Learning Cross-View Binary Identities for Fast Person Re-Identification

Feng Zheng[1], **Ling Shao**[2]
[1]Department of Electronic and Electrical Engineering, The University of Sheffield.
[2]Department of Computer Science and Digital Technologies, Northumbria University.
cip12fz@sheffield.ac.uk, ling.shao@ieee.org

Abstract

In this paper, we propose to learn cross-view binary identities (CBI) for fast person re-identification. To achieve this, two sets of discriminative hash functions for two different views are learned by simultaneously minimising their distance in the Hamming space, and maximising the cross-covariance and margin. Thus, similar binary codes can be found for images of a same person captured at different views by embedding the images into the Hamming space. Therefore, person re-identification can be solved by efficiently computing and ranking the Hamming distances between the images. Extensive experiments are conducted on two public datasets and CBI produces comparable results as state-of-the-art re-identification approaches but is at least 2200 times faster.

1 Introduction

Person re-identification addresses the problem of associating people, at different locations and times, observed by the non-overlapping Closed-Circuit TeleVision system. It has various potential applications, such as long-term multi-person tracking, person re-acquisition and forensic search [Gong *et al.*, 2013]. Due to the various difficulties including illumination changes, viewpoint and pose variations, inter-object occlusions and low resolution images, person re-identification is still a very challenging task and far from being tackled.

Most of the existing approaches can be categorised into two groups: learning features which are invariant to view changes [Gray and Tao, 2008; Li *et al.*, 2014] and learning the metric functions which are used to rank the pairs of observations from different views [Prosser *et al.*, 2010; Zheng *et al.*, 2012; Pedagadi *et al.*, 2013]. However, in spite of their good performance on public datasets, existing methods generally neglect considering the efficiency of the algorithm in the matching stage. In fact, the searching speed of a re-identification algorithm plays a significant role in real-world applications.

In general, the efficiency of matching mainly depends on two aspects: (1) the number of samples stored in the gallery set; (2) the definition of similarity. As for the first aspect, it is impossible to reduce the number of samples. It is because [Gong *et al.*, 2013]: (1) A large number of surveillance cameras have been installed in public spaces assembled with hundreds of thousands of persons, even in a day. (2) Re-identification in open environments can potentially scale to arbitrary levels, covering huge spatial areas spanning not just different buildings but also different cities, or countries, leading to an overwhelming quantity of "big data". (3) Person re-identification can be extended from multi-camera networks to distributed Internet spaces. Therefore, with an explosive growth of images, speeding up the matching stage of a re-identification system by designing a more advantageous similarity criterion is an essential and non-replaceable option.

Figure 1: Assuming that the left and middle images are of one person and the middle and right images are of different persons but captured by one camera. Our aim is to learn two sets of hash functions (one for each view) which embed the images to binary codes (IDs) so that the IDs (second row) of a same person are similar with each other and the IDs of different persons are quite dissimilar. As illustrated in this figure, the learned binary codes play a same role as fingerprints.

In this paper, a novel approach, learning Cross-view Binary Identities (CBI), is proposed to reduce the computational burden for person re-identification. In fact, hashing has been widely used for nearest neighbour search in computer vision areas, such as image retrieval, object recognition and image matching, but it has been seldom used in re-identification. Using the hash functions, various special properties can be preserved in the learned codes, such as locality, variance and affinity. For two observations x_a and x_b of one person in two different views, CBI can learn two similar codes, which are considered as the identity (ID) of that person, as shown in Fig. 1. The learned binary codes enable efficient similarity search in different views using the Hamming distance between codes. Moreover, compact binary codes are extremely economical for large-scale data storage.

Our contributions are three-fold: (1) By learning the binary codes, each person has a similar identity across different views. Due to the efficiency of binary codes, person re-identification in a huge dataset can be realised. (2) In CBI, variances of learned bits, cross-covariance and margin of learned hash codes are simultaneously maximised and an efficient iterative optimisation solution is introduced. (3) Moreover, in CBI, a theoretical proof is given to guarantee the transfer from Hamming space to Euclidean space. Unlike most methods, which directly relax the sign function, such as [Rastegari *et al.*, 2013], we consider the theoretical reason behind when it is safe to relax the sign function.

2 Related Works

To address the challenge of person re-identification, many efforts have been made along the two directions: learning discriminative features and learning the metric functions. Moreover, both aspects are considered to further improve the performance in [Liao *et al.*, 2015].

On the one hand, the learned features are generally invariant to the view changes and simple metrics are used for matching. Various methods including local patches [Gray and Tao, 2008], colour distributions over colour names (SCNCD) [Yang *et al.*, 2014], and salience learning [Zhao *et al.*, 2013b] are proposed to learn discriminative features for person re-identification. In [Li *et al.*, 2014], deep learning is exploited to automatically learn features for the re-identification task and the deep framework has been improved in [Ahmed *et al.*, 2015] by incorporating neighbouring locations of other images.

On the other hand, complex distance metrics are learned to rank the pairs of observations from different views. Some metric learning related algorithms including Support Vector Ranking (PRSVM) [Prosser *et al.*, 2010], relative distance comparison (PRDC) [Zheng *et al.*, 2012], equivalence constraints [Kostinger *et al.*, 2012] and dimensionality reduction [Pedagadi *et al.*, 2013] to learn effective distance for person re-identification. Very recently, ensemble metrics, such as a mixture of similarities [Chen *et al.*, 2015] and an ensemble of distances [Paisitkriangkrai *et al.*, 2015], are exploited to discover multiple matching patterns.

Despite the promising performance achieved by the existing methods, all of them suffer from a huge computational burden in the test stage. Due to the efficient nearest neighbour search using binary codes, hashing techniques have been widely adopted in many vision applications, especially in indexing large-scale data. Composite Hashing with Multiple Information Sources (CHMIS) [Zhang *et al.*, 2011] and the Cross View Hashing (CVH) [Kumar and Udupa, 2011] extend the SH [Weiss *et al.*, 2008] from different aspects, respectively. The boosting algorithms are adopted to embed the input data from two arbitrary spaces into a same Hamming space by Cross-Modality Similarity Sensitive Hashing (CMSSH) [Bronstein and Bronstein, 2010]. Considering the maximum margin, Predictable Dual-view Hashing (PDH) [Rastegari *et al.*, 2013] explores a joint formulation for learning binary codes of data from two different views. Collective Matrix Factorisation Hashing (CMFH) [Ding *et al.*, 2014] is based on the assumption that the interlinked data should have the same latent factors and the hash codes can be learned from these factors. Moreover, local functions [Zhai *et al.*, 2013] and correlation-maximal mappings [Long *et al.*, 2015] are exploited to learn the common binary codes.

