Person Re-identification by Smooth Metric Learning

Yimin Wang[1], Ruimin Hu[1], Chao Liang[1], Chunjie Zhang[2], and Qingming Leng[1]

[1] National Engineering Research Center for Multimedia Software, School of Computer, Wuhan University, Wuhan, 430072, China
[2] School of Computer and Control Engineering, University of Chinese Academy of Sciences, Beijing, 100190, China

Abstract. Person re-identification, which is to match people across areas covered by multiple non-overlapping surveillance cameras, has drawn great research interests in the video surveillance domain. Previous research mainly focus on feature extraction and distance measure, where the former aims to find robust feature representation while the latter seeks to learn an optimal metric space. Because of the introduction of supervised information, metric learning methods can usually achieve better performance over feature based methods. As one of the most representative metric learning method, the Large Margin Nearest Neighbor (LMNN) algorithm was recently applied in person re-identification task and achieved satisfactory results [1]. However, LMNN uses a standard hinge loss function, which is neither differentiable everywhere nor time-efficiency due to the usage of all training samples. In this paper, we propose to replace hinge loss function with the logistic loss function, which transforms LMNN to a smooth unconstrained convex optimization problem easily solved with gradient descent algorithm. Thereafter, we further design a stochastic sampling scheme to accelerate the optimization process of the above problem with randomly selected training samples. Extensive comparative experiments conducted on two standard datasets have shown the effectiveness and efficiency of the proposed algorithm over a series of standard baseline methods.

Keywords: Person re-identification, Surveillance video, Multiple cameras, Metric learning.

1 Introduction

Recently, increasing number of surveillance camera networks have been set up for monitoring pedestrian activities over a large public areas, such as airports, metro stations and parking lots. In these scenarios, matching person across non-overlapping cameras in a surveillance camera network, a.k.a, person re-identification, is becoming a hot research spot in the computer vision community. During the past five years, a lot of research effort has been devoted to this field [2–4, 1, 5]. However, due to significant changes of view angles, scales

B. Huet et al. (Eds.): PCM 2013, LNCS 8294, pp. 561–573, 2013.

Fig. 1. Some typical samples from VIPeR dataset [6] where appearance changes are mainly caused by variant views and lighting conditions. Each column shows two images of the same person captured by two different cameras.

and illumination conditions (see Fig. 1), different persons captured by the same camera usually appear more alike than the same person across different cameras, making person re-identification remain a unsolved and challenging problem.

Generally speaking, person re-identification can be considered as a visual retrieval problem [5], i.e. given a query person image taken in one camera, the algorithm is expected to search images of the same person captured by other cameras. Typically, it consists of two stages: feature extraction and distance measure. The former aims to seek a discriminative and robust feature representation which can easily separate different persons in various cameras. Wang et al. [7] used a co-occurrence matrix to capture the spatial distribution of the appearance relative to each of object parts to modeling the appearance of people. Farenzena et al. [2] divided the image of person into 5 regions by exploiting symmetry and asymmetry perceptual principles, and then combine multiple color and texture features to represent the appearance of people. However, in the condition of significant intra-object appearance variation cased by severe viewing changes, computing a set of features that are both distinctive and stable is extremely difficult [4].

In contrast, distance learning based methods focus on the second stage, distance measure, and aim to seek a suitable distance metric function through supervised learning. Gray and Tao [8] transformed the matching problem into a classification problem, in which they assigned a pair of images belonging to the same individual with positive sample or negative sample otherwise and learn the classifier using AdaBoost. The essence of the method is learning a weight vector assigning higher weight to the more discriminative feature. Recently, metric learning methods targeting for a matrix rather than a vector are exploited for person re-identification problem and are reported better performance. Hizer et al. [1] learned an optimal distance metric through LMNN algorithm, which was designed for k Nearest Neighbor classification [9]. The core idea of LMNN algorithm is that the feature distance of the same individual is small while that of different persons is large under the learned metric. However, standard LMNN algorithm uses a hinge loss function that is not differentiable everywhere resulting in difficulty of solution. Instead, Zheng et al. [4] used a logistic loss function and got a smooth objective function to learn the metric, in which a probabilistic relative distance comparison (PRDC) constrain was exploited to learn a low rank projection matrix. Recently, Kostinger et al. [5] developed a simpler algorithm to achieve an analytical form of the learned metric with the assumption that all samples coming from the Gaussian distribution.

