The re-ID problem. (a) Challenges (left to right): low resolution, occlusion, viewpoint, pose, and illumination variations and similar appearance of different people. (b) Projecting the data improves classification performance.

as Euclidean distance to match images based on this type of features results in poor performance due to the large variations in pose and illumination and limited training data. Thus, recent approaches [18,20,21,29,31] design classifiers to learn specialized metrics (see Figure 1(b)), that enforce features from the same individual to be closer than features from different individuals. Yet, state-of-the-art performance remains low, slightly above 30% for the best match. Performance is often reported on standard datasets that bring in different biases. Moreover, the number of datasets and the experimental evaluation protocols used vary greatly across approaches, making difficult to compare them.

This paper focuses on all aspects of the problem, feature extraction, distance learning for re-ID classification, and performance evaluation. In particular:

- We explore the effect of the size and location of support regions for commonly used histogram-based feature vectors may have on classification performance.
- We propose four kernel-based distance learning approaches to improve re-ID classification accuracy when the data space is under-sampled: regularized Pairwise Constrained Component Analysis (rPCCA), kernel Local Fisher Discriminant Classifier (kLFDA), Marginal Fisher Analysis (MFA) [26], and a ranking ensemble voting (REV) scheme.
- We provide a comprehensive performance evaluation using four sets of features, three kernels (linear, $\chi^2$ and RBF-$\chi^2$) and four challenging re-ID datasets: VIPeR [14], CAVIAR [8], 3DPeS [4] and iLIDS [30]. Using this protocol, we compare the proposed methods against four state-of-the-art methods: Pairwise Constrained Component Analysis (PCCA) [20], Local Fisher Discriminant Analysis (LDFA) [21], SVMML [18] and KISSME [15].

Our experiments not only allow us to compare previously published classification techniques using a common set of features and datasets (an experiment that to the best of our knowledge has not been reported so far) but also show that the classification methods proposed here result in a significant improvement in performance over the state-of-the-art.

## 2   Related Work

Re-ID data samples consist of images of individuals, cropped such that the target occupies most of the image. The most commonly used features are inspired on a "bag-of-words" approach and are histograms based using local support regions within the target's bounding box [10]. Yet, the number of support regions and the dimension of the feature vector can vary widely. For example, Mignon and Jurie [20] use feature vectors of dimension 2,580 while [21] use feature vectors of dimension 22,506. In our experiments we evaluate the effect of these choices on re-ID accuracy performance. As shown in our experiments, using too many of these features can decrease performance.

Most re-ID approaches can be formalized as a supervised metric/distance learning algorithm where a projection matrix $\mathbf{P}$ is sought so that the projected Mahalanobis-like distance $D_{\mathbf{M}}(\mathbf{x}_{i_k}, \mathbf{x}_{j_k}) = (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{M}(\mathbf{x}_i - \mathbf{x}_j)$, where $\mathbf{M} = \mathbf{P}^T\mathbf{P}$, is small when feature vectors $\mathbf{x}_{i_k}$ and $\mathbf{x}_{j_k}$ represent the same person and large otherwise.

The best reported performance on the VIPeR dataset [18] was achieved using an adaptive boundary approach that jointly learns the distance metric and an adaptive thresholding rule. However, a drawback of this approach is that it scales poorly since its computational complexity is $O(d^2)$ where $d$ is the dimension of the feature vector $\mathbf{x}_{i_k}$. An alternative approach is to use a logistic function to approximate the hinge loss so that the global optimum still can be achieved by iteratively gradient search along $\mathbf{P}$ as in Pairwise Constrained Component Analysis (PCCA) [20] and in (PRDC) [29]. However, these methods are prone to over fitting. We propose to address this problem by introducing a regularization term that uses the additional degrees of freedom available in the problem to maximize the inter-class margin.

The state-of-the-art performance on the CAVIAR and the 3DPeS datasets was achieved by using a Local Fisher Discriminant Classifier (LFDA) as proposed by Pedagadi et al. [21]. While this approach has a closed form solution for the Mahalanobis matrix, it requires an eigenanalysis of a $d \times d$ scatter matrix. For large $d$, [21] proposed to first reduce the dimensionality of the data using principal component analysis (PCA). However, PCA can eliminate discriminant features defeating the benefits of LFDA. We propose instead to use a kernel approach to preserve discriminant features while reducing the dimension of the problem to a $N \times N$ eigendecomposition, where $N << d$ is the number of images.

### 2.1   Notation

For the sake of clarity, we list the notation used in this paper here. $\mathbf{x}_i \in \mathbf{R}^d$ is a feature vector representing the $i$th image. $l_i \in \{1, \cdots, c\}$ is the *identity label* for the $i$th image. A pair of samples $(\mathbf{x}_{i_k}, \mathbf{x}_{j_k})$ has associated a *class label* $y_k = 1$ if $l_i = l_j$ and $y_k = -1$, otherwise. $N << d$, $N_c$ and $N' \leq N^2$ represent the total number of samples, the number of images with label $c$, and the number of pairs of images used, respectively[1]. $\phi(\mathbf{x})$ is a mapping from feature to Kernel space.

---

[1] We will use all possible positive pairs but only a fraction of the negative ones.

## 3    Proposed Methods

In this section we propose four possible approaches towards increasing accuracy performance. The first approach, rPCCA, is a new iterative procedure that introduces a regularization term to maximize the inter-class margin to the hinge loss PCCA approach. The second approach, kLDFA, is a new closed-form method that uses a kernel trick to handle large dimensional feature vectors while maximizing a Fischer optimization criteria. The third approach is to use the Marginal Fisher Analysis method introduced in [26] which to best of our knowledge has not been used for re-ID before. Finally, we also propose a new ensemble approach where the results of multiple classifiers are combined to exploit their individual strengths.

### 3.1    Regularized PCCA (rPCCA)

In [20] Mignon and Jurie proposed to use PCCA with an approximation to the hinge loss to learn the projected metric. Their motivation was that the projected distances between samples from the same class should be smaller than a given threshold $T$ while the distances between inter-class samples should be larger than $T$. To this effect, without loss of generality, they set $T = 1$ and then approximated the hinge loss with the generalized logistic loss function [27] $\ell_\beta(x) = \frac{1}{\beta}log(1 + e^{\beta x})$ to form the objective function:

$$\min_{\mathbf{P}} E(\mathbf{P}) = \sum_{k=1}^{N'} \ell_\beta(y_k(D_{\mathbf{P}}^2(\mathbf{x}_{i_k}, \mathbf{x}_{j_k}) - 1)) \tag{1}$$

where $\mathbf{P}$ is a $d' \times d$ matrix ($d' < d$) that is found using a gradient descent-based method. Additionally, it is possible to use a "kernel trick" to improve classification when the data is not linearly separable. In this case, a projection $d' \times N$ matrix $\mathbf{Q}$ is applied to the feature vectors in the kernel space $\mathbf{P} = \mathbf{Q}\phi^T(\mathbf{X})$ and the objective function becomes:

$$E(\mathbf{Q}) = \sum_{k=1}^{N'} \ell_\beta[y_k((\mathbf{e}_{i_k} - \mathbf{e}_{j_k})^T \mathbf{K}\mathbf{Q}^T \mathbf{Q}\mathbf{K}(\mathbf{e}_{i_k} - \mathbf{e}_{j_k}) - 1)] \tag{2}$$

where $\mathbf{K} = \phi(\mathbf{X})^T \phi(\mathbf{X})$ is the $N \times N$ kernel matrix and $\mathbf{e}_i$ is the $i$th vector of the canonical basis in $\mathbf{R}^N$ – i.e. a unit vector with 1 at position $i$. Using trace,

$$E(\mathbf{Q}) = \sum_{k=1}^{N'} \sum \ell_\beta y_k \text{trace}[\mathbf{Q}\mathbf{K}(\mathbf{e}_{ik} - \mathbf{e}_{jk})(\mathbf{e}_{ik} - \mathbf{e}_{jk})^T \mathbf{K}\mathbf{Q}^T] - 1 \tag{3}$$

and the gradient of the new objective function $E(\mathbf{Q})$ is:

$$\frac{\partial E}{\partial \mathbf{Q}} = 2\mathbf{Q} \sum_{k=1}^{N'} y_k \sigma_\beta(y_k(D_{\mathbf{P}}^2(\mathbf{x}_{i_k}, \mathbf{x}_{j_k}) - 1))\mathbf{K}\mathbf{C}_k\mathbf{K} \tag{4}$$