In this paper, we accomplish person re-identification by learning a set of hash functions for each view. From the feature learning perspective, CBI learns a discriminative binary representation for each person. Furthermore, from the metric learning perspective, a more efficient distance metric in the Hamming space is learned for matching. Moreover, most of the above hashing methods are exploited to preserve the local properties in intra-module and inter-module. However, since normally very few images for each person exist in one view, the extracted features do not meet the local smoothness assumption. Instead, CBI focuses on learning similar binary codes for a person under different views but the Hamming distances between different persons will be maximised.

3 Learning Cross-view Binary Identities

For two different camera views: a and b, we can collect two training datasets $X_a = \{x_a^1, x_a^2, \cdots, x_a^n\}$ and $X_b = \{x_b^1, x_b^2, \cdots, x_b^n\}$, where x_a^i is a column vector observed by view a for person i and n is the number of paired samples (x_a^i, x_b^i). Our aim is to find K hash functions $F = \{f_v^1, \cdots, f_v^K\}$ for each view $v \in \{a, b\}$ and $y_v(k) = f_v^k(x_v)$. In this paper, the hash functions are constructed by a set of linear hyperplanes: $W_v = \{w_v^1, w_v^2, \cdots, w_v^K\}$. Thus, for dataset X_v, we obtain $Y_v = \{y_v^1, y_v^2, \cdots, y_v^n\}$ by using $y_v^{ik} = sign((w_v^k)^T x_v^i)$. It is obvious that $y_v^i \in \{-1, 1\}^K$. For simplicity, we can write it as: $Y_v = sign(W_v^T X_v)$.

For a person with an image in the probe view, the first step is to calculate the ID by using the learned projections. Next, the ID can be used to retrieve the images of persons with similar IDs in the gallery view. The IDs of persons in the gallery view can be obtained in advance. Finally, the re-identification can be achieved by ranking the Hamming distances. Because the learned pairs of projections can embed the images of a same person into a same ID, the top list of ranking will conclude the ones corresponding to the probe image.

3.1 Maximising the variance of bits

We want to produce an efficient code for each view v, in which the variance of each bit is maximised and the bits are pairwise uncorrelated [Gong and Lazebnik, 2011]. Thus, we achieve this by maximising the following objective function:

$$\begin{array}{rl} & \mathcal{I}_v = \sum_k var(f_v^k(x_v)) \\ s.t. & cor(f_v^{k_1}(x_v), f_v^{k_2}(x_v)) = 0, \\ & cor(f_v^k(x_v), f_v^k(x_v)) = 1, \end{array} \tag{1}$$

where $k_1 \neq k_2$. However, the requirement of exact balancedness makes the above objective function intractable. By signed magnitude relaxation, we get the following continuous objective function based on dataset X_v: $\mathcal{I}_v = \sum_k E(||(w_v^k)^T x_v||_2^2) \approx \frac{1}{n} \sum_k (w_v^k)^T X_v X_v^T w_v^k = \frac{1}{n} tr(W_v^T X_v X_v^T W_v), s.t.\, W_v^T W_v = I$.

We relax the constraints as: $((w_v^{k_1})^T w_v^{k_2})^2 < \delta_v, k_1 \neq k_2$, without considering the norm of each linear projection. δ_v is a minimal positive value. In fact, in the following, we can see that it is not necessary to require the unit norm constraints if the functions satisfy the hinge loss constraint.

3.2 Minimising the Hamming distance

In a single-view problem, the main consideration is that the learned codes are discriminative to represent all the samples by preserving some special properties. However, it is not enough in a multi-view problem, such as person reidentification. Our main goal, in this paper, is to learn K hash functions for each view so that two observations of each person have the most similar binary codes (IDs). That is to say, the Hamming distance between two sets of codes of one person should be minimised. For a pair of sample sets (X_a, X_b) collected under the two views a and b, the Hamming distance between them is defined as:

$$\mathcal{L}_h(X_a, X_b) = \sum_i D_h(y_a^i, y_b^i), \tag{2}$$

where D_h indicates the Hamming distance. D_h is equal to the number of ones in $y_a^i \oplus y_b^i$, where $\oplus$ is a logical operation that outputs true whenever the inputs differ.

However, despite its efficiency, minimisation of the Hamming distance is generally intractable, because it is nondifferentiable to the linear functions. Thus, we seek to minimise an alternative item, which guarantees the Hamming distance will be minimised simultaneously. Fortunately, Proposition 1 shows that we can achieve this, when the linear hash functions satisfy the hinge loss constraint defined as follows.

Definition 1: Hinge loss constraint. *For any sample x_v^i in one view v, if the linear function w_v^k is satisfying*

$$y_v^{ik}(w_v^k)^T x_v^i \geq 1 - \xi_v^{ki}, \tag{3}$$

where ξ_v^{ki} is a minimal non-negative value, thus w_v^k is the hinge loss constraint satisfied function.

The hinge loss function is used for "maximum-margin" classification, most notably for Support Vector Machines (SVM) [Cortes and Vapnik, 1995]. It penalises the items satisfying $y_v^{ik}(w_v^k)^T x_v^i < 1$ so that all items can be correctly classified and the classification score should keep stable as well. In our framework, we hope all the samples can be projected outside of $[-1, 1]$ by each linear function so that the learned codes are relatively stable for all the samples. Moreover, if W_a^k and W_b^k are hinge loss constraint satisfied functions, the Hamming distance between the learned codes are constrained by the Euclidean distance.