In this paper, we focus on the distance learning based methods. Our idea is motivated from LMNN algorithm, which is known as the-state-of-the-art metric learning algorithm. However, according to the above discussion, LMNN is difficult to solve due to the usage of a hinge loss function, which is not differentiable everywhere. Instead, we use a logistic loss function, a soft approximation to original hinge loss function, to improve the standard LMNN algorithm, which transforms the original problem to a smooth unconstrained convex optimization problem. In addition, the new model is more flexible and achieves better performance with seeking a suitable approximation parameter. Besides, the LMNN algorithm is time consuming, even though an active set strategy was exploited in [9] to improve the efficiency. Motivated by Stochastic Gradient Descent (SGD) algorithm [10], we further propose a stochastic sampling based gradient descent algorithm. In particular, for each positive examples, we randomly selected some negative samples to consist real triple training samples. Extensive experiment results show that the proposed method is superior than the standard LMNN method in both accuracy and efficiency.

2 The Approach

This section presents our approach. We begin with terms and notations used in metric learning based person re-identification. Then, technical details of improved LMNN using logistic loss function is discussed and a stochastic sampling strategy is given to accelerate the algorithm implementation.

2.1 Person Re-identification Based on Metric Learning

For the convenience of presentation, we consider a pair of cameras C_a and C_b covering different area without overlapping, and $O_a = \{o_a^1, o_a^2, \cdots, o_a^m\}$ and $O_b = \{o_b^1, o_b^2, \cdots, o_b^n\}$ are images of persons captured by C_a and C_b, respectively. The task of person re-identification is that finding images of the same person from O_b for each instance o_a in O_a. Usually, the algorithm executes as follows: First, a d-dimensional feature vector is extracted to represent each instance. Second, for each pair images, a specialized distance function, such as Euclidean distance, is defined to computing the distance between the feature vectors. Finally, a ranking list is obtained with top positions corresponding to more alike the query person images.

Specially, let $\mathbf{x} = (x_1, \cdots, x_d)$ represent the d-dimensional feature vector of of a person o, then the Euclidean distance between o_a^i and o_b^j can be formulated as

$$D(o_a^i, o_b^j) = D(\mathbf{x}_a^i, \mathbf{x}_b^j) = \left\|\mathbf{x}_a^i - \mathbf{x}_b^j\right\|_2^2 = (\mathbf{x}_a^i - \mathbf{x}_b^j)^\top (\mathbf{x}_a^i - \mathbf{x}_b^j) \tag{1}$$

where $(\cdot)^\top$ represents the transpose of a vector or matrix. With the above notation, the formulation of metric learning can be denoted as:

$$D_{\mathbf{M}}(\mathbf{x}_a^i, \mathbf{x}_b^j) = (\mathbf{x}_a^i - \mathbf{x}_b^j)^\top \mathbf{M}(\mathbf{x}_a^i - \mathbf{x}_b^j) \tag{2}$$

where $\mathbf{M}$ is a positive semi-definite matrix for the validity of metric. In this case, $\mathbf{M}$ can be factored into real-valued matrices as $\mathbf{M} = \mathbf{L}^\top \mathbf{L}$ [11], and the Eq. (2) can be rewritten as

$$D_{\mathbf{L}}(\mathbf{x}_a^i, \mathbf{x}_b^j) = \left\|\mathbf{L}(\mathbf{x}_a^i - \mathbf{x}_b^j)\right\|_2^2 \tag{3}$$

The purpose of metric learning is to learn the positive semi-define matrix $\mathbf{M}$ or the real-valued matrix $\mathbf{L}$.

2.2 The Smooth LMNN Method

Before going further, we first briefly review the classic LMNN metric learning algorithm [9]. LMNN learns a real-valued matrix $\mathbf{L}$ under two constrain terms. One pulls the same labeled examples closer together, the other pushes examples with different labels further apart.