where $\sigma_\beta(x) = (1 + e^{-\beta x})^{-1}$ for $\beta = 1$ and $\mathbf{C}_k = (\mathbf{e}_{i_k} - \mathbf{e}_{j_k})(\mathbf{e}_{i_k} - \mathbf{e}_{j_k})^T$. The matrix $\mathbf{K}$ is full rank since $\phi(\mathbf{X})$ is $D \times N$ and $D > d >> N$. Then, one can multiply the gradient with a preconditioner $\mathbf{K}^{-1}$ and iteratively solve the problem by updating $\mathbf{Q}$ using the expression

$$\mathbf{Q}_{t+1} = \mathbf{Q}_t(\mathbf{I} - 2\eta \sum_{k=1}^{N'} \mathcal{L}_k^t \mathbf{K} \mathbf{C}_k) \qquad (5)$$

where $\eta$ is the learning rate and where $\mathcal{L}_k^t$ denotes the value of $y_k \sigma_\beta(y_k(D_{\mathbf{P}}^2(\mathbf{x}_{i_k}, \mathbf{x}_{j_k}) - 1))$ at time $t$. It can be easily shown that the effect of this preconditioning step is that using changes in direction of $\mathbf{Q}$ results in the desired optimal change in direction of $\mathbf{P}$. Furthermore, it should be noted that updating $\mathbf{Q}$ uses $\mathbf{K}$ but does not require to compute its inverse.

PCCA can result in poor classification performance due to large variations among samples and limited training data. We propose to address this problem by using the additional degrees of freedom available in the problem to maximize the inter-class margin. To this effect, motivated by the objective functions used on SVMs, we propose the *regularized* PCCA (rPCCA) objective function with a regularization term penalizing the Frobenius norm of $\mathbf{P}$:

$$E(\mathbf{P}) = \sum_{k=1}^{N'} \ell_\beta(y_k(D_{\mathbf{P}}^2(\mathbf{x}_{i_k}, \mathbf{x}_{j_k}) - 1)) + \lambda ||\mathbf{P}||_F^2 \qquad (6)$$

where $\lambda$ is the regularization parameter. Briefly, the intuition behind this new objective function is to treat each of the rows $\mathbf{p}_i$ of $\mathbf{P}$ as the separating hyperplane in an SVM and use the fact that the classification margin is precisely given by $(||\mathbf{p}_i||_2)^{-1}$. Substituting $\mathbf{P}$ with $\mathbf{Q}\phi^T(\mathbf{X})$, the derivative of the regularized objective function with respect to $\mathbf{Q}$ becomes:

$$\frac{\partial E}{\partial \mathbf{Q}} = 2\mathbf{Q}(\sum_{k=1}^{N'} \mathcal{L}_k^t \mathbf{K} \mathbf{C}_k + \lambda \mathbf{I})\mathbf{K} \qquad (7)$$

Similarly to PCCA, the global optimum can be achieved by multiplying the gradient with the preconditioner $\mathbf{K}^{-1}$ and iteratively updating the matrix $\mathbf{Q}$.

### 3.2   Kernel LFDA (kLDFA)

A drawback of using LFDA is that it requires solving a generalized eigenvalue problem of very large scatter $d \times d$ matrices. For example, in [21] the authors use feature vectors with $d = 22506$ features. To circumvent this problem, [21] proposed to exploit the redundancy among the features by performing a preprocessing step where principal component analysis (PCA) is used to reduce the dimensionality of the data. However, a potential difficulty here is that this unsupervised dimensionality reduction step, when applied to relatively small datasets, can result in an undesirable compression of the most discriminative features. To

avoid this problem, we propose to use a kernel approach, based on the method introduced in [22] in the context of supervised dimensionality reduction. The benefits of this approach are twofold: it avoids performing an eigenvalue decomposition of the large scatter matrices and it can exploit the flexibility in choosing the kernel to improve the classification accuracy.

The proposed *kernel* LDFA (kLDFA) method finds a projection matrix $\mathbf{P} \in \mathbf{R}^{d' \times d}$ to maximize the 'between-class' scatter while minimizing the 'within-class' scatter for *similar* samples using the Fisher discriminant objective:

$$\mathbf{P} = \max_{\mathbf{P}}(\mathbf{P}\mathbf{S}^{\mathbf{w}}\mathbf{P})^{-1}\mathbf{P}^{T}\mathbf{S}^{\mathbf{b}}\mathbf{P} \tag{8}$$

where the *within* and *between* scatter matrices are $\mathbf{S}^{\mathbf{w}} = \frac{1}{2}\phi(\mathbf{X})\tilde{\mathbf{S}}^{\mathbf{w}}\phi(\mathbf{X})^{T}$ and $\mathbf{S}^{\mathbf{b}} = \frac{1}{2}\phi(\mathbf{X})\tilde{\mathbf{S}}^{\mathbf{b}}\phi(\mathbf{X})^{T}$ where $\tilde{\mathbf{S}}^{\mathbf{w}} = \sum_{i,j=1}^{N}\mathbf{A}_{i,j}^{w}(\mathbf{e}_{i} - \mathbf{e}_{j})(\mathbf{e}_{i} - \mathbf{e}_{j})^{T}$ and $\tilde{\mathbf{S}}^{\mathbf{b}} = \sum_{i,j=1}^{N}\mathbf{A}_{i,j}^{b}(\mathbf{e}_{i} - \mathbf{e}_{j})(\mathbf{e}_{i} - \mathbf{e}_{j})^{T}$. Then, representing the projection matrix with the data samples in the kernel space $\mathbf{P} = \mathbf{Q}\phi^{T}(\mathbf{X})$, the kLFDA problem is formulated as:

$$\mathbf{Q} = \max_{\mathbf{Q}}(\mathbf{Q}\mathbf{K}\tilde{\mathbf{S}}^{\mathbf{w}}\mathbf{K}\mathbf{Q})^{-1}\mathbf{Q}\mathbf{K}\tilde{\mathbf{S}}^{\mathbf{b}}\mathbf{K}\mathbf{Q} \tag{9}$$

Since the within class scatter matrix $\tilde{\mathbf{S}}^{\mathbf{w}}$ is usually rank deficient, a regularized $\hat{\mathbf{S}}^{\mathbf{w}}$ defined below is used instead:

$$\hat{\mathbf{S}}^{\mathbf{w}} = (1 - \alpha)\tilde{\mathbf{S}}^{\mathbf{w}} + \frac{\alpha}{N}trace(\tilde{\mathbf{S}}^{\mathbf{w}})\mathbf{I} \tag{10}$$

### 3.3    Marginal Fisher Analysis(MFA)

Marginal Fisher Analysis (MFA) was proposed in [26] as yet another graph embedding dimension reduction method. Similarly to kLDFA and LDFA, it has a closed form solution given by a general eigenvalue decomposition. However, in contrast to LDFA, its special discriminant objective allows to maximize the marginal discriminant even when the assumption of a Gaussian distribution for each class is not true. Moreover, the results in [26] showed that the learned discriminant components have larger margin between classes, similar to a SVM. The scatter matrices for MFA are defined as:

$$\tilde{\mathbf{S}}^{\mathbf{w}} = (\mathbf{D}^{\mathbf{w}} - \mathbf{W}^{\mathbf{w}}) \text{ and } \tilde{\mathbf{S}}^{\mathbf{b}} = (\mathbf{D}^{\mathbf{b}} - \mathbf{W}^{\mathbf{b}}) \tag{11}$$

where $\mathbf{D}^{\mathbf{b}}_{ii} = \sum_{j}\mathbf{W}^{\mathbf{b}}_{ij}$, $\mathbf{D}^{\mathbf{w}}_{ii} = \sum_{j}\mathbf{W}^{\mathbf{w}}_{ij}$ as well as the sparse matrices $\mathbf{W}^{\mathbf{w}}$ and $\mathbf{W}^{\mathbf{b}}$ are defined as: $\mathbf{W}_{ij}^{w} = 1$ if and only if $\mathbf{x}_{i}$ or $\mathbf{x}_{j}$ is the $k_{w}$ nearest within class neighbor of other; and $\mathbf{W}_{ij}^{b} = 1$ if and only if $\mathbf{x}_{i}$ or $\mathbf{x}_{j}$ is the $k_{b}$ nearest between class neighbor of other.