Proposition 1: *If two sets of linear projections W_a and W_b for two views are the hinge loss constraint satisfied functions and their corresponding binary codes are defined by $y_v^i = sign(W_v^T x_v^i), v \in \{a, b\}$, thus the inequality can be established when satisfying $\forall k, \xi_a^{ki} + \xi_b^{ki} \leq 1$:*

$$D_h(y_a^i, y_b^i) < ||W_a^T x_a^i - W_b^T x_b^i||_2^2. \tag{4}$$

3.3 Overall objective function

To construct our objective function, we have to consider: (1) The cumulative Hamming distance should be minimised while the variance of bits should be maximised, thus

$$\begin{aligned}\mathcal{L}(W_a, W_b) &= \textstyle\sum_i ||W_a^T x_a^i - W_b^T x_b^i||_2^2 - \sum_v \mathcal{I}_v \\ &= -2tr(W_a^T S_{ab} W_b),\end{aligned} \tag{5}$$

where $S_{v_1 v_2} = X_{v_1} X_{v_2}^T, v_1, v_2 \in \{a, b\}$. (2) For conditions $\xi_a^{ki} + \xi_b^{ki} <= 1$, we can sum all of them over samples and functions to obtain the relaxed inequality $\Upsilon = \sum_{ki} \xi_a^{ki} + \sum_{ki} \xi_b^{ki} <= K * n$. (3) To increase the generalisation of the model, it is necessary to penalise each learned projection by maximising the margin of two separated samples $||W||^2 = \frac{1}{2} \sum_{v \in \{a,b\}} \sum_k ||w_v^k||^2$, which is same as an SVM classification model. Therefore, we obtain

$$\mathcal{L} = \lambda_2 \mathcal{L}(W_a, W_b) + C\Upsilon + ||W||^2, \tag{6}$$

where λ_2 and C are used to balance the losses. In Eqn 6, the quantities $\mathcal{L}(W_a, W_b)$, Υ and $||W||^2$ can be considered as a cross-view loss function for matching, a within-view quantisation loss for hashing and a regularisation, respectively.

Proposition 2: *Substituting $\mathcal{L}(W_a, W_b)$, Υ and $||W||^2$ into (6) with considering the conditions, we have:*

$$\begin{aligned}\{W_a^*, W_b^*\} = \arg\min\nolimits_{W_a, W_b} & -\lambda_2 tr(W_a^T S_{ab} W_b) \\ & + \textstyle\sum_k (\frac{1}{2} \sum_{v \in \{a,b\}} ||w_v^k||^2 + C \sum \xi_v^{ki}) \\ s.t. \qquad & \forall v \in \{a, b\}, i, k, k_1 \neq k_2 : \\ & ((w_v^{k_1})^T w_v^{k_2}))^2 \leq \delta_v, \\ & (y_v^{ik}(w_v^k)^T x_v^i) \geq 1 - \xi_v^{ki}, \xi_v^{ki} \geqslant 0.\end{aligned} \tag{7}$$

Firstly, we can see that the proposed CBI is related to Canonical Correlation Analysis (CCA) [Hotelling, 1936], but without minimising the covariance of intra-module. A solution of CCA may be affected by highly correlated but unimportant (in the sense of low variation and/or covariation) variables. However, a preserved large variance will increase the stability and discriminativeness of the learned codes. Secondly, we can see that S_{ab} is the cross-covariance matrix between the two views a and b. Maximum Cross-variance Analysis (MCA) [Lampert and Kromer, 2010] is a typical dimensionality reduction method for two cross sets of highly correlated variables in the low dimensional space. The proposed CBI can also learn the compact, highly correlated binary codes by maximising the cross-covariance in the new space. Finally, although PDH [Rastegari *et al.*, 2013] also learns the projection by maximising the margins, there are two significant differences between CBI and PDH. On the one hand, both the cross-variance and the variances of bits have been maximised in CBI but neither of them is considered in PDH. On the other hand, PDH obtains the projection by directly using the classical SVM, but, in CBI, a novel dual problem with a first degree item is solved to learn the projections. That is why PDH cannot improve the performance by increasing the number of bits.

4 Optimisation

Despite the complex formula in Proposition 2, in general, the problem can be solved by gradient descend with iterative projection. However, we adopt a more efficient way to search the local optimal solution, considering that the objective is convex to each variable with other variables fixed. Following [Lee *et al.*, 2006], we can iteratively optimise the projections one by one. The training procedure of CBI is summarised in Algorithm 1.

Algorithm 1 CBI training

```
Input: Training dataset $X_a$, $X_b$ and parameters$\lambda_1$, $\lambda_2$, $C$ and $K$.
Output: $W_a$ and $W_b$.
Initialisation
(1) Initiate $W_a$ and $W_b$ by a random generator.
Repeat $t = 1, \cdots$
(2) Choose the $\mathbf{k}$th pair of projections using Eqn. 10.
(3) Decide the optimisation order of $v_1$ and $v_2$.
    (a) Calculate $\Theta_{v_1}^{\mathbf{k}}$ and $s_{v_1 v_2}^{v_2 \mathbf{k}}$.
    (b) Solve the quadratic programming problem in Eq. 8.
    (c) Calculate the projection for view $v_1$ using Eq. 9.
    (d) Update the codes of view $v_1$ by $y_{v_1}^{ik} = sign((w_{v_1}^k)^T x_{v_1}^i)$.
(4) Assign the codes for view $v_2$ by $y_{v_2}^{ik} = y_{v_1}^{ik}$.
    (a) Calculate $\Theta_{v_2}^{\mathbf{k}}$ and $s_{v_2 v_1}^{v_1 \mathbf{k}}$.
    (b) Solve the quadratic programming problem in Eq. 8.
    (c) Calculate the projection for view $v_2$ using Eq. 9.
    (d) Update the codes of view $v_2$ by $y_{v_2}^{ik} = sign((w_{v_2}^k)^T x_{v_2}^i)$.
 If satisfy conditions: Exit.
Return Update the $\mathbf{k}$th binary codes and hash functions.
```

4.1 Sequential optimisation

For further simplifying the optimisation, the orthogonal constraint of projections in intra-module has been added into the objective function. Thus, as shown in Proposition 3, we can see that the problem is the same as the classical SVM but only by adding an item of first degree.

Proposition 3: *We fix all other variables except for w_a^k and ξ_a^{ki} $(i = 1, \cdots, n)$. By removing the irrelevant items, we obtain:*

$$\begin{aligned} w_a^k = \arg\min \tfrac{1}{2}(w_a^k)^T \Theta_a^k w_a^k - (w_a^k)^T s_{ab}^{bk} + C \textstyle\sum_i \xi_a^{ki}, \\ s.t. \forall i, k \;\; y_a^{ik} (w_a^k)^T x_a^i \geq 1 - \xi_a^{ki}, \xi_a^{ki} \geqslant 0, \end{aligned} \tag{8}$$

where $\Theta_a^k = \lambda_1 \sum_{j \neq k} w_a^j (w_a^j)^T + I$ and $s_{ab}^{bk} = \lambda_2 S_{ab} w_b^k$.