The *pull* term penalizes large distances between each instance and its target match, the formulation of which is given by:

$$\varepsilon_{pull}(\mathbf{L}) = \sum_{i,j \rightsquigarrow i} D_{\mathbf{L}}(\mathbf{x}_a^i, \mathbf{x}_b^j) \tag{4}$$

where $j \rightsquigarrow i$ means that $\mathbf{x}_b^j$ has the same label as $\mathbf{x}_a^i$.

The *push* term penalizes small distances between differently labeled examples. In particular, the term penalizes the differently labeled examples that invalidate the inequality in Eq. (5), named as *imposter*:

$$D_{\mathbf{L}}(\mathbf{x}_a^i, \mathbf{x}_b^l) \geq D_{\mathbf{L}}(\mathbf{x}_a^i, \mathbf{x}_b^j) + 1 \tag{5}$$

where $\mathbf{x}_a^i$ and $\mathbf{x}_b^j$ have the same label (belong to the same person), while $\mathbf{x}_b^l$ has another label. The formulation of *push* term is given by:

$$\varepsilon_{push}(\mathbf{L}) = \sum_{i,j \rightsquigarrow i}^{m} \sum_{l} (1 - y_{il})[1 + D_{\mathbf{L}}(\mathbf{x}_a^i, \mathbf{x}_b^j) - D_{\mathbf{L}}(\mathbf{x}_a^i, \mathbf{x}_b^l)]_+ \tag{6}$$

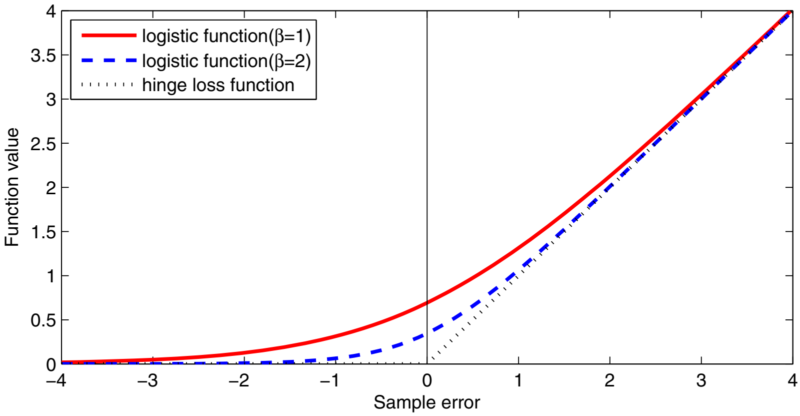


Fig. 2. Explanation for that the logistic loss gives a soft approximation to hinge loss, where the β is larger, the logistic loss is more near to hinge loss

where y_{il} is an indicator variable that equals to 1 if and only if $\mathbf{x}_a^i$ and $\mathbf{x}_b^l$ having the same label, and 0 otherwise; $[z]_+ = max(z, 0)$ denotes the standard hinge loss function.

Finally, the two terms $\varepsilon_{pull}(\mathbf{L})$ and $\varepsilon_{push}(\mathbf{L})$ are combined into an integrated loss function for distance metric learning:

$$\varepsilon(\mathbf{L}) = (1-\mu)\varepsilon_{pull}(\mathbf{L}) + \mu\varepsilon_{push}(\mathbf{L}) \tag{7}$$

where $\mu \in [0, 1]$ is the balance parameter.

Note that, the hinge loss $[z]_+$ is not differentiable at $z = 0$, which can be replaced with a smooth logistic loss function as shown in Fig. 2. The new loss function can be rewritten as

$$\varepsilon_{\text{LMNN-S}}(\mathbf{L}) = (1-\mu)\varepsilon_{pull}(\mathbf{L}) + \mu\varepsilon^*_{push}(\mathbf{L}) \tag{8}$$

$$\varepsilon^*_{push}(\mathbf{L}) = \sum_{i,j \rightsquigarrow i}^{m} \sum_{l} (1-y_{il})\ell_\beta(1 + D_{\mathbf{L}}(\mathbf{x}_a^i, \mathbf{x}_b^j) - D_{\mathbf{L}}(\mathbf{x}_a^i, \mathbf{x}_b^l)) \tag{9}$$

where $\ell_\beta(x) = \frac{1}{\beta}\log(1 + e^{\beta x})$ is the generalized logistic loss function. It is easy to see from the Fig. 2 that the logistic loss gives a soft approximation to hinge loss. The parameter of logistic loss β is the approximation parameter. As larger as the β is, the logistic is more near to hinge loss. The LMNN-S improves the performance with seeking a more suitable β (see the experiments results in Sec. 3.4).

