U.P.B. Sci. Bull., Series C, Vol. 82, Iss. 1, 2020 ISSN 2286-3540

LEARNING MULTI-VIEW BINARY CODES FOR SPEEDING UP CROSS-VIEW VEHICLE RE-IDENTIFICATION

Zhiyuan JIN[1], Lu LU[2], Tiejun PAN[1], Jun LIU[3], Shoubiao TAN[2 *]

In this paper, a novel Multi-View Binary Identities (MVBI) algorithm with the help of a hash technique is proposed to vehicle re-indentification (Re-ID). The hashing method can transform the real-valued multiplications into binary XOR operations to speed up the target lookup process. First, MVBI aims at simultaneously minimizing the Hamming distance between the samples of similar vehicles in different views and maximizing that between samples of different vehicles. Hence, the semantic structure of different examples across all views of a same vehicle can be definitely maintained. Then, with considering the constraint condition, a set of hash functions can be learned to project all samples from different views into a suitable common Hamming space. The hash code calculated in this space can capture the discriminant as much as possible. Finally, through the simple Hamming distance calculation, we can realize the efficient vehicle Re-ID. The experimental results in the two bench-mark datasets show that the proposed method significantly outperforms some state-of-the-art methods.

Keywords: Vehicle Re-identification, Binary Codes, Hamming Distance, Cross-view Identities.

1. Introduction

With the rapid economic development, a large number of cars have flooded into the streets and, as a result, traffic safety has become a serious social problem. In order to monitor public transport, most of the traffic jam areas have installed a large number of surveillance cameras, making the acquisition of vehicles more convenient. However, in the majority of cases, it is not possible to obtain all the images of license plates from video recording due to the challenge of perspectives and the environment. Therefore, automatic vehicle re-identification based on other visual features and clues is particularly desirable.

To this end, in recent years, researchers have begun to pay attention to the problem of vehicle re-identification in a wild. One of the most well-known work

[1] College of Dgital Technology and Engineering, Ningbo University of Finance & Economics, Ningbo 315175, China
[2] School of Electronics and Information Engineering, Anhui University, Hefei 230601, China.
[3] Associate professor, School of Management Science and Engineering, Nanjing University of Finance and Economics, Nanjing, 210023, China
[*] Corresponding author: Shoubiao Tan. E-mail: starry226@ahu.edu.cn.

[13] is to first obtain the vehicle characteristics for matching by wireless magnetic sensors and then use dynamic time warping method to calculate the distance of signatures and reduce the recognition error rate through iteration parameters. In order to quickly train a large amount of data, in [16], the problem of re-identification is solved by using a colour histogram of the directional gradient of the SVM linear classifier, in [8], proposed a dataset to facilitate the research of vehicle Re-ID (this paper also uses this experimental dataset). In this method,the high-level semantic information of vehicles was first extracted based on the color features, local texture features and semantic features of CNN learning [6] and then the re-identification was conducted through K-means clustering. At the same time, the magnetic sensor [14] is used to obtain the matched vehicle features, in which the dual window vehicle detection algorithm is first used to obtain the waveform data, and then the features of each vehicle are extracted.

Fig. 1. First row: five different views of the vehicle picture. Second row: different view of the vehicle image that vehicle detection has been resolved.

The Euclidean distance-based metrics are time-consuming in the application of large-scale data, the hash method can improve the retrieval speed, so it can be a good solution to this problem.

- We propose a novel cross multi-modal hashing algorithm which utilizes three constraints to enable fast vehicle re-identification.
- As far as we know, this work is the first attempt to apply the hashing approach to multi-view vehicle re-identification.
- In order to find the optimal solution to the hash function, we propose an iterative method to solve the complex objective function.
- Our experimental research on two different public datasets clearly demon-strates the advantage of our method (MVBI) under cross-view scenarios

2. Related work

This paper focuses on the re-identification of different types of vehicles, which, compared to the Person Re-ID [5], is in the early stage of research and has low attention. As shown in Fig. 1, the method is based on the situation in which the vehicle has been located in the image. It means the location of the vehicle in the image is completely known, and the input information for vehicle re-identification is a picture that just contains the vehicle. Various semantic information such as color and shape etc. are considered in this paper.

Actually, hash algorithm has been used widely in the field of computer vision nearest neighbor search, such as image retrieval, object recognition and image matching, but rarely used in re-identification. The core idea of the hash algorithm is to use the hash function to convert the data into the corresponding binary code string, where the semantic information similarity between the data and the Hamming distance [10] of the corresponding binary code string are consistent. This method can reduce the storage space (binary code storage only need to consume very little storage space) and improve the retrieval speed (bitwise XOR operation can be quickly calculated between the coding string Hamming distance) [11]. Consequently, more and more research interest has been devoted to cross-modal hashing. For instance, the core idea of Cross-Modality Similarity Sensitive Hashing (CMSSH) [1] is to treat the hash function corresponding to each binary code as a weak classifier and learning the hash function by eigenvalue decomposition and AdaBoost algorithm. Similarly, CMFH learns a common subspace by decomposing the data into different modes through the cooperative matrix, and then generating a unified binary code string by the quantization method. Moreover, Predictable Dual-View Hashing (PDH) [12] algorithm to maintain predictability of the the binary codes by applying an iterative optimization method based on block coordinate descent.