So far, all variables related to view b have been absorbed into the vector s_{ab}^{bk}. The objective function becomes a classical convex quadratic programming problem. Same as SVM, a dual problem is designed to obtain the optimal solution:

$$(w_a^k)^* = (\Theta_a^k)^{-1} (s_{ab}^{bk} + X_a^{yk} \alpha_a^k), \tag{9}$$

where α_a^k is the optimal solution of the dual problem and $X_a^{yk} = (y_a^{1k} x_a^1, \cdots, y_a^{nk} x_a^n)$.

Eqs. 9 and 8 are similar to the equations in the classical linear SVM. However, it is meaningful to point out the two differences between them, which constitute the advantages of CBI and distinguish from PDH. On the one hand, the inverse of Q in the quadratic item forces that the learned projection must be orthogonal to the other projections within the same view. On the other hand, the s in the first degree item forces that the learned projection should be highly related to the corresponding projection within another view. Moreover, because CBI optimises the projections on each view separately, it can be easily extended to the $n_v (n_v > 2)$ situation by directly computing $s = \sum_{b \neq a, b=1}^{n_v} \lambda_2 S_{ab} w_b^k$. Furthermore, CBI can also be generalised to multiple-shot cases by minimising the hamming distances between all the image pairs of one person for any two different views.

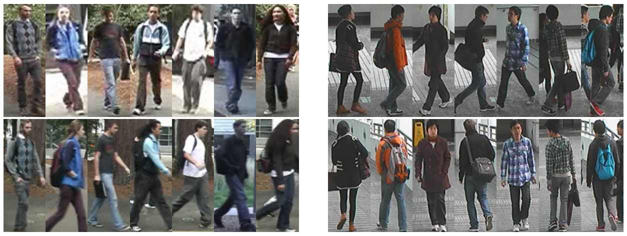


Figure 2: Image samples: VIPeR (left) and CUHK01 (right).

4.2 Greedy selection

Then, the problem becomes how to choose a projection which will be optimised at present. Once one projection has been selected, the new optimal projection will be obtained by solving the problem in Proposition 3. Assume the loss of each projection w_a^k, at the present iteration, is defined as: $\mathcal{L}(w_a^k) = \frac{1}{2}(w_a^k)^T \Theta_a^k w_a^k - (w_a^k)^T s_{ab}^{bk} + C \sum_i \lfloor 1 - y_a^{ik} (w_a^k)^T x_a^i \rfloor_+$, where $\lfloor \rfloor_+$ is the hinge loss function. Therefore, greedy selection will be achieved by:

$$\mathbf{k} = max_k (\mathcal{L}(w_a^k) + \mathcal{L}(w_b^k)). \tag{10}$$

We hope the overall loss will be decreased by minimising the items which have a high loss. The next step is to optimise the selected $\mathbf{k}$th pair of projections, which are detailed as follows.

First, view v_1 with less loss will be optimised in advance, because the learned binary codes probably approach the optimal ones. The binary codes $y_{v_1}^{ik}$ of the last round will be considered as the initials to optimise the problem in Proposition 3 for view v_1. Next, the binary codes will be updated according to $y_{v_1}^{ik} = sign((w_{v_1} v^k)^T x_{v_1}^i)$ by using the learned projection. After that, the learned codes of view v_1 will be used to optimise the projection in view v_2. This means y_{v_2} is initiated by y_{v_1}. This process is the same as in [Rastegari *et al.*, 2013]. Finally, the same optimisation of Proposition 3 will be conducted for view v_2. Thus, the binary codes of view v_2 will be also updated by $y_{v_2}^{ik} = sign((w_{v_2} v^k)^T x_{v_2}^i)$.

The optimisation procedure can be terminated by different criteria, such as difference between two binary codes of two views less than a small positive number or the fixed number of iterations. In our experiments, we observed that when the number of iterations is around the number of projections K, the difference between two binary codes will be the least.

The proofs of Proposition 1 and 3 and Eq. 9 will be given in the Supplemental Materials.

4.3 Convergence

In this section, theoretical analysis is provided by rigorous proof of the convergence of the objective function in Proposition 2.

Proposition 4: *$\mathcal{L}$ in Proposition 2 monotonically decreases with each optimization step for w_a^k and ξ_a^{ki}, and therefore $\mathcal{L}$ converges to a local optimum.*

Proof: Denote $J(w_a^k, \xi_a^{ki} | i = 1, \cdots, n)$ as the objective function in Proposition 3 and R as the remaining part which is unrelated to w_a^k and ξ_a^{ki} in Proposition 2, respectively. Then, we obtain the objective function in Proposition

Methods	CBI-100	CBI-500	CBI-700	SDALF	KISSME
Time(s)	5.9e-07	1.1e-06	1.4e-06	3.6e+00	9.2e-03
Methods	PRDC	eSDC	PRSVM	Mrank	SCNCD
Time(s)	9.3e-03	1.14e+01	3.2e-03	3.4e-02	4.2e-03

Table 1: Time comparison of computing the similarities between one probe sample and all the gallery samples (316) using the compared methods. CBI-100 denotes that only 100 hash codes have been learned.

2: $\mathcal{L} = J(w_a^k, \xi_a^{ki} | i = 1, \cdots, n) + R$. At the tth step of optimisation, suppose that w_a^k has been chosen (Otherwise, the same conclusion can be also obtained for w_b^k.). Then, we can denote $\mathcal{L}^{t-1}$ as the objective function before optimising w_a^k and $\mathcal{L}^t$ is the function after we obtain the optimum $(w_a^k)^*$ of $J(w_a^k, \xi_a^{ki} | i = 1, \cdots, n)$. Since $J(w_a^k, \xi_a^{ki} | i = 1, \cdots, n)$ is a convex problem, then $J(w_a^k, \xi_a^{ki} | i = 1, \cdots, n) \geq J((w_a^k)^*, \xi_a^{ki} | i = 1, \cdots, n)$. Moreover, because R is fixed, the following inequality can be established:

$$\cdots \geq \mathcal{L}^{t-1} \geq \mathcal{L}^t \geq \cdots . \tag{11}$$

5 Experiments

We test our proposed CBI[1] for person re-identification on two public datasets: VIPeR [Gray and Tao, 2008] and CUHK01 [Li *et al.*, 2014]. Some example images of the two datasets are shown in Fig. 2. To illustrate the performance and efficiency of CBI, 17 recent algorithms, including 13 person re-identification methods and 4 multi-modal hash function learning methods, are used for comparison.