Since the logistic loss function is convex and differentiable everywhere, Eq. (9) is a smooth convex optimization problem with respect to $\mathbf{L}$ and hence Eq. (9) is smooth convex optimization problem as quadratic term is smooth convex.

Algorithm 1. Learning the matrix $\mathbf{L}$

Input: The training set data: Positive samples with pair form $S_p = \{(\mathbf{x}_p^i, \mathbf{x}_p^j)_k\}$, Negative Samples with triple form $S_n = \{(\mathbf{x}_n^i, \mathbf{x}_n^j, \mathbf{x}_n^l)'_k\}$

```
Initialize L_0 as identical matrix;
for i = 1 to MaxIter do
    Compute ∇ε(L) = ∇ε_pull(L) + ∇ε_push(L) as Eq.(10-12)
    Choose a proper step λ
    Compute L_{i+1} = L_i − λ∇ε(L) as Eq.(13)
    if converge then
        break;
    end if
end for
```

Output: The optimal matrix $\mathbf{L}^*$

2.3 Stochastic Sampling Based Optimization Algorithm

As the above discussion, we transform the problem to a smooth convex optimization problem, which can be solved with a simple gradient descent method. However, it is time consuming with massive training samples. In particular, assume there are N different persons, the size of positive samples is $O(N)$. In addition, for each positive samples, there are $O(N-1)$ negative samples and hence the size of total training samples is $O(N^2)$. In [9], an active set strategy was exploited to improve the efficiency. The active set consists of the *imposter* samples which is relatively little. Even so, it is time consuming for using the k closet within-class samples [12].

SGD algorithm was designed to solve large scale linear prediction [10] where the gradient was compute on randomly selected samples. Motivated by SGD algorithm, we propose a stochastic sampling based gradient descent algorithm. In particular, for each positive examples, we randomly selected $k \ll N$ negative samples, with which the size of training set reduce to $O(kN)$ from $O(N^2)$.

With stochastic sampling strategy, a simple gradient-descent method is exploited to learn the projection matrix $\mathbf{L}$. The detail is as follows: The gradient of the loss function $\varepsilon(\mathbf{L})$ is given as

$$\frac{\partial \varepsilon_{\text{LMNN-S}}(\mathbf{L})}{\partial \mathbf{L}} = \frac{\partial \varepsilon_{pull}(\mathbf{L})}{\partial \mathbf{L}} + \frac{\partial \varepsilon_{push}(\mathbf{L})}{\partial \mathbf{L}} \tag{10}$$

$$\frac{\partial \varepsilon_{pull}(\mathbf{L})}{\partial \mathbf{L}} = 2L \sum_{i,j \rightsquigarrow i}^{m} (\mathbf{x}_a^i - \mathbf{x}_b^j)(\mathbf{x}_a^i - \mathbf{x}_b^j)^\top \tag{11}$$

$$\frac{\partial \varepsilon_{pull}(\mathbf{L})}{\partial \mathbf{L}} = 2L \sum_{i,j \rightsquigarrow i}^{m} \sum_{l} g(\varepsilon_{pull}(\mathbf{L}))((\mathbf{x}_b^j)(\mathbf{x}_b^j)^\top - (\mathbf{x}_b^l)(\mathbf{x}_b^l)^\top - 2(\mathbf{x}_b^j - \mathbf{x}_b^l)(\mathbf{x}_a^i)^\top) \tag{12}$$

where $g(x) = (1 + e^{-\beta x})^{-1}$ is the derivative of logistic loss function $\ell_\beta(x)$.