### 3.4    Ranking Ensemble Voting

Classification accuracy is affected by the method used to learn the projected metric, the kernel used and the features used to represent the data. Thus, it

is possible to design an ensemble of classifiers that use different kernels and sets of features. Then, given a test image and a gallery of candidate matches, each of these classifiers will produce, in principle, a different ranking among the candidates which, in turn, could be combined to produce a single and better ranking. That is, instead of tuning for the best set of parameters through cross-validation, one could independently run different ranking classifiers and merge the results. In this paper, we will consider two alternative ways on how to combine the results from the individual rankings into a ranking ensemble voting (REV) scheme; *"Ensemble 1"*: adding the rankings in a simple voting scheme; or *"Ensemble 2"*: assuming that the output of a ranking algorithm represents the probability of the $r^{th}$ closest reference image is the correct match, given the ranking algorithm $\mathcal{R}_m$, $p(r|\mathcal{R}_m)$; $m = 1, \ldots, N_r$, for each of the $N_r$ algorithms. Then, assuming conditional independence among the different algorithms we have $p(r) = \prod_{i=1}^{N_r} p(r|\mathcal{R}_i)$.

## 4 Experiments

In this section we describe the set of experiments used to evaluate the proposed methods as well as the choice of features and kernels. In particular, we compared the performance of rPCCA, kLFDA, MFA and REV, against the current state-of-art PCCA, LFDA, SVMML and KISSME, using four different sets of features, three different kernels, and four different datasets, as described below.



(a) VIPeR          (b) iLIDS          (c) CAVIAR          (d) 3DPeS
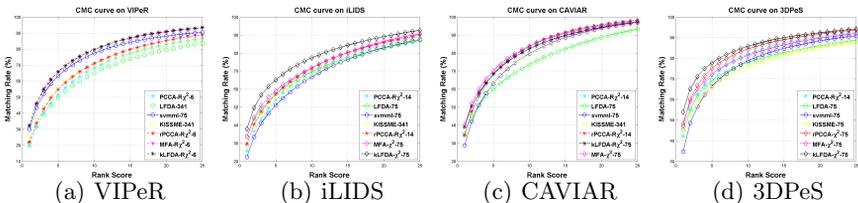
**Fig. 2.** Best CMC curves for each method on four datasets.

### 4.1 Datasets and Experimental Protocol

All the algorithms were evaluated using the four most challenging and commonly used throughout the literature datasets. The **VIPeR** dataset [14] is composed of 1264 images of 632 individuals, with 2 images of $128 \times 48$ pixels per individual. The images are taken from horizontal viewpoints but in widely different directions. The **iLIDS** dataset [30] has 476 images of 119 pedestrians. The number of images for each individual varies from 2 to 8. Since this dataset was collected at an airport, the images often have severe occlusions caused by people and luggage. The **CAVIAR** dataset [8] contains 1220 images of 72 individuals from 2 cameras in a shopping mall. Each person has 10 to 20 images. The image sizes

**Table 1.** CMC at $r = 1, 5, 10, 20$ and PUR scores on VIPeR with $p = 316$ test individuals (highest scores in red).

| | | PCCA | | | LFDA | SVMML | KISSME | rPCCA | | | kLFDA | | | MFA | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $r$ | $L$ | $\chi^2$ | $R_{\chi^2}$ | $w/o$ | $w/o$ | $w/o$ | $L$ | $\chi^2$ | $R_{\chi^2}$ | $L$ | $\chi^2$ | $R_{\chi^2}$ | $L$ | $\chi^2$ | $R_{\chi^2}$ |
| 6 | 1 | 14.3 | 16.7 | 19.6 | 19.7 | 27.0 | 23.8 | 19.1 | 19.9 | 22.0 | 20.6 | 28.1 | **32.3** | 21.1 | 28.4 | 32.2 |
| | 5 | 40.5 | 46.0 | 51.5 | 46.7 | 60.9 | 52.9 | 48.3 | 50.6 | 54.8 | 46.2 | 60.0 | 65.8 | 48.7 | 60.1 | **66.0** |
| | 10 | 57.5 | 62.6 | 68.2 | 62.1 | 75.4 | 67.1 | 64.9 | 67.8 | 71.0 | 60.8 | 75.0 | **79.7** | 63.9 | 74.8 | **79.7** |
| | 20 | 74.7 | 79.6 | 82.9 | 77.0 | 87.3 | 80.5 | 80.9 | 83.2 | 85.3 | 75.9 | 87.8 | **90.9** | 78.9 | 87.7 | 90.6 |
| | PUR | 36.1 | 39.6 | 42.9 | 38.2 | 47.6 | 42.1 | 41.0 | 42.7 | 44.8 | 37.5 | 48.4 | **52.5** | 39.9 | 48.3 | 52.4 |
| 14 | 1 | 15.0 | 17.0 | 19.7 | 20.0 | 25.3 | 22.6 | 19.3 | 19.7 | 21.1 | 21.2 | 28.9 | 31.8 | 20.9 | 28.7 | **32.2** |
| | 5 | 41.5 | 46.1 | 50.7 | 46.9 | 58.3 | 51.0 | 47.8 | 49.6 | 52.9 | 47.1 | 60.4 | 64.8 | 49.3 | 59.7 | **65.5** |
| | 10 | 58.2 | 63.1 | 67.2 | 61.9 | 73.0 | 65.1 | 64.5 | 65.9 | 69.2 | 61.3 | 74.7 | **79.1** | 63.9 | 74.4 | 79.0 |
| | 20 | 75.9 | 79.2 | 82.5 | 76.2 | 85.1 | 78.3 | 80.6 | 81.5 | 83.6 | 76.2 | 87.1 | **90.3** | 78.2 | 86.6 | **90.3** |
| | PUR | 36.8 | 39.5 | 42.2 | 38.0 | 45.4 | 40.5 | 40.7 | 41.8 | 43.5 | 38.0 | 48.3 | 51.9 | 39.8 | 47.9 | **52.0** |
| 75 | 1 | 18.3 | 18.4 | 16.4 | 21.5 | 30.1 | 25.2 | 21.1 | 20.5 | 20.5 | 23.3 | 30.3 | 30.9 | 23.6 | 29.6 | **31.1** |
| | 5 | 46.9 | 46.4 | 45.0 | 49.6 | 63.2 | 54.2 | 51.1 | 50.5 | 51.3 | 52.8 | 63.5 | 64.4 | 52.1 | 63.0 | **65.2** |
| | 10 | 63.7 | 63.4 | 61.4 | 64.6 | 77.4 | 68.4 | 67.5 | 67.4 | 67.7 | 68.4 | 77.9 | 79.3 | 67.4 | 77.3 | **79.6** |
| | 20 | 80.2 | 79.3 | 77.0 | 79.1 | 88.1 | 81.6 | 82.9 | 82.4 | 82.3 | 82.4 | 89.8 | **90.6** | 81.5 | 88.9 | **90.6** |
| | PUR | 40.1 | 39.9 | 37.9 | 40.2 | 49.4 | 43.2 | 42.7 | 42.5 | 42.6 | 43.2 | 51.0 | 51.9 | 42.6 | 50.3 | **52.0** |
| 341 | 1 | 16.2 | 15.2 | 11.8 | 21.4 | **28.0** | 25.8 | 19.2 | 19.0 | 16.8 | 23.6 | 27.0 | 24.5 | 22.7 | 27.3 | 24.8 |
| | 5 | 43.5 | 41.5 | 35.5 | 49.6 | **61.5** | 56.2 | 49.4 | 48.4 | 45.1 | 54.4 | 60.1 | 56.0 | 53.8 | 60.2 | 56.9 |
| | 10 | 59.0 | 57.0 | 51.1 | 65.2 | **76.7** | 70.1 | 65.5 | 64.7 | 60.9 | 70.1 | 75.3 | 72.1 | 69.1 | 75.2 | 72.3 |
| | 20 | 75.6 | 73.3 | 68.4 | 79.5 | 88.2 | 82.9 | 80.8 | 80.3 | 77.2 | 84.0 | **88.6** | 86.8 | 83.3 | 88.2 | 86.3 |
| | PUR | 37.2 | 35.7 | 32.0 | 40.8 | 48.7 | 44.4 | 41.3 | 40.9 | 38.6 | 44.4 | **48.9** | 46.7 | 43.9 | 48.8 | 46.7 |

of this dataset vary significantly (from $141 \times 72$ to $39 \times 17$). Finally, the **3DPeS** dataset [4] includes 1011 images of 192 individuals captured from 8 outdoor cameras with significantly different viewpoints. In this dataset each person has 2 to 26 images. Except for VIPeR, the size of the images from the other three datasets is not constant so they were scaled to $128 \times 48$ for our experiments.