Image representation: In this paper, to reflect the advantage of our CBI to learn binary codes for different descriptors, three representations, including ELF [Gray and Tao, 2008], SCNCD [Yang *et al.*, 2014] and LOMO [Liao *et al.*, 2015], are adopted as the basic descriptors. CBI is not sensitive to the parameters for the two datasets and we set $\lambda_1 = 2$ and $C = 200$ for all the experiments. However, λ_2 will be set to 0.05, 10 and 5 for ELF, SCNCD and LOMO, respectively.

Evaluation protocol: We randomly partition a dataset into two parts without overlap on person identities, according to a certain percentage. The expectation is reported by conducting 10 trials of evaluation. The parameters of other hashing algorithms are carefully tuned so that the best results are obtained. The results of other person re-identification methods either come from original papers or by running their offered codes, with exactly the same experimental setting. Same as most person re-identification publications, the standard Cumulated Matching Characteristics (CMC) [Wang *et al.*, 2007] curves and the corresponding Area Under Curve (AUC) are used to illustrate the performance of different methods.

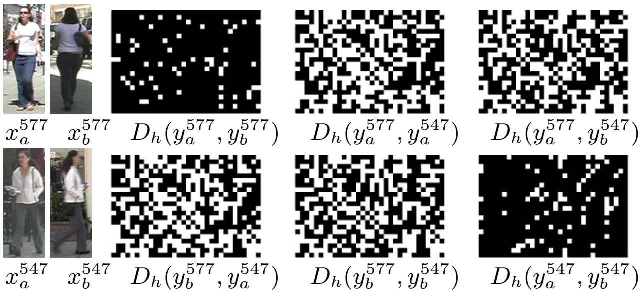

x_a^{577}
x_b^{577}
$D_h(y_a^{577}, y_b^{577})$
$D_h(y_a^{577}, y_a^{547})$
$D_h(y_a^{577}, y_b^{547})$
x_a^{547}
x_b^{547}
$D_h(y_b^{577}, y_a^{547})$
$D_h(y_b^{577}, y_b^{547})$
$D_h(y_a^{547}, y_b^{547})$

Figure 3: Left most: the images in the two views of the 577th (first row) and 547th (second row) persons in the VIPeR dataset. QR code: the Hamming distances between the learned codes for the four samples and the exact Hamming distances are 57, 331, 317, 328, 316 and 78, respectively.

5.1 The efficiency of CBI

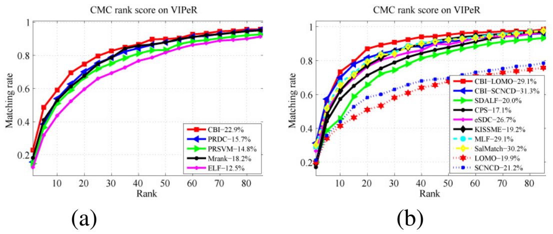
The Hamming distance comparison of the learned binary codes for two different persons in two views from the test set on the VIPeR dataset is shown in Fig. 3. According to the proposed CBI, binary codes with length 704 for each image are learned and resampled into an image with 22 × 32 pixels so that it is easy to illustrate the difference of learned codes. From this figure, we can see that the Hamming distance between two images of a same person in two views is much lower than that between the images of different persons no matter they are captured in the same view or not.

CBI is efficient for similarity search in the testing stage, since the bit XOR operation is applied when calculating the Hamming distance. To illustrate the efficiency of CBI, we compare the time of similarity computation for various methods on the VIPeR dataset. To simulate a real situation, the time includes the feature projection for the probe image but the embedded features of gallery images are obtained in advance. All algorithms are run on a Matlab 7 platform installed on Windows 7 with Intel Core $3.4GHz$ CPU and $8G$ memory. The codes of compared methods are provided by their original authors and comparison results are shown in Table 1. We can see that the proposed CBI is at least 2200 times faster than other methods. It is worth to point out that the local patches based methods, including eSDC [Zhao *et al.*, 2013b], MLF [Zhao *et al.*, 2014] and SalMatch [Zhao *et al.*, 2013a], achieve advantageous performance (Rank 1: eSDC-26.74%). However, the methods exploiting the local patches introduces a huge computational burden and they are 10^7 times slower than CBI. In general, in a real-world application, the number of samples in the gallery set n is very huge and the original dimension n_d is much larger than the number of learned bits K. In theory, the efficiency of CBI is at least $n(n_d + 1)/K$ times faster than other metric learning based methods.

5.2 Comparison with the state-of-the-art methods

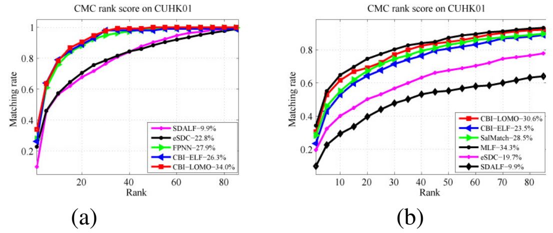
For evaluating on the VIPeR, we compare CBI with recent published algorithms, including: ELF [Gray and Tao, 2008], PRDC [Zheng *et al.*, 2012], PRSVM [Prosser *et al.*, 2010], SDALF [Farenzena *et al.*, 2010], CPS [Cheng *et al.*, 2011], Mrank [Loy *et al.*, 2013], eSDC [Zhao *et al.*, 2013b], SalMatch [Zhao *et al.*, 2013a], MLF [Zhao *et al.*, 2014], KISSME [Kostinger *et al.*, 2012], SCNCD [Yang *et al.*, 2014] and LOMO [Liao *et al.*, 2015]. The comparison results are shown in Fig. 4 (a) and (b). Among them, PRDC, Mrank and PRSVM used the ELF feature. We can see that the proposed CBI achieves much better results than the three methods al-

[1] The codes are released on a website: https://sites.google.com/site/crossmodalhashing/re-identification



(a) (b)

Figure 4: The CMC rankings on the VIPeR dataset. Numbers in legend are the Rank-1 accuracies. (a) All methods adopted the ELF feature; (b) Comparison with other methods.