With the gradient, an iterative optimization algorithm is used. Starting from an initial identical matrix, which means no projection, $\mathbf{L}$ is optimized iteratively as follows

$$\mathbf{L}_{i+1} = \mathbf{L}_i - \lambda \cdot \frac{\partial \varepsilon(\mathbf{L}_i)}{\partial \mathbf{L}_i} \tag{13}$$

where $\lambda > 0$ is a step length automatically determined at each updating step using a similar strategy in [4]. The iteration is terminated when the updating times is greater than the maximum iterative times (i.e. 1000 in this work) or the following criterion is met.

$$\mid \varepsilon_{i+1} - \varepsilon_i \mid < \epsilon \tag{14}$$

where ϵ is a small positive value and fixed to 10^{-9} in this paper. The pseudo-code of algorithm is given in Algorithm 1.

3 Experimental Results

In this section, we evaluated the proposed approach for person re-identification on two widely used datasets, the VIPeR dataset [6] and the i-LIDS Multiple-Camera Tracking Scenario (MCTS) dataset [3]. First, we compared our method with several state-of-the-art person re-identification methods on both datasets, under the similar experiment settings. Moreover, we validate the influence of performance with different algorithm parameters on VIPeR dataset.

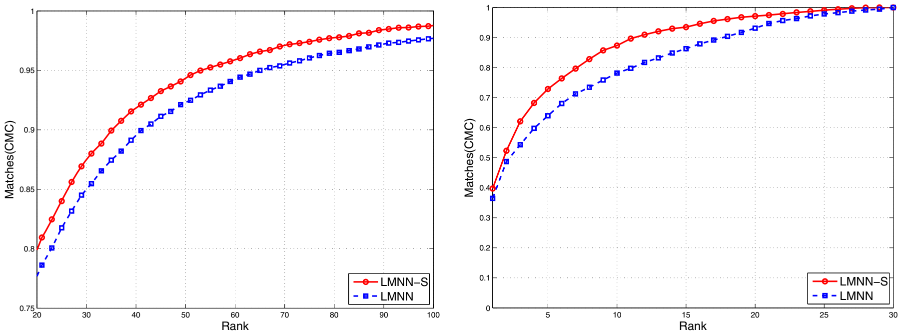
3.1 Datasets

The VIPeR dataset is collected by Gray et al. [8] and contains 632 persons with two images for each person taken from two non-overlapping outdoor cameras. All images of individuals are normalized to a size of 128×48 pixels. Example images are shown in Fig. 1, in which each column contains two images of the same person under different cameras. As shown, view changes are the most significant cause of appearance change, and most of the examples contain a viewpoint of 90 degrees. Besides, other variations are also considered, such as illumination conditions.

The i-LIDS MCTS dataset [3] is taken at a busy airport hall with several disjoint indoor cameras. There are 119 individuals with 476 images in all. As VIPeR dataset, all images are normalized to 128×48 pixels. The differences are that the light variances are more and some examples subject to large occlusion.

3.2 Image Representation

A combination feature descriptor consisting of color and texture features is used to represent images of individuals. Specifically, for each image, the RGB and HSV color histograms and LBP descriptor are extracted from overlapping blocks of size 16×16 and stride of 8×8, i.e. 50% overlap in both directions. RGB and HSV



(a) On VIPeR dataset (b) On i-LIDS MCTS dataset

Fig. 3. Comparative results with LMNN algorithm on VIPeR and i-LIDS MCTS datasets

histograms encode the different color distribution information in the RGB and HSV color space, respectively. LBP descriptors, encoding the texture feature, are extracted in grayscale images. All of the features are then put together to concatenated to a vector. To accelerate the learning process and reduce noise, we conducted principle component analysis (PCA) to obtain a low-dimension representation.

3.3 Baselines and Settings

To evaluate the effectiveness of the proposed LMNN-S, we compare with several representative person re-identification methods, including the Adaboost based feature selection method (Adaboost) [8], Rank support vector machine (RandSVM) [3], symmetry-driven accumulation of local feature (SDALF) [2] and probabilistic relative distance comparison (PRDC) [4], and metric learning methods, containing large margin nearest neighbor (LMNN) [9], information theoretical metric learning (ITML) [13] and KISSME [5]. As LMNN and ITML are not development for person re-identification, we use codes of these methods provided by their authors and report their results under the optimal parameter configurations.