In our experiments, we adopted a *Single-Shot* experiment setting. All the datasets were randomly divided into two subsets so that the test set contains $p$ individuals. This partition was repeated 10 times. Under each partition, one image for each individual in the test set was randomly selected as the reference image set and the rest of the images were used as query images. This process was repeated 10 times, as well, and it can be seen as the *recall* at each rank. The rank of the correct match was recorded and accumulated to generate the match characteristic $M(r)$.

For easy comparison with other algorithms, we report the widely used accumulated $M(r)$, Cumulative Match Characteristic (CMC) performance curves, averaged across the experiments. In addition, we also report the *proportion of uncertainty removed* (PUR) [21] scores:

$$\text{PUR} = \frac{\log(N) + \sum_{r=1}^{N} M(r) \log(M(r))}{\log(N)} \tag{12}$$

where $N$ is the size of the gallery set. This score compares the uncertainty under random selection among a gallery of images and the uncertainty after using a ranking method. Finally, since the first few retrieved images can be quickly inspected by a human, higher scores at rank $r \geq 1$ are preferred.

**Table 2.** CMC at $r = 1, 5, 10, 20$ and PUR scores on iLIDS with $p = 60$ test individuals (highest scores shown in red).

| | | PCCA | | | LFDA | SVMML | KISSME | rPCCA | | | kLFDA | | | MFA | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | L | $\chi^2$ | $R_{\chi^2}$ | w/o | w/o | w/o | L | $\chi^2$ | $R_{\chi^2}$ | L | $\chi^2$ | $R_{\chi^2}$ | L | $\chi^2$ | $R_{\chi^2}$ |
| | r | | | | | | | | | | | | | | | |
| 6 | 1 | 21.7 | 23.0 | 24.1 | 32.2 | 20.8 | 28.0 | 25.5 | 26.6 | 28.0 | 32.3 | 36.5 | **36.9** | 30.5 | 32.6 | 32.1 |
| | 5 | 49.7 | 51.1 | 53.3 | 56.0 | 49.1 | 54.2 | 53.8 | 54.3 | 56.5 | 57.2 | 64.1 | **65.3** | 53.9 | 58.5 | 58.8 |
| | 10 | 65.0 | 67.0 | 69.2 | 68.7 | 65.4 | 67.9 | 68.4 | 69.7 | 71.8 | 70.0 | 76.5 | **78.3** | 66.3 | 71.5 | 72.2 |
| | 20 | 81.4 | 83.3 | 84.8 | 81.6 | 81.7 | 81.6 | 83.0 | 84.5 | 85.9 | 83.9 | 88.5 | **89.4** | 80.4 | 84.8 | 85.9 |
| | PUR | 21.3 | 22.8 | 24.4 | 26.6 | 20.9 | 24.7 | 24.2 | 25.4 | 27.0 | 27.9 | 33.7 | **34.9** | 24.8 | 28.8 | 29.1 |
| 14 | 1 | 23.9 | 24.5 | 25.7 | 32.0 | 20.3 | 29.4 | 27.8 | 28.0 | 29.6 | 33.3 | **37.8** | 37.4 | 30.7 | 34.2 | 33.7 |
| | 5 | 53.0 | 53.2 | 54.0 | 54.2 | 48.6 | 54.9 | 55.3 | 56.0 | 57.3 | 57.5 | **64.8** | **64.8** | 54.0 | 58.9 | 59.5 |
| | 10 | 68.3 | 68.8 | 69.6 | 66.4 | 64.5 | 68.8 | 70.2 | 70.4 | 71.7 | 70.1 | 76.6 | **77.3** | 66.2 | 71.1 | 72.0 |
| | 20 | 83.9 | 84.9 | 84.4 | 80.5 | 80.9 | 82.1 | 84.6 | 85.3 | 85.9 | 83.5 | 88.6 | **89.1** | 80.7 | 85.3 | 86.0 |
| | PUR | 23.9 | 24.5 | 25.1 | 25.7 | 20.3 | 25.8 | 26.1 | 26.6 | 27.8 | 28.3 | 34.7 | **34.8** | 25.1 | 29.9 | 30.0 |
| 75 | 1 | 24.0 | 23.8 | 24.0 | 33.8 | 22.3 | 28.5 | 28.4 | 28.9 | 29.2 | 34.1 | **38.0** | 36.2 | 30.3 | 33.7 | 32.1 |
| | 5 | 53.6 | 52.9 | 51.7 | 57.4 | 51.1 | 55.3 | 57.0 | 57.1 | 57.2 | 60.4 | **65.1** | 63.5 | 56.2 | 59.3 | 57.4 |
| | 10 | 69.1 | 68.6 | 67.1 | 69.7 | 66.7 | 68.7 | 71.4 | 71.4 | 71.1 | 73.5 | **77.4** | 76.1 | 68.9 | 71.7 | 70.5 |
| | 20 | 84.4 | 84.1 | 82.8 | 82.8 | 83.0 | 83.4 | 85.8 | 85.7 | 85.4 | 86.5 | **89.2** | **89.2** | 83.6 | 86.5 | 85.9 |
| | PUR | 24.4 | 24.2 | 23.4 | 28.1 | 22.4 | 25.9 | 27.3 | 27.6 | 27.6 | 30.8 | **35.4** | 33.9 | 26.7 | 30.3 | 28.9 |
| 341 | 1 | 21.4 | 21.4 | 20.2 | 32.7 | 21.4 | 28.4 | 26.0 | 26.6 | 25.9 | 32.2 | **34.2** | 30.5 | 29.2 | 30.2 | 26.8 |
| | 5 | 49.1 | 48.5 | 45.1 | 56.7 | 49.6 | 55.7 | 53.3 | 53.4 | 52.5 | 59.9 | **61.5** | 57.3 | 55.1 | 55.3 | 50.3 |
| | 10 | 65.5 | 64.9 | 61.1 | 69.0 | 65.5 | 68.9 | 68.9 | 68.7 | 67.7 | 73.8 | **74.8** | 71.8 | 69.3 | 69.3 | 64.8 |
| | 20 | 82.1 | 81.3 | 78.4 | 82.3 | 82.8 | 83.4 | 84.5 | 84.3 | 83.0 | 86.5 | **87.7** | 85.6 | 83.8 | 84.3 | 82.1 |
| | PUR | 21.5 | 21.1 | 18.7 | 27.3 | 21.6 | 26.1 | 24.7 | 25.0 | 24.1 | 30.2 | **31.8** | 28.4 | 26.3 | 27.0 | 23.4 |

## 4.2　Features, Kernels and Implementation Details

In [20], PCCA was applied to feature vectors made of 16-bins histograms from the RGB, YUV and HSV color channels, as well as texture histograms based on Local Binary Patterns extracted from 6 non-overlapping horizontal bands[2]. In the sequel we will refer to these features as the *band* features.

On the other hand, the authors in [21] applied LDFA to a set of feature vectors consisting of 8-bins histograms and 3 moments extracted from 6 color channels (RGB and HSV) over a set of 341 dense overlapping $8 \times 8$ pixel regions, defined every 4 pixels in both the horizontal and vertical directions, resulting in 11,253 dimensional vectors. These vectors were then compressed into 100 dimensional vectors using PCA before applying LDFA. In the sequel, we will refer to these features as the *block* features.

Even though the authors of [20] and [21] reported performance analysis using the same datasets, they used different sets of features to characterize the sample images. Thus, it is difficult to conclude whether the differences on the reported performances are due to the classification methods or to the feature selection. Therefore, in order to fairly evaluate the benefits of each algorithm and the effect of the choice of features, in our experiments we tested each of the algorithms using the *same* set of features. Moreover, while both band and block features are extracted within rectangular or square regions, their size and location are very different. Thus, to evaluate how these regions affect the re-identification accuracy, we run experiments varying their size and position. In addition to the

---

[2] Since the parameters for the LBP histogram and horizontal bands were not given in [20], we found values that provide even better matching accuracy than in [20].