We accomplish the vehicle Re-ID by learning a set of hash functions for each view. The MVBI algorithm learns the discriminant binary representation of each vehicle image to make more efficient distance measurements in the Hamming space. In particular, MVBI focuses on learning similar binary codes of similar vehicles from different perspectives but maximizes the Hamming distance between different vehicles of different perspectives.

3. Learning cross-view Binary Identities

Without loss of generality, this paper considers only five views of the vehicle. In fact, five views are also common and universal in the real-world monitoring system. Hence, we can reorganize the original dataset into five training sub-datasets as: $X_a = \{x_a^1, x_a^2, \cdots x_a^n\}$, $X_b = \{x_b^1, x_b^2, \cdots x_b^n\}$, $X_c = \{x_c^1, x_c^2, \cdots x_c^n\}$, $X_d = \{x_d^1, x_d^2, \cdots x_d^n\}$, $X_e = \{x_e^1, x_e^2, \cdots x_e^n\}$, Our aim is to find K hash functions F = $\{f_v^1, f_v^2, \cdots, f_v^k\}$ for each view $v \in \{a, b, c, d, e\}$ and $y_v(k) = f_v^k(x_v)$ to simultaneously minimize the Hamming distance between the samples of similar vehicles in different views and maximize that between samples of different vehicles. In this paper, the hash functions are constructed by a set of linear hyper planes: $W_u = \{w_v^1, w_v^2, \cdots, w_v^K\}$ $W_u = \{w_v^1, w_v^2, \cdots, w_v^K\}$. Thus, for the 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 there is $y_v^i \in \{-1; 1\}^K$ and, for simplicity, we can rewrite it as: $Y_v = sign(W_v^T X_v)$.

3.1 Minimizing the intra-class distance

Our first consideration is to generate a valid binary code for each sample, where the differences between the samples of the same car are minimized. The smaller the distance, the higher the probability that the samples are similar, indicating that they may belong to the same type of vehicle. Therefore, we can match the pair of samples using the Hamming distance calculated on these binary codes.

For a pair of sample sets (X_v, X_u) collected under the two views v and u, we assume that all the image samples between the two sets can be matched. Then, we can define the cumulative Hamming distance between them as:

$$L_{intra}(X_v, X_u) = \sum_i D_h(y_v^i, y_u^i); \tag{1}$$

In order to minimize the Hamming distance calculated based on binary codes between the samples of the same vehicle, it is necessary to minimize the cumulative Hamming distance defined in Eq.1 for all the views to obtain our hash projections in the stage of training.

$$F^* = \arg min \sum_u \sum_v L_{intra}(X_v, X_u); \tag{2}$$

where there are v, u $\in$ (a, b, c, d, e).

3.2 Maximizing the inter-class distance

In fact, merely minimizing the intra-class distance is not enough to determine whether a pair of samples belong to the same vehicle or not. Thus, inspired by the method [7] which constructs a triple loss function to maximize the relative distance between matched and unmatched pairs, we also consider maximizing the Hamming distance between different vehicles. By learning the hash functions for all views, the positive pairs of samples can be pulled closer whilst the negative pairs of samples are pulled far away. In other words, we should simultaneously minimize the intra-class Hamming distance and maximize the inter-class Hamming distance for all the pairs of samples in the learned Hamming space. Therefore, based on these considerations, the following formula can be established for a given pair of sample sets (X_v, X_u):

$$\mathrm{L}_{inter}(X_v, X_u) = \sum_i \sum_{i \neq j} D_h(y_v^i, y_u^j); \tag{3}$$

where there are v, u ∈ (a, b, c, d, e), $y_v^i \in Y_v, y_u^j \in Y_{u.}$

Similarly, in order to maximize the Hamming distance of binary codes between the samples of the different vehicle, the Hamming distance of all views in Eq.3 is summed to obtain the hash projections in the stage of training.

$$F^* = \, arg \;\; max \sum_u \sum_v L_{inter}(X_v, X_u); \tag{4}$$

where there are v, u ∈ (a, b, c, d, e).

3.3 Minimizing the maximum mean difference

Maximum Mean Difference (MMD) is a nonparametric distance measure for the data distribution of two domains [3]. It uses fewer illustration and safer exploration strategies than the existing methods while maintains a strong theoretical guarantee on performance. This method can make use of the information acquired from the source domain to construct the feature extraction model suitable for the target domain to a certain extent. The principle of maximum mean di erence is to find a function, assuming different expectations for two different distributions.In addition, to better preserve the semantic structure of the training in-stances, the distribution of hash codes can be also required to guide the learning of mapping by adding some kinds of constrains. Obviously, this constraint can be achieved by minimizing the mean of the hash codes from different views, that is, minimizing the maximum mean difference (MMD) [4] between different views. Therefore, given two datasets X_u and X_v with their corresponding binary codes Y_u and Y_v, the mean difference can be defined as follows:

$$L_{MMD}(X_u, X_v) \; = \; sup[| \, E(Y_v) - E(Y_u) \, |]; \tag{5}$$

where E(.) is the average of the matrix elements and sup[.] means upper bound, that is, greater than the minimum of all its values.