The experiments are conducted as follows: for each dataset, we randomly selected p people as training set and the rest as testing set. For training set, positive samples consist a pair of images from the same person, which used to pull the the same labeled persons together closer, meanwhile, k images with different label are randomly selected for each positive samples to form a triple of images making up negative samples, which used to push the different labeled people further apart. Then, each testing set is divided into a probe and gallery set. The gallery set consists of one image for each person in testing set, and the remaining images are set as the probe set. Finally, each image in the probe set is matched with all images of the gallery set. The procedure was repeated 20

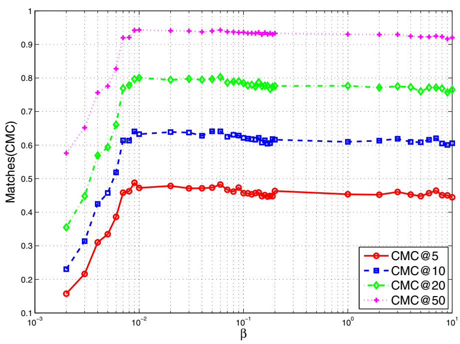


Fig. 4. Comparative results of different β

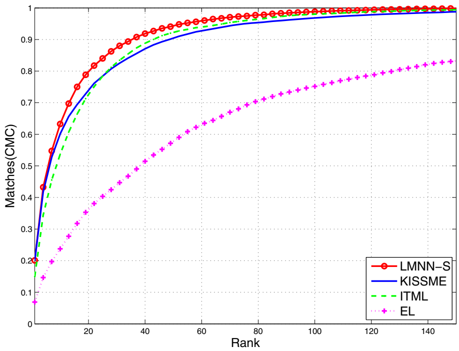


Fig. 5. Comparative results of metric learning

times, and the average of the Cumulative Matching Characteristic (CMC) curve, which is suggested in [7], was reported.

The CMC curve is exploited by most papers on the person re-identification problem [2, 3, 8, 5, 4] as well as. The value of CMC@k indicates the percentage of the real match ranked in the top k. In particular, let $P = \{p_1, \cdots, p_{|P|}\}$ be a probe set, where $|P|$ is the size of P, and $G = \{g_1, \cdots, g_n\}$ be a gallery set. For each probe image $p_i \in P$, all gallery images $g_j \in G$ are ranked based a defined distance function. The correct match is denoted as g_{p_i}, and the rank index of which is denoted as $r(g_{p_i})$. The CMC@k is defined as

$$CMC_k = \frac{\sum_{i=1}^{|P|} \mathbf{1}(r(g_{p_i}) \leq k)}{|P|} \tag{15}$$

where $\mathbf{1}(\cdot)$ is the indicator function.

3.4 Results and Analysis

Comparing to LMNN. We first compare LMNN-S with the LMNN method used in [11] on both VIPeR and i-LIDS MCTS datasets with settings in above subsection. As can be seen from Fig. 3, obviously, the LMNN-S is outperform LMNN on both datasets. The maximum improvement achieves 10% at CMC@5 on i-LIDS MCTS dataset. These results empirically validate the previous discussion in Sec. 1.

In addition, we evaluate the influence of the parameter of logistic loss β. We change the β from 0.001 to 10 and the results are shown in Fig. 4. The performance is going straight up when β change from 0.001 to 0.01, then reduce slightly and nearly stable when $\beta > 1$. It is consistent with the discussion in Sec. 1.

Comparing to the State-of-the-art Person Re-identification Methods. The comparison results with the state-of-the-art methods for person

Table 1. Comparative results with state-of-the-art person re-identification methods on top ranked matching rate(%)