**Table 3.** CMC at $r = 1, 5, 10, 20$ and PUR scores on CAVIAR with $p = 36$ test individuals (highest scores shown in red).

| | | PCCA | | | LFDA | SVMML | KISSME | rPCCA | | | kLFDA | | | MFA | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | CAVIAR | | | | | | | | | | | | | | |
| | $r$ | $L$ | $\chi^2$ | $R_{\chi^2}$ | $w/o$ | $w/o$ | $w/o$ | $L$ | $\chi^2$ | $R_{\chi^2}$ | $L$ | $\chi^2$ | $R_{\chi^2}$ | $L$ | $\chi^2$ | $R_{\chi^2}$ |
| 6 | 1 | 25.7 | 29.1 | 33.4 | 31.7 | 25.8 | 31.4 | 28.8 | 30.4 | 34.0 | 31.5 | 36.2 | 35.9 | 33.8 | 37.7 | **38.4** |
| | 5 | 57.9 | 62.5 | 67.2 | 56.1 | 61.4 | 61.9 | 61.3 | 63.6 | 67.5 | 55.4 | 64.0 | 63.6 | 62.0 | 67.2 | **69.0** |
| | 10 | 75.8 | 79.7 | 83.1 | 70.4 | 78.6 | 77.8 | 78.0 | 80.4 | 83.4 | 69.5 | 78.7 | 77.9 | 77.2 | 82.1 | **83.6** |
| | 20 | 92.0 | 94.2 | 95.7 | 86.9 | 93.6 | 92.5 | 93.2 | 94.5 | **95.8** | 86.1 | 92.2 | 91.2 | 92.1 | 94.6 | 95.1 |
| | PUR | 21.5 | 25.5 | 29.8 | 20.7 | 23.7 | 24.9 | 24.3 | 26.5 | 30.3 | 20.2 | 27.5 | 26.9 | 25.6 | 30.7 | **32.0** |
| 14 | 1 | 28.8 | 30.7 | 33.9 | 33.4 | 26.5 | 32.9 | 30.6 | 31.8 | 34.6 | 33.6 | 38.5 | 37.9 | 35.3 | **39.0** | 38.9 |
| | 5 | 62.3 | 64.8 | 67.8 | 58.8 | 62.1 | 64.0 | 64.0 | 65.9 | 68.5 | 59.1 | 66.7 | 67.0 | 63.8 | 68.6 | **69.7** |
| | 10 | 79.1 | 81.4 | 83.5 | 73.0 | 79.5 | 79.8 | 80.4 | 82.1 | **83.9** | 73.1 | 80.7 | 81.0 | 78.6 | 83.0 | 83.7 |
| | 20 | 94.0 | 94.9 | 95.6 | 88.4 | 94.2 | 93.4 | 94.5 | 95.0 | **95.8** | 88.5 | 93.3 | 92.7 | 92.8 | 94.8 | 94.9 |
| | PUR | 25.2 | 27.5 | 30.3 | 22.9 | 24.6 | 26.9 | 26.7 | 28.4 | 31.0 | 23.1 | 30.1 | 29.7 | 27.3 | 32.0 | **32.5** |
| 75 | 1 | 31.9 | 32.9 | 33.2 | 35.2 | 28.8 | 34.1 | 33.0 | 34.1 | 35.1 | 35.7 | 39.1 | 39.1 | 36.6 | **40.2** | 39.4 |
| | 5 | 65.2 | 66.3 | 65.9 | 59.9 | 63.1 | 64.9 | 66.0 | 67.1 | 67.2 | 62.6 | 66.8 | 68.4 | 65.5 | **70.2** | 69.7 |
| | 10 | 81.6 | 82.4 | 81.9 | 73.7 | 79.8 | 80.1 | 82.0 | 82.9 | 83.1 | 77.0 | 80.9 | 82.4 | 80.2 | **83.9** | 83.7 |
| | 20 | 95.3 | 95.5 | 95.2 | 88.8 | 93.9 | 93.0 | 95.4 | 95.5 | **95.6** | 91.4 | 93.4 | 94.3 | 93.3 | 95.1 | 95.0 |
| | PUR | 28.2 | 29.1 | 28.8 | 24.2 | 25.5 | 27.5 | 29.0 | 29.9 | 30.4 | 26.4 | 30.5 | 31.6 | 28.8 | **33.4** | 32.7 |
| 341 | 1 | 30.8 | 31.3 | 30.4 | 35.1 | 28.9 | 34.9 | 32.5 | 33.0 | 33.4 | 34.7 | 37.7 | 36.4 | 34.9 | **37.8** | 36.3 |
| | 5 | 64.6 | 64.1 | 62.2 | 59.4 | 62.5 | 64.7 | 64.9 | 65.3 | 64.4 | 62.0 | 65.9 | 65.6 | 64.5 | **67.9** | 66.4 |
| | 10 | 80.2 | 80.5 | 79.1 | 73.1 | 79.2 | 79.7 | 81.2 | 81.6 | 80.6 | 76.6 | 80.5 | 80.6 | 79.7 | **82.4** | 81.6 |
| | 20 | 94.6 | 94.7 | 93.6 | 88.2 | 93.3 | 93.3 | 94.9 | **95.0** | 94.3 | 91.2 | 93.6 | 93.6 | 93.3 | 94.6 | 94.2 |
| | PUR | 26.7 | 27.1 | 25.4 | 23.8 | 25.0 | 27.8 | 28.0 | 28.4 | 27.8 | 25.7 | 29.6 | 29.0 | 27.7 | **31.1** | 29.5 |

band and block features described above, we used a set of features extracted from $16 \times 16$ and $32 \times 32$ pixels overlapping square regions, similar to the ones used in the block features, but defined with a step half of the width/height of the square regions in both directions. Thus, a total of 75 and 14 regions were selected in these two feature sets. The feature vectors were made of 16-bins histogram of 8 color channels extracted on these image patches. To represented the texture patterns, 8-neighbors of radius 1 and 16-neighbors of radius 2 uniform LBP histograms were also computed for each region. Finally, the histograms were normalized with the $\ell_1$ norm in each channel and concatenated to form the feature for each image.

The projected feature space dimensionality was set to $d' = 40$ for the PCCA algorithm. To be fair, we also used $d' = 40$ with rPCCA. The parameter in the generalized logistic loss function was set to 3 for both PCCA and rPCCA. Since we could not reproduce the reported results of LFDA using their parameters setting, we set the projected feature space as 40 and the regularizing weight $\beta$ as 0.15 for LFDA[3]. In kLFDA, we used the same $d'$ and set the regularizing weight to 0.01. For MFA, we used all positives pairs of each person for the within class sets and set $k_b$ to 12, $\beta = 0.01$, and $d' = 30$. Since SVMML in [18] used different features, we also tuned the parameters to achieve results as good as possible. The two regularized parameters of A and B were set to $10^{-8}$ and $10^{-6}$, respectively. Since KISSME is very sensitive to the PCA dimensions, we chose the dimension for each dataset that gives best PUR and rank 1 CMC score, which are 77, 45, 65 and 70 for VIPeR, iLIDS, CAVIAR and 3DPeS, respectively. In the training

---

[3] It was set as 0.5 in [21]. However, we could not reproduce their reported results with this parameter.