(a) (b)

Figure 5: The CMC rankings on the CUHK01 dataset. (a) 100 test persons; (b) 486 test persons.

most in all ranking. Following [Liao *et al.*, 2015], a Cosine similarity measure is applied to SCNCD and LOMO. To compare with the three types of original features, we can see that the performance is boosted by CBI for at least 30%. Moreover, using the SCNCD feature, CBI is the best method at rank 1 and is better at low ranks ($\leq$ 30) than other state-of-the-art methods. Finally, by using LOMO feature, CBI has a 29.1% accuracy at rank 1 and outperforms other methods almost at all ranks.

For comparing on the CUHK01, we follow two partitions as in [Li *et al.*, 2014; Zhao *et al.*, 2014] and the results are shown in Fig. 5. For the first partition with 100 test persons, three methods including FPNN [Li *et al.*, 2014], eSDC [Zhao *et al.*, 2013b] and SDALF [Farenzena *et al.*, 2010] are compared with. We can see that CBI can achieve much better results than eSDC and SDALF at all ranks, no matter what features are used. To compare with the deep architecture based method FPNN, using the LOMO feature, CBI can achieve better results at all ranks while, using the SCNCD feature, is only slightly inferior FPNN at rank 1 (1.6 %) but better than FPNN at all other ranks. For the second partition with 486 test persons, the task is relatively more difficult and four state-of-the-art methods including eSDC [Zhao *et al.*, 2013b], SDALF [Farenzena *et al.*, 2010], SalMatch [Zhao *et al.*, 2013a] and MLF [Zhao *et al.*, 2014] are compared against. In this setting, MLF is the best method but, using the LOMO feature, CBI achieves very similar performance to MLF. In fact, MLF is a local patches based method thus the computational burden of feature calculating and matching is very high.

In total, CBI can achieve competitive performance with the state-of-the-art methods. We have to point out that recent works on improved deep learning [Ahmed *et al.*, 2015] and fusion based methods (LOMO+XQDA [Liao *et al.*, 2015], MLF+LADF [Zhao *et al.*, 2014], mixture of similarities [Chen *et al.*, 2015] and ensemble of distances [Paisitkriangkrai *et al.*, 2015]) reported higher results. However, since this paper mainly focuses on the efficiency, the combination of different methods is not considered, because they are computationally very expensive. Naturally, the performance of binary coding, a.k.a. hashing, methods will be lower than their corresponding non-hashing based methods due to the quantization loss [Gong and Lazebnik, 2011]. In the following section, the comparison of CCA [Hotelling, 1936] and CVH [Kumar and Udupa, 2011], which is a hashing version of CCA, will also prove this point. Therefore, we can conclude that, as a binary coding method for person re-identification, the performance of CBI is acceptable.

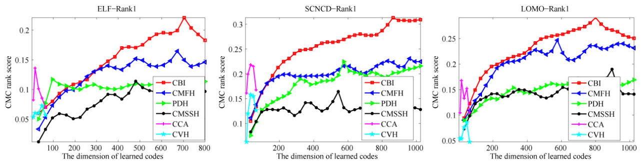


Figure 6: Rank 1 comparison with four hashing methods and CCA on the VIPeR dataset with the three features.

5.3 Comparison with other hashing methods

We compare our CBI with CCA [Hotelling, 1936] and multimodal binary code learning methods, including PDH [Rastegari *et al.*, 2013], CVH [Kumar and Udupa, 2011], CMSSH [Bronstein and Bronstein, 2010] and CMFH [Ding *et al.*, 2014] on the VIPeR and CUHK01 datasets.

The comparison results at rank 1 are shown in Fig. 6. Firstly, due to the finite rank of variance matrix, the dimensions of the features learned by CCA and CVH are constrained, thus their best performance is poor. Secondly, at very low dimensions, most methods achieve similar results and the performance of CCA is better than others. Thirdly, the ranking scores of other three methods including CMFH, PDH and CMSSH do not progressively increase by the increase of the length of codes. This is because the later learned codes tend to add little discriminative information, due to ignoring the orthogonal constraint between different hash functions. Fourthly, our proposed method achieves much better results than other methods when the code length is over 400. Finally, Tables 2 and 3 show the best performance of each method with the optimal length of the learned codes. The overall AUC at ranks from 1 to 100 and the rank 1 accuracies have been reported. From the tables, we can see that CCA achieves much better results than the corresponding hashing version CVH. Moreover, we can observe that CBI outperforms all other binary code learning methods on the both datasets.

6 Conclusion

In this paper, a cross-view binary code learning method has been proposed for fast person re-identification. The main advantage of this method is that it hugely speeds up the procedure of the ranking or retrieval stage, when achieving equivalent performance to the state-of-the-art methods. Moreover,

Method	CBI	CMFH	PDH	CMSSH	CCA	CVH
ELF-AUC	**80.28**	78.81	75.63	73.96	73.49	50.12
SCNCD-AUC	**87.43**	86.43	84.30	77.30	72.93	70.94
LOMO-AUC	**89.92**	88.80	84.06	85.04	79.96	58.77
ELF-R1	**0.221**	0.165	0.117	0.114	0.083	0.063
SCNCD-R1	**0.313**	0.222	0.225	0.165	0.218	0.158
LOMO-R1	**0.291**	0.247	0.171	0.190	0.168	0.085

Table 2: AUC and Rank 1 performance comparisons on VIPeR with 316 test persons. R1 denotes Rank 1.

Method	CBI	CMFH	PDH	CMSSH	CCA	CVH
ELF-AUC	**75.14**	73.45	49.87	57.03	68.53	53.59
LOMO-AUC	**80.63**	79.78	49.81	57.58	73.63	52.45
ELF-R1	**0.235**	0.150	0.056	0.102	0.156	0.065
LOMO-R1	**0.306**	0.188	0.058	0.099	0.153	0.059

Table 3: AUC and Rank 1 performance comparisons on CUHK01 with 486 test persons.

two more important points have also been observed. On the one hand, we firstly give an inside view of the intrinsic mechanism that the Hamming distance can be minimised by minimising the Euclidean distance when the learned linear hash functions satisfy the hinge loss constraint. In the future, it is meaningful to give a more compact boundary via the statistical perspective to enable a faster convergence of the algorithm. On the other hand, just dual modules have been used to learn the IDs of different persons. In fact, in a real world scenario, even in a building or a shopping mall, much more than two cameras are installed to monitor the human activities. Therefore, learning the IDs of persons from more than two views is useful. From this point of view, we can see that CBI is just a starting point in this area.