Methods	VIPeR Dataset					iLIDS MCTS			
	r=1	5	10	20	50	1	5	10	20
SDALF [2]	**20**	40	50	65	85	28.5	48.0	57.5	68.0
Adaboost [8]	8.2	24.2	36.6	52.1	90.1	35.6	66.4	79.9	93.2
RankSVM [3]	16.3	38.2	53.7	69.9	85	43.0	71.3	85.2	97.0
PRDC [4]	15.7	38.4	53.9	70.1	87	**44.1**	72.7	84.7	96.3
KISSME [5]	**20.0**	46.0	60.1	74.2	90.5	-	-	-	-
ITML [13]	15.0	38.5	53.7	72.8	92.3	36.4	68.0	83.1	95.55
LMNN [9]	18.7	47.4	62.6	77.8	92.3	36.5	64.0	78.2	93.1
Ours	19.9	**49.3**	**64.3**	**79.8**	**94.1**	39.8	**72.9**	**87.3**	**97.1**

Table 2. Comparative results of computing time with varying numbers of negative samples on VIPeR dataset

Methods	1	2	3	4	5	6	7	8	9	10	LMNN
Time(s)	1.36	2.89	4.11	5.6	6.72	7.94	9.51	11.66	13.12	14.56	20.1
(STD)	(0.53)	(1.53)	(0.91)	(1.19)	(1.55)	(0.84)	(1.15)	(1.50)	(1.80)	(2.24)	(0.61)

re-identification are shown in Table 1. For a fair comparison, the results of Adaboost, RankSVM, SDALF, PRDC and KISSME are directly taken from their original papers. As shown in Table 1, our approach obtains competitive results across all ranks. Besides, the metric learning based methods, such as PRDC and KISSME, outperforms the non-metric learning methods, such as SDALF. As the discussion in Sec. 1, the metric learning seeks a optimal distance function with the supervise of the training samples. The learned distance function is more suitable for the problem than standard distance function, such as Euclidean distance. Moreover, we conducted several experiments with several popular metric learning methods with the same feature representation. The Fig. 5 gives a summery of comparison results. EL indicates the Euclidean distance without learning. Obviously, the metric learning significantly improve the performance and our approach outperform others metric learning methods.

Influence of the Number of Negative Samples. As discussion in Sec. 2.3, the number of negative samples is much larger than positive samples, and we use a stochastic sampling based algorithm. Therefore, we conducted experiments where the number of negative samples is set from one to twenty for each positive sample. As shown in Fig. 6, the effectiveness is about the same when the number

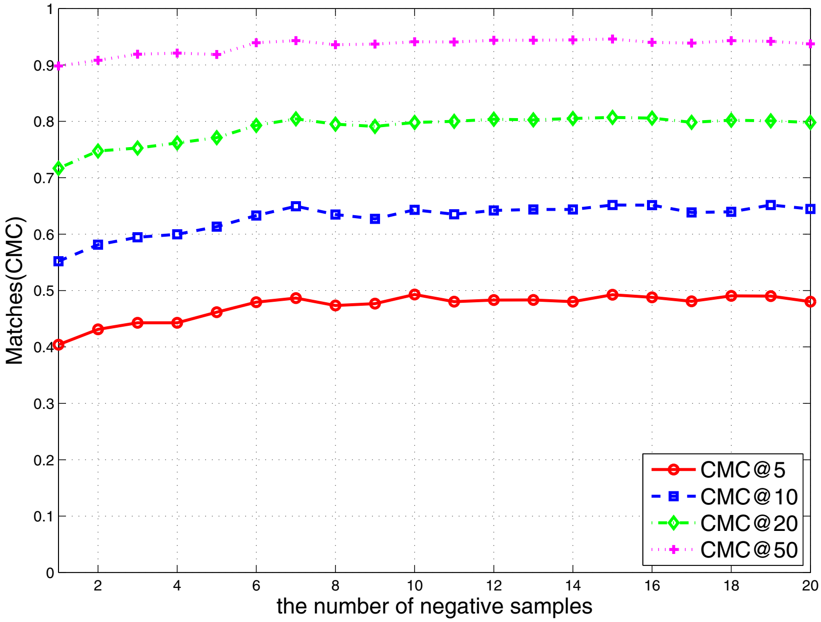


Fig. 6. Comparative results with different size of negative samples on VIPeR dataset

is larger than eight. It is very important for the real system, as the total size of training set can be reduced to $O(n)$ rather than $O(n^2)$.