**Table 4.** CMC at $r = 1, 5, 10, 20$ and PUR scores on 3DPeS with $p = 95$ test individuals (highest scores shown in red).

| | | 3DPeS | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | PCCA | | | LFDA | SVMML | KISSME | rPCCA | | | kLFDA | | | MFA | | |
| | $r$ | $L$ | $\chi^2$ | $R_{\chi^2}$ | $w/o$ | $w/o$ | $w/o$ | $L$ | $\chi^2$ | $R_{\chi^2}$ | $L$ | $\chi^2$ | $R_{\chi^2}$ | $L$ | $\chi^2$ | $R_{\chi^2}$ |
| 6 | 1 | 33.4 | 36.4 | 39.7 | 39.1 | 27.7 | 34.2 | 39.2 | 40.4 | 43.5 | 38.8 | 48.4 | **48.7** | 35.9 | 42.3 | 41.8 |
| | 5 | 63.5 | 66.3 | 68.4 | 61.7 | 58.5 | 58.7 | 68.3 | 69.5 | 71.6 | 62.0 | 72.5 | **73.7** | 58.5 | 65.3 | 65.5 |
| | 10 | 75.8 | 78.1 | 79.6 | 71.8 | 72.1 | 69.6 | 79.7 | 80.5 | 81.8 | 72.6 | 82.1 | **83.1** | 69.3 | 75.2 | 75.7 |
| | 20 | 86.9 | 88.6 | 89.5 | 82.6 | 84.1 | 80.2 | 89.3 | 90.0 | **91.0** | 82.7 | 89.9 | 90.7 | 79.9 | 84.8 | 85.2 |
| | PUR | 37.7 | 40.4 | 42.7 | 36.4 | 32.9 | 32.9 | 42.5 | 43.6 | 46.0 | 36.7 | 47.6 | **48.5** | 33.2 | 40.0 | 40.1 |
| 14 | 1 | 37.3 | 39.8 | 42.2 | 43.2 | 31.8 | 39.4 | 41.9 | 44.0 | 46.2 | 44.1 | 51.9 | **52.2** | 40.0 | 45.6 | 45.0 |
| | 5 | 67.4 | 69.6 | 71.1 | 65.3 | 63.0 | 63.1 | 71.3 | 72.6 | 74.7 | 66.5 | 75.1 | **75.9** | 62.6 | 69.0 | 68.3 |
| | 10 | 79.4 | 80.9 | 82.1 | 75.0 | 75.6 | 73.1 | 82.2 | 82.9 | 84.2 | 75.8 | 83.6 | **84.6** | 72.9 | 78.4 | 78.1 |
| | 20 | 89.3 | 89.8 | 90.5 | 84.3 | 86.1 | 82.2 | 90.6 | 91.0 | **91.5** | 84.7 | 90.9 | **91.5** | 82.9 | 87.1 | 86.9 |
| | PUR | 41.4 | 43.4 | 45.1 | 40.1 | 36.5 | 37.0 | 45.2 | 46.6 | 48.7 | 41.3 | 50.5 | **51.3** | 37.4 | 43.7 | 43.2 |
| 75 | 1 | 40.7 | 41.6 | 40.2 | 45.5 | 34.7 | 41.3 | 46.9 | 47.3 | 47.6 | 47.6 | **54.0** | 52.4 | 42.4 | 48.4 | 46.3 |
| | 5 | 70.3 | 70.5 | 68.4 | 69.2 | 66.4 | 66.2 | 74.5 | 75.0 | 74.6 | 71.8 | **77.7** | 77.1 | 66.8 | 72.4 | 70.5 |
| | 10 | 81.5 | 81.3 | 79.6 | 78.0 | 78.8 | 76.3 | 84.4 | 84.5 | 84.1 | 81.1 | **85.9** | 85.7 | 76.5 | 81.5 | 80.0 |
| | 20 | 90.7 | 90.4 | 89.3 | 86.1 | 88.5 | 85.3 | 91.8 | 91.9 | 91.7 | 88.8 | **92.4** | 92.4 | 86.0 | 89.8 | 89.1 |
| | PUR | 44.5 | 44.6 | 42.7 | 43.2 | 39.7 | 40.1 | 49.1 | 49.3 | 49.1 | 46.4 | **53.5** | 52.5 | 41.2 | 47.6 | 45.6 |
| 341 | 1 | 37.9 | 38.4 | 33.8 | 44.8 | 34.4 | 40.5 | 45.2 | 45.2 | 43.8 | 46.8 | **51.6** | 48.2 | 41.8 | 46.0 | 42.0 |
| | 5 | 67.2 | 66.9 | 61.8 | 68.6 | 65.9 | 66.2 | 72.8 | 72.6 | 70.5 | 72.5 | **76.4** | 73.9 | 66.6 | 70.6 | 66.5 |
| | 10 | 79.0 | 78.5 | 74.2 | 77.7 | 77.8 | 76.1 | 82.5 | 82.4 | 80.8 | 81.8 | **84.9** | 83.1 | 76.8 | 80.1 | 77.1 |
| | 20 | 89.1 | 88.5 | 85.4 | 86.0 | 87.8 | 85.7 | 90.8 | 90.6 | 89.5 | 89.5 | **92.0** | 91.0 | 86.2 | 89.0 | 86.3 |
| | PUR | 41.5 | 41.1 | 36.2 | 42.7 | 38.9 | 40.1 | 47.0 | 46.9 | 44.7 | 46.8 | **51.7** | 48.6 | 41.0 | 45.6 | 41.4 |

process for PCCA, rPCCA and KISSME, the number of negative pairs was set to 10 times the number of positive pairs. Finally, we tested three kernels with each algorithm and feature set: a linear, a $\chi^2$ and a $RBF - \chi^2$ kernel which are denoted with $L$, $\chi^2$ and $R_{\chi^2}$, respectively.

### 4.3 Performance Analysis

For both, the VIPeR and iLIDS datasets, the test sets were randomly selected using half of the available individuals. Specifically, there are $p = 316$, $p = 60$, $p = 36$, and $p = 95$ individuals in each of the test sets for the VIPeR, iLIDS, CAVIAR, and 3DPeS datasets, respectively. Figure 2 shows the best CMC curves for each algorithm on the four datasets. The results are also summarized in Tables 1 to 4, along with the PUR scores for all the experiments. The experiments show that the VIPeR dataset is more difficult than the iLIDS dataset. This can be explained by observing that VIPeR has only two images per individual, resulting in much lower $r = 1$ CMC scores. On the other hand, the overall PUR score is higher for the VIPeR set, probably because the iLIDS set has less than half of the images than VIPeR has.

The highest CMC and PUR scores in every experiment at every ranking were highlighted in red in the given table. The highest CMC and PUR scores were achieved using the proposed methods with either a $\chi^2$ or a $R_{\chi^2}$ kernel. The proposed approaches achieve as much as 19.6% at $r = 1$ and a 10.3% PUR score improvement on the VIPeR dataset, 14.6% at $r = 1$ and a 31.2% PUR score improvement on the iLIDS dataset, 15.0% at $r = 1$ and a 7.4% PUR score improvement on the CAVIAR dataset and 22.7% at $r = 1$ and a 13.6% PUR score improvement on the 3DPeS dataset, when using band features (6 bands).

**Table 5.** The best reported CMC scores in the existing literature.

| | VIPeR | | iLIDS | | CAVIAR | | 3DPeS | |
|---|---|---|---|---|---|---|---|---|
| | SVMML [18] | kLFDA | PRDC [31] | kLFDA | LFDA [21] | MFA | LFDA [21] | kLFDA |
| $r = 1$ | 30.0 | **32.3** | 37.83 | **38.0** | 32.0 | **40.2** | 33.43 | **54.0** |
| $r = 5$ | 65.0 | **65.8** | 63.7 | **65.1** | 56.3 | **70.2** | | 77.7 |
| $r = 10$ | **80.0** | 79.7 | 75.09 | **77.4** | 70.7 | **83.9** | 69.98 | **85.9** |
| $r = 20$ | **91.0** | 90.9 | 88.35 | **89.2** | 87.4 | **95.1** | | 92.4 |
| PUR | | 52.5 | | 35.4 | 21.2 | **33.4** | 34.85 | **53.5** |

In general, rPCCA performed better than LFDA which, in turn, performed better than PCCA. The better performance of rPCCA over PCCA and LFDA shows that the regularizer term $\|\mathbf{P}\|_F$ plays a significant role in preventing over-fitting of noisy data. However, the best performance is achieved by kLFDA because this approach does a better job at selecting the features by avoiding the PCA pre-processing step while taking advantage of the locally scaled affinity matrix.

It should be noted that using 6, 14, 75 and 341 regions results in similar performance, but using 341 results in slightly lower PUR scores. Moreover, the RBF-$\chi^2$ kernel does not help improving the matching accuracy when the regions are small. It was observed in our experiments that the $\chi^2$ distance of the positive and negative pairs were distributed within a small range around 1 and that the kernel mapping of these values were hard to distinguish. A possible explanation for this effect, is that the histograms are noisier and sparser when the base regions are small.