Appendix

Proof of Proposition 1

Proof: The Hamming distance between two binary codes y_a and y_b is defined by:

$$\begin{aligned} D_h(y_a, y_b) &= \sum_k y_a^k \oplus y_b^k \\ &= \sum_k \mathbf{1}(sign(w_a^k x_a) \neq sign(w_b^k x_b)), \end{aligned}$$

where $\mathbf{1}(\cdot)$ is an indicator function. Thus, for any k, we consider two conditions: (1) If $sign(w_a^k x_a) = sign(w_b^k x_b)$, it is obvious that

$$y_a^k \oplus y_b^k = 0 \leq |w_a^k x_a - w_b^k x_b|.$$

(2) If $sign(w_a^k x_a) \neq sign(w_b^k x_b)$, we assume that $sign(w_a^k x_a) = 1$ (Otherwise, same conclusion can be also obtained). There must be $sign(w_b^k x_b) = -1$. Since the two linear projections are both hinge loss constraint satisfied functions, we have $w_a^k x_a \geq 1 - \xi_a^k$ and $w_b^k x_b \leq -1 + \xi_b^k$. So, there is $2 - \xi_a^k - \xi_b^k \leq |w_a^k x_a - w_b^k x_b|$. Provided that $\xi_a^k + \xi_b^k \leq 1$, the following inequation is true:

$$y_a^k \oplus y_b^k = 1 \leq 2 - \xi_a^k - \xi_b^k \leq |w_a^k x_a - w_b^k x_b|.$$

In total, provided with $\mathbf{1}(.)^2 = \mathbf{1}(.)$, we obtain the following conclusion by satisfying $\forall k, \xi_a^k + \xi_b^k \leq 1$:

$$\begin{aligned} D_h(y_a, y_b) &= \sum_k \mathbf{1}^2(sign(w_a^k x_a) \neq sign(w_b^k x_b)) \\ &\leq ||W_a x_a - W_b x_b||_2^2. \end{aligned}$$

Proof of Proposition 3

Proof: The original objective function in Proposition 2 is:

$$\begin{gathered} \{W_a^*, W_b^*\} = \arg\min_{W_a, W_b} -\lambda_2 tr(W_a^T S_{ab} W_b) \\ + \sum_k (\tfrac{1}{2} \sum_{v \in \{a,b\}} ||w_v^k||^2 + C \sum \xi_v^{ki}) \\ s.t. \qquad \forall v \in \{a, b\}, i, k, k_1 \neq k_2 : \\ ((w_v^{k_1})^T w_v^{k_2}))^2 \leq \delta_v, \\ (y_v^{ik} (w_v^k)^T x_v^i) \geq 1 - \xi_v^{ki}, \xi_v^{ki} \geqslant 0. \end{gathered}$$

If we fix all other variables except for w_a^k and ξ_a^{ki}, then the objective function and the constraints become:

$$\begin{gathered} -\lambda_2 {w_a^k}^T S_{ab} w_b^k + \tfrac{1}{2} ||w_a^k||^2 + C \sum \xi_a^{ki} \\ \forall i, k, k_1 \neq k : ((w_a^k)^T w_a^{k_1}))^2 \leq \delta_a, \\ (y_a^{ik} (w_a^k)^T x_a^i) \geq 1 - \xi_a^{ki}, \xi_a^{ki} \geqslant 0. \end{gathered}$$

If the orthogonal constraint of projections in intra-module has been added into the objective function using a balance parameter λ_1, we obtain:

$$\begin{gathered} -\lambda_2 {w_a^k}^T S_{ab} w_b^k + \tfrac{1}{2} ||w_a^k||^2 + C \sum \xi_a^{ki} + \tfrac{\lambda_1}{2} (w_a^k)^T Q_a^k w_a^k, \\ s.t. \forall i, k, (y_a^{ik} (w_a^k)^T x_a^i) \geq 1 - \xi_a^{ki}, \xi_a^{ki} \geqslant 0. \end{gathered}$$

where $Q_a^k = \sum_{j \neq k} w_a^j (w_a^j)^T$. If we set $\Theta_a^k = \lambda_1 Q_a^k + I$ and $s_{ab}^{bk} = \lambda_2 S_{ab} w_b^k$, then we obtain:

$$\begin{gathered} w_a^k = \arg\min \tfrac{1}{2} (w_a^k)^T \Theta_a^k w_a^k - (w_a^k)^T s_{ab}^{bk} + C \sum_i \xi_a^{ki}, \\ s.t. \forall i, k, y_a^{ik} (w_a^k)^T x_a^i \geq 1 - \xi_a^{ki}, \xi_a^{ki} \geqslant 0, \end{gathered}$$

Proof of Equation 10

Proof: For simplicity, we delete the subscripts of views and the index of projections k in this subsection. The optimal parameters w_a^k and ξ_a^{ki} can be obtained by solving the following objective function:

$$\begin{gathered} w = \arg\min \tfrac{1}{2} w^T \Theta w - w^T s + C \sum_i \xi^i, \\ s.t. \quad y^i w^T x^i \geq 1 - \xi^i, \xi^i \geqslant 0. \end{gathered}$$

The objective function becomes a classical convex quadratic programming problem. To simplify the optimisation by transferring inequality constraints to equality constraints, a dual problem is designed. Thus, the Lagrange function can be defined as:

$$\begin{gathered} L(w, \xi, \alpha, \gamma) = \tfrac{1}{2} w^T \Theta w - w^T s + C e^T \xi \\ -w^T X^y \alpha + e^T \alpha - \alpha^T \xi - \gamma^T \xi, \end{gathered}$$

where $e = (1, \cdots, 1)^T$, $\xi = (\xi_1, \cdots, \xi_n)^T$, $\alpha = (\xi_1, \cdots, \alpha_n)^T$, $\gamma = (\xi_1, \cdots, \gamma_n)^T$ and $X^y = (y^1 x^1, \cdots, y^n x^n)$. The gradient with respect to the parameters: $\frac{\partial L}{\partial w} = \Theta w - s - X^y \alpha$ and $\frac{\partial L}{\partial \xi} = Ce - \alpha - \gamma$. Then, the optimal values should satisfy the following constraints:

$$\begin{gathered} w = \Theta^{-1}(s + X^y \alpha); \\ \gamma = Ce - \alpha. \end{gathered}$$

Substituting the above equations into the original Lagrange function, we obtain the dual problem:

$$\begin{gathered} \alpha = \arg\min_\alpha \tfrac{1}{2} \alpha^T (X^y)^T \Theta^{-1} X^y \alpha + (s^T \Theta^{-1} X^y - e^T) \alpha \\ 0 \leq \alpha_i \leq C. \end{gathered}$$

References

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

[Bronstein and Bronstein, 2010] Michael M. Bronstein and Alexander M. Bronstein. Data fusion through cross-modality metric learning using similarity-sensitive hashing. In *Proc. CVPR*, 2010.