Computational Cost. Finally, we conducted our experiments on Intel dual-core 2.59GHz CPU PC to evaluate the efficiency of the proposed method, LMNN-S, and compare it with LMNN as well as. The VIPeR dataset was used to evaluate, with which the training size is 316 and dimension of feature vector is 50. Then, we count the computation time of the provided algorithm with varying numbers of negative samples from 1 to 10. For each setting, we repeated 20 times, and the average time was reported in Table 2. As shown in Table 2, we observed that LMNN-S is more efficient than LMNN even though we did not employ the active set method which was designed to improve efficiency of LMNN [9]. Besides, as increasing negative samples, the computational cost descend significant. It is a good selection that improve our implementation with the active set technology at processing large number of training samples. Fortunately, it achieves an enough good effectiveness selecting relatively little negative samples.

4 Conclusions and Future Work

In this paper, we exploit logistic loss function to improve a popular metric learning method, LMNN, and hence transform the problem to a smooth convex optimization problem. Then, we provide a stochastic sampling based optimization algorithm to solve the problem motivated by SGD algorithm. Extensive comparative experimental results reported in Sec. 3 show that the proposed approach superior than the standard LMNN method in both accuracy and efficiency and outperforms the state-of-the-art person re-identification methods and several popular metric learning methods on two challenging public datasets, VIPeR and i-LIDS MCTS. In the future, we plan to further investigate the metric learning methods with the environment in which the instance of each person consist of a sequence of images and metric learning methods can not be directly used.

Acknowledgement. The research was supported by the major national science and technology special projects(2010ZX03004-003-03, 2010ZX03004-001-03), the National Basic Research Program of China (973 Program) (2009CB320906), the National Natural Science Foundation of China (60970160, 61070080, 61003184, 61172173, 61170023, 61231015), the China Postdoctoral Science Foundation funded project (2013M530350), the President Fund of UCAS, the Open Project Program of the National Laboratory of Pattern Recognition (NLPR).

References

1. Hirzer, M., Beleznai, C., Kstinger, M., Roth, P.M., Bischof, H.: Dense appearance modeling and efficient learning of camera transitions for person re-identification. In: ICIP (2012)
2. 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)
3. Prosser, B., Zheng, W.S., Gong, S., Xiang, T.: Person re-identification by support vector ranking. In: British Machine Vision Conference (BMVC) (2010)
4. Zheng, W.S., Gong, S., Xiang, T.: Person re-identification by probabilistic relative distance comparison. In: Computer Vision and Pattern Recognition (CVPR) (2011)
5. Kostinger, M., Hirzer, M., Wohlhart, P., Roth, P., Bischof, H.: Large scale metric learning from equivalence constraints. In: Computer Vision and Pattern Recognition (CVPR) (2012)
6. Gray, D., Brennan, S., Tao, H.: Evaluating appearance models for recognition, reacquisition, and tracking. In: IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS) (2007)
7. Wang, X., Doretto, G., Sebastian, T., Rittscher, J., Tu, P.: Shape and appearance context modeling. In: International Conference on Computer Vision (ICCV) (2007)
8. Gray, D., Tao, H.: Viewpoint invariant pedestrian recognition with an ensemble of localized features. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 262–275. Springer, Heidelberg (2008)
9. Weinberger, K.Q., Blitzer, J., Saul, L.K.: Distance metric learning for large margin nearest neighbor classification. Journal of Machine Learning Research 10, 207–244 (2009)
10. Zhang, T.: Solving large scale linear prediction problems using stochastic gradient descent algorithms. In: Proceedings of the Twenty-First International Conference on Machine Learning (ICML) (2004)

11. Dikmen, M., Akbas, E., Huang, T.S., Ahuja, N.: Pedestrian recognition with a learned metric. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part IV. LNCS, vol. 6495, pp. 501–512. Springer, Heidelberg (2011)
12. Tao, D., Jin, L., Wang, Y., Yuan, Y., Li, X.: Person re-identification by regularized smoothing kiss metric learning. IEEE Trans. on Circuits and Systems for Video Technology (2013)
13. Davis, J.V., Kulis, B., Jain, P., Sra, S., Dhillon, I.S.: Information-theoretic metric learning. In: International Conference on Machine learning (ICML) (2007)