For sake of completeness, we also compared the best performance for the proposed algorithms against the best results as reported in the existing literature (even though as pointed above, the values reported elsewhere do not use the same set of features or experimental protocol) [1, 2, 6, 11, 16–21, 28, 29, 31] in Table 5.

Our algorithm matches the best reported results for the VIPeR and iLIDS datasets, even though the reported PRDC [31] ranking was obtained under easier experiment settings[4]. Note that both SVMML[5] [18] and PRDC require an iterative optimization which is very expensive on both computation and memory. In comparison, computing the closed-form solution for the proposed kLFDA and MFA algorithms is much cheaper. When using a 3.8Hz Intel quad-core computer with 16GB RAM, the average training times for VIPeR, using 6 patches with a linear kernel are 0.24s, 0.22s and 155.86s for kLFDA, MFA and SVMML, respectively. While average training times for the iLIDS are 0.07s, 0.04s and 155.6s for kLFDA, MFA and PRDC, respectively. In the experiments on the CAVIAR and 3DPeS datasets, our ranking is more accurate than LFDA algorithm[6].

---

[4] Only 50 individuals were selected as test, while our test set is composed of 60 individuals. Thus, the ranking accuracy is computed in an easier experiment setting.

[5] The ranking accuracy was read from the figure.

[6] The CAVIAR ranking reported in [21] was obtained by using the mean of the features from the sample person in the test set as the reference feature. We believe this is equivalent to knowing the ground truth before ranking. Hence we report the result in Table 5 via following our protocol but using the same features as in [21].

Finally, Table 6 shows the results for ranking ensembles voting using different learning algorithms, feature sets, kernels, and aggregating methods. Since the features extracted from $8 \times 8$ pixels regions provided the worst performance for almost all the algorithms, we do not use this set of features in the ensemble. Therefore, for each metric learning algorithm, we created an ensemble with 9 ranking algorithms, combining 3 kernels (if applicable) and 3 feature sets, which were used to vote for a final ranking. The best performances of the individual ranking case for each of the metric learning methods from Tables 1 to 4 are also shown (with a gray background) for easy comparison. The experimental results show that the ensemble methods produced different level of improvements for each dataset and in general "Ensemble 1" results in larger gains. For single ensemble metric learning algorithm, the performance of ensemble rPCCA improved from 1.56% to 7.91% across all four datasets whereas the ensemble kLFDA benefited much less. The performance on iLIDS datasets improved on all experiments whereas the ones on 3DPeS decreased for ensemble kLFDA and MFA. Since the images in the iLIDS dataset have severe occlusions, using an ensemble of different feature sets is beneficial with this dataset. The highest improvement is all algorithms ensemble on CAVIAR dataset, the rank1 score increased 4.73% and the PUR score increased 8.08% These results suggest that combining different feature grids can improve the performance.

**Table 6.** CMC scores of ensembles of rPCCA, kLFDA, MFA on all four datasets. The columns with gray background show the performance of the best ranking algorithm in this category (highest scores shown in red).

| | $r$ | VIPeR | Ensb 1 | Ensb 2 | iLIDS | Ensb 1 | Ensb 2 | CAVIAR | Ensb 1 | Ensb 2 | 3DPeS | Ensb 1 | Ensb 2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| rPCCA | 1 | 22.0 | 23.7 | **23.9** | 29.6 | 32.6 | **32.7** | 34.6 | 37.3 | **37.4** | 47.3 | 49.8 | **50.0** |
| | 5 | 54.8 | 55.3 | **55.7** | 57.3 | 59.4 | **59.8** | 68.5 | 69.5 | **70.0** | 75.0 | 76.1 | **76.6** |
| | 10 | 71.0 | 71.7 | **72.3** | 71.7 | 73.3 | **73.5** | 83.9 | 84.6 | **84.7** | 84.5 | 85.4 | **85.5** |
| | 20 | 85.3 | 85.4 | **86.0** | 85.9 | 86.8 | **86.9** | 95.8 | 96.2 | **96.3** | 91.9 | **92.6** | 92.4 |
| | PUR | 44.8 | **45.5** | 44.7 | 27.8 | 29.9 | **30.0** | 31.0 | 32.7 | **32.8** | 49.3 | 51.1 | **51.3** |
| kLFDA | 1 | 32.3 | **32.8** | 31.8 | 38.0 | 40.2 | **40.3** | 39.1 | **39.4** | 39.1 | **54.0** | 53.1 | 52.6 |
| | 5 | **65.8** | 65.5 | 64.6 | 65.1 | 66.0 | **66.7** | **68.4** | 67.2 | 66.9 | **77.7** | 76.1 | 76.2 |
| | 10 | **79.7** | 79.1 | 78.4 | 77.4 | **78.1** | 78.1 | **82.4** | 81.5 | 81.0 | **85.9** | 84.7 | 84.7 |
| | 20 | 90.9 | 90.0 | 89.3 | 89.2 | **89.6** | 89.6 | **94.3** | 93.8 | 93.6 | **92.4** | 91.4 | 91.5 |
| | PUR | **52.5** | 51.9 | 49.6 | 35.4 | **36.7** | 36.7 | **31.6** | 31.0 | 30.6 | **53.5** | 51.8 | 51.6 |
| MFA | 1 | 32.2 | **34.1** | 33.2 | 33.7 | 36.8 | **37.0** | 40.2 | **41.5** | 41.4 | **48.4** | 48.2 | 47.9 |
| | 5 | 66.0 | **66.5** | 66.1 | 59.3 | 61.3 | **61.7** | 70.2 | **70.8** | 70.7 | **72.4** | 71.3 | 71.2 |
| | 10 | 79.7 | **80.1** | 79.7 | 71.7 | **73.8** | 73.6 | 83.9 | **85.0** | 84.9 | **81.5** | 80.9 | 80.7 |
| | 20 | **90.6** | 90.3 | 89.8 | 86.5 | 87.3 | **87.5** | 95.1 | **95.4** | 95.4 | **89.8** | 89.0 | 88.7 |
| | PUR | 52.4 | **52.8** | 50.9 | 30.3 | 32.3 | **32.5** | 33.4 | **34.4** | 34.4 | **47.6** | 46.6 | 46.3 |
| kLFDA + rPCCA + MFA | 1 | 32.3 | **33.9** | 32.7 | 38.0 | **39.4** | 39.0 | 40.2 | **41.8** | 41.5 | 54.0 | **54.2** | 53.2 |
| | 5 | 65.8 | **67.0** | 66.1 | **65.1** | 65.0 | 65.1 | 70.2 | **72.0** | 71.7 | **77.7** | 77.7 | 77.5 |
| | 10 | 79.7 | **80.5** | 79.6 | 77.4 | 76.9 | **77.5** | 83.9 | **85.8** | 85.5 | 85.9 | **86.1** | 85.8 |
| | 20 | 90.9 | **90.6** | 88.4 | **89.2** | 89.0 | 88.7 | 95.1 | **96.4** | 96.2 | 92.4 | **92.8** | 92.3 |
| | PUR | 52.5 | **53.1** | 49.0 | 35.4 | **35.6** | 35.1 | 33.4 | **35.7** | 35.3 | 53.5 | **53.8** | 52.7 |
| All | 1 | 32.3 | 35.1 | **36.1** | 38.0 | **39.8** | 39.4 | 40.2 | **42.1** | 41.7 | 54.0 | **54.1** | 53.4 |
| | 5 | 65.8 | 68.2 | **68.7** | 65.1 | **65.3** | 65.2 | 70.2 | **72.2** | 72.0 | **77.7** | 77.7 | 77.4 |
| | 10 | 79.7 | **81.3** | 80.1 | **77.4** | 77.1 | 77.4 | 83.9 | **86.2** | 85.9 | 85.9 | **86.0** | 85.9 |
| | 20 | 90.9 | **91.1** | 85.6 | **89.2** | 89.2 | 88.4 | 95.1 | **96.5** | 96.4 | 92.4 | **92.6** | 92.0 |
| | PUR | 52.5 | **53.9** | 48.8 | 35.4 | **35.9** | 35.1 | 33.4 | **36.1** | 35.6 | 53.5 | **53.6** | 52.6 |
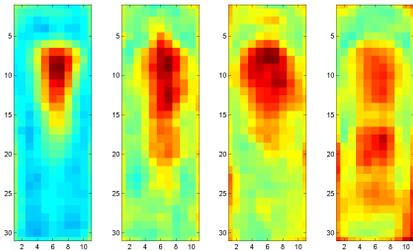
**Fig. 3.** The kLFDA projection weight map for 3DPeS, CAVIAR, iLIDS and VIPeR.