[Chen *et al.*, 2015] Dapeng Chen, Zejian Yuan, Gang Hua, Nanning Zheng, and Jingdong Wang. Similarity learning on an explicit polynomial kernel feature map for person re-identification. In *Proc. CVPR*, 2015.

[Cheng *et al.*, 2011] Dong Seon Cheng, Marco Cristani, Michele Stoppa, Loris Bazzani, and Vittorio Murino. Custom pictorial structures for re-identification. In *Proc. BMVC*, 2011.

[Cortes and Vapnik, 1995] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995.

[Ding *et al.*, 2014] Guiguang Ding, Yuchen Guo, and Jile Zhou. Collective matrix factorization hashing for multimodal data. In *Proc. CVPR*, 2014.

[Farenzena *et al.*, 2010] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani. Person re-identification by symmetry-driven accumulation of local features. In *Proc. CVPR*, 2010.

[Gong and Lazebnik, 2011] Yunchao Gong and Svetlana Lazebnik. Iterative quantization: A procrustean approach to learning binary codes. In *Proc. CVPR*, 2011.

[Gong *et al.*, 2013] Shaogang Gong, Marco Cristani, and Shuicheng Yan. *Person Re-Identification*. Advances in Computer Vision and Pattern Recognition. Springer, 2013.

[Gray and Tao, 2008] Douglas Gray and Hai Tao. Viewpoint invariant pedestrian recognition with an ensemble of localized features. In *Proc. ECCV*, 2008.

[Hotelling, 1936] Harold Hotelling. Relations between two sets of variates. *Biometrika*, 28:321–377, 1936.

[Kostinger *et al.*, 2012] Martin Kostinger, Martin Hirzer, Paul Wohlhart, Peter M. Roth, and Horst Bischof. Large scale metric learning from equivalence constraints. In *Proc. CVPR*, 2012.

[Kumar and Udupa, 2011] Shaishav Kumar and Raghavendra Udupa. Learning hash functions for cross-view similarity search. In *Proc. IJCAI*, 2011.

[Lampert and Kromer, 2010] Christoph H. Lampert and Oliver Kromer. Weakly-paired maximum covariance analysis for multimodal dimensionality reduction and transfer learning. In *Proc. ECCV*, 2010.

[Lee *et al.*, 2006] Honglak Lee, Alexis Battle, Rajat Raina, and Andrew Y. Ng. Efficient sparse coding algorithms. In *Proc. NIPS*, 2006.

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

[Liao *et al.*, 2015] Shengcai Liao, Yang Hu, Xiangyu Zhu, and Stan Z. Li. Person re-identification by local maximal occurrence representation and metric learning. In *Proc. CVPR*, 2015.

[Long *et al.*, 2015] Mingsheng Long, Jianmin Wang, , and Philip S. Yu. Quantized correlation hashing for fast cross-modal search. In *Proc. IJCAI*, 2015.

[Loy *et al.*, 2013] Chen Change Loy, Chunxiao Liu, and Shaogang Gong. Person re-identification by manifold ranking. In *Proc. ICIP*, 2013.

[Paisitkriangkrai *et al.*, 2015] Sakrapee Paisitkriangkrai, Chunhua Shen, and Anton van den Hengel. Learning to rank in person re-identification with metric ensembles. In *Proc. CVPR*, 2015.

[Pedagadi *et al.*, 2013] Sateesh Pedagadi, James Orwell, Sergio Velastin, and Boghos Boghossian. Local fisher discriminant analysis for pedestrian re-identification. In *Proc. CVPR*, 2013.

[Prosser *et al.*, 2010] Bryan Prosser, Wei-Shi Zheng, Shaogang Gong, and Tao Xiang. Person re-identification by support vector ranking. In *Proc. BMVC*, 2010.

[Rastegari *et al.*, 2013] Mohammad Rastegari, Jonghyun Choi, Shobeir Fakhraei, Hal Daume III, and Larry S. Davis. Predictable dual-view hashing. In *Proc. ICML*, 2013.

[Wang *et al.*, 2007] Xiaogang Wang, Gianfranco Doretto, Thomas Sebastian, Jens Rittscher, and Peter Tu. Shape and appearance context modeling. In *Proc. CVPR*, 2007.

[Weiss *et al.*, 2008] Yair Weiss, Antonio Torralba, and Rob Fergus. Spectral hashing. In *Proc. NIPS*, 2008.

[Yang *et al.*, 2014] Yang Yang, Jimei Yang, Junjie Yan, Shengcai Liao, Dong Yi, and Stan Z. Li. Salient color names for person re-identification. In *Proc. ECCV*, 2014.

[Zhai *et al.*, 2013] Deming Zhai, Hong Chang, Yi Zhen, Xianming Liu, Xilin Chen, and Wen Gao. Parametric local multimodal hashing for cross-view similarity search. In *Proc. IJCAI*, 2013.

[Zhang *et al.*, 2011] Dan Zhang, Fei Wang, and Luo Si. Composite hashing with multiple information sources. In *Proc. SIGIR*, 2011.

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

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

[Zhao *et al.*, 2014] Rui Zhao, Wanli Ouyang, and Xiaogang Wang. Learning mid-level filters for person re-identification. In *CVPR*, 2014.

[Zheng *et al.*, 2012] Wei-Shi Zheng, Shaogang Gong, and Tao Xiang. Re-identification by relative distance comparison. *IEEE TPAMI*, 2012.