**Fig. 4.** View point variation in 3DPeS.

### 4.4   Dataset Analysis

Figure 3 shows a heat map illustrating the projection weight map for each of the datasets when using kLDFA with 341 patches and a linear kernel. There, it is seen that the upper body features are the most discriminant ones in all four datasets. This is expected since the bounding-boxes of the samples are reasonably accurate and the torsos are relatively well aligned. On the other hand, the feature projection weights at the bottom of the sample are different across the four datasets. This can be explained by the fact that the viewpoint variations in the 3DPeS dataset are the most severe among all the datasets. As shown in Figure 4, when looking from a top view the legs for the pedestrians occupy fewer pixels and their locations change more than when seen from an horizontal viewpoint as is the case for the VIPeR samples.

Moreover, the projection weights for the VIPeR dataset are larger for patches in the background than for the other three datasets. This reflects the fact that the VIPeR samples were taken in three different scenes, walk way through a garden, play ground and street side way with distinctive backgrounds and that the two images for each person were always taken in the same scene.

## 5   Conclusion

We proposed and evaluated the performance of four alternatives for re-ID classification: rPCCA, kLFDA, MFA and two ranking ensemble voting (REV) schema, used in conjunction with sets of histogram-based features and linear, $\chi^2$ and RBF-$\chi^2$ kernels. Comparison against four state-of-the-art approaches (PCCA, LDFA, SVMML and KISSME) showed consistently better performance and up to a 19.6%, 14.6%, 15.0% and 22.7% accuracy improvements at rank 1 and 10.3%, 31.2%, 7.4% and 13.6% PUR scores improvements, on the VIPeR, iLIDS, CAVIAR and 3DPeS datasets, respectively, when using 6 bands as support regions for the extracted features and using an RBF-$\chi^2$ kernel with the kLFDA and MFA approaches. With the *Ensemble 1* voting schema, we can further increase accuracy by 8.7%, 4.7%, 4.7% at rank 1 and by 2.7%, 1.4%, 8.1% at PUR on the VIPeR, iLIDS, CAVIAR datasets, respectively.

# References

1. An, L., Kafai, M., Yang, S., Bhanu, B.: Reference-based person re-identification. In: Advanced Video and Signal Based Surveillance (AVSS), 2013 10th IEEE International Conference on. pp. 244–249 (Aug 2013)
2. Bak, S., Corvee, E., Brémond, F., Thonnat, M.: Person re-identification using spatial covariance regions of human body parts. In: Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on. pp. 435–440. IEEE (2010)
3. Bak, S., Corvee, E., Bremond, F., Thonnat, M.: Multiple-shot human re-identification by mean riemannian covariance grid. In: Advanced Video and Signal-Based Surveillance (AVSS), 2011 8th IEEE International Conference on. pp. 179–184. IEEE (2011)
4. Baltieri, D., Vezzani, R., Cucchiara, R.: 3dpes: 3d people dataset for surveillance and forensics. In: Proceedings of the 1st International ACM Workshop on Multimedia access to 3D Human Objects. pp. 59–64. Scottsdale, Arizona, USA (Nov 2011)
5. Bauml, M., Stiefelhagen, R.: Evaluation of local features for person re-identification in image sequences. In: Advanced Video and Signal-Based Surveillance (AVSS), 2011 8th IEEE International Conference on. pp. 291–296. IEEE (2011)
6. Bazzani, L., Cristani, M., Murino, V.: Symmetry-driven accumulation of local features for human characterization and re-identification. Computer Vision and Image Understanding 117(2), 130–144 (2013)
7. Bedagkar-Gala, A., Shah, S.K.: A survey of approaches and trends in person re-identification. Image and Vision Computing (2014)
8. Cheng, D.S., Cristani, M., Stoppa, M., Bazzani, L., Murino, V.: Custom pictorial structures for re-identification. In: British Machine Vision Conference (BMVC). pp. 68.1–68.11 (2011)
9. Dantcheva, A., Dugelay, J.L.: Frontal-to-side face re-identification based on hair, skin and clothes patches. In: Advanced Video and Signal-Based Surveillance (AVSS), 2011 8th IEEE International Conference on. pp. 309–313 (Aug 2011)
10. Doretto, G., Sebastian, T., Tu, P., Rittscher, J.: Appearance-based person reidentification in camera networks: problem overview and current approaches. Journal of Ambient Intelligence and Humanized Computing 2(2), 127–151 (2011)
11. Farenzena, M., Bazzani, L., Perina, A., Murino, V., Cristani, M.: Person re-identification by symmetry-driven accumulation of local features. In: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. pp. 2360–2367. IEEE (2010)
12. Gheissari, N., Sebastian, T., Hartley, R.: Person reidentification using spatiotemporal appearance. In: CVPR (2006)
13. Gong, S., Cristani, M., Yan, S., Loy, C.C.: Person Re-Identification. Springer-Verlag London (2014)
14. Gray, D., Tao, H.: Viewpoint invariant pedestrian recognition with an ensemble of localized features. In: Computer Vision–ECCV 2008, pp. 262–275. Springer (2008)
15. Kostinger, M., Hirzer, M., Wohlhart, P., Roth, P.M., Bischof, H.: Large scale metric learning from equivalence constraints. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. pp. 2288–2295. IEEE (2012)
16. Kuo, C.H., Khamis, S., Shet, V.: Person re-identification using semantic color names and rankboost. Applications of Computer Vision, IEEE Workshop on 0, 281–287 (2013)

17. Li, W., Wang, X.: Locally aligned feature transforms across views. In: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on. pp. 3594–3601 (June 2013)
18. Li, Z., Chang, S., Liang, F., Huang, T.S., Cao, L., Smith, J.R.: Learning locally-adaptive decision functions for person verification. In: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on. pp. 3610–3617. IEEE (2013)
19. Loy, C.C., Liu, C., Gong, S.: Person re-identification by manifold ranking. In: IEEE International Conference on Image Processing. vol. 20 (2013)
20. Mignon, A., Jurie, F.: Pcca: A new approach for distance learning from sparse pairwise constraints. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. pp. 2666–2672. IEEE (2012)
21. Pedagadi, S., Orwell, J., Velastin, S., Boghossian, B.: Local fisher discriminant analysis for pedestrian re-identification. In: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on. pp. 3318–3325. IEEE (2013)
22. Sugiyama, M.: Local fisher discriminant analysis for supervised dimensionality reduction. In: 23rd international conference on Machine learning. pp. 905–912. ACM (2006)
23. Vezzani, R., Baltieri, D., Cucchiara, R.: People reidentification in surveillance and forensics: A survey. ACM Computing Surveys (CSUR) 46(2), 29 (2013)
24. Wang, L., Tan, T., Ning, H., Hu, W.: Silhouette analysis-based gait recognition for human identification. Pattern Analysis and Machine Intelligence, IEEE Transactions on 25(12), 1505–1518 (2003)
25. Wang, X., Doretto, G., Sebastian, T., Rittscher, J., Tu, P.: Shape and appearance context modeling. In: CVPR (2007)
26. Yan, S., Xu, D., Zhang, B., Zhang, H.J., Yang, Q., Lin, S.: Graph embedding and extensions: a general framework for dimensionality reduction. Pattern Analysis and Machine Intelligence, IEEE Transactions on 29(1), 40–51 (2007)
27. Zhang, T., Oles, F.: Text categorization based on regularized linear classification methods. Information Retrieval 4, 5–31 (2001)
28. Zhao, R., Ouyang, W., Wang, X.: Person re-identification by salience matching. ICCV (2013)
29. Zheng, W., Gong, S., Xiang, T.: Re-identification by relative distance comparison. PAMI 35(3), 653–668 (2013)
30. Zheng, W.S., Gong, S., Xiang, T.: Associating groups of people. In: BMVC (2009)
31. Zheng, W.S., Gong, S., Xiang, T.: Person re-identification by probabilistic relative distance comparison. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on. pp. 649–656. IEEE (2011)