In addition, it is necessary to minimize the sum of the maximum average differences defined in Eq.5 to constrain the samples.

$$F^* = \, arg \;\; max \sum_u \sum_v L_{MMD}(X_v, \; X_u); \tag{6}$$

where there are v, u ∈ (a, b, c, d, e).

3.4 Overall objective function

In the above sections, we introduce three requirements to obtain the desirable binary codes for the cross-view vehicle re-identification. The aim of learning is to find optimal hash functions which can embed the samples from the original feature space into the Hamming space with satisfying the above requirements. To avoid that the hash functions for each pair of views are learned separately, we integrate all the requirements into a unified framework and all these hash functions can be jointly learned. Since the constraint of distance between vehicles and class, the latter needs to be multiplied by a fixed value because it is not in one order of magnitude. The value n is the number of samples, we obtain:

$$F^* = (n-1)\ arg\ min \sum_u \sum_v L_{intra}(X_v, X_u) + arg\ max \sum_u \sum_v L_{MMD}(X_v, X_u); \quad (7)$$

Substitute (1), (3) into (7) and convert it to find the minimum

$$F^* = arg\ min \sum_u \sum_v \sum_i \sum_{i \neq j} [(n-1) D_h(y_v{}^i, y_u{}^i) - D_h(y_v{}^i, y_u{}^j)]; \quad (8)$$

To integrate the first two constraints, we define a new triplet loss as:

$$L_{triple} = \sum_i \sum_{j \neq i} (D_h(y_v{}^i, y_u{}^i) - D_h(y_v{}^i, y_u{}^j)); \quad (9)$$

where $y_v^i = f_v(x_v^i)$. Combining the Eq.1 and 3, we can see that, when the sample is considered as an anchor, our framework is also related to the triplet loss. However, there are some differences between the two approaches. Obviously, the positive and negative samples come from the same view, whereas the anchor sample comes from a different view. The classical triplet loss is a special case of our framework when only one view is available. To this end, we randomly choose a sample from one view as the anchor. Then, from another view, we select a sample that belongs to the same vehicle with the anchor as the positive sample and another sample that belongs to a different vehicle as the negative sample. It is easy to prove that minimizing the triplet loss in Eq.9 can decrease the corresponding item in Eq.1 and increase the corresponding item in Eq.3.

Therefore, the overall objective function combines the both triplet loss Eq. 9 and semantic structure of the maintained instances in Eq.5 form all views as follows:

$$F^* = \arg\min_F \sum_u \sum_v L_{triplet}(X_u, X_v) + \lambda L_{MMD}(X_u, X_v); \quad (10)$$

where is a hyper-parameter to balance the two items.

The integrated objective function guarantees that each view can learn only one set of hash functions and while the requirements between all the pairs of views can be satisfied. Similar to most of existing hashing methods [17], the sign function in the objective function is omitted and thus the classical gradient descend based optimization strategies can be used. More specifically, the objective function can be expressed as:

$$\begin{aligned}\{W_a,W_b,W_C,W_d,W_e\} = &\mathop{\arg\min}_{W_a,W_b,W_c,W_d,W_e} \sum_{i=1}^{n}\sum_{u}^{k}\sum_{u\neq v}^{k}(n-1)\| W_u^T x_u^i - W_v^T x_v^i \|_F^2 \\ &-\sum_{i=1}^{n}\sum_{j=1,i\neq j}^{m}\sum_{u}^{k}\sum_{u\neq v}^{k}\| W_u^T x_u^i - W_u^T x_v^j \|_F^2 + \lambda\sum_{u}^{k}\sum_{u\neq v}^{k}\| \frac{1}{n}\sum_{i=1}^{n}Y(x_u^i) - \frac{1}{m}\sum_{j=1}^{m}Y(x_v^j)\|_H^2 \\ &+\gamma_1 R(W_a,W_b,W_C,W_d,W_e)\end{aligned} \quad (11)$$

where v, u ∈ (a, b, c, d, e), Y (·) : X → H is a linear projection from the source space to the Hamming space, e.g $Y(x_v^i) = w_v^T x_v^i . \lambda, \gamma_1$ are tradeoff parameters.

3.5 Learning hash functions

The Eq. 11 builds a Lagrange equation and it is non-convex with respect to the five matrix variables W_a, W_b, W_c, W_d, and W_e. Fortunately, it is convex with respect to anyone of them while fixing the others. The optimal solution of the function can be transformed into the extreme of the function. Thus, the function would get the extreme point when the partial derivative of convex function is 0. Therefore, the optimization problem can be solved by following the listed five steps iteratively until convergence:

If fix W_b,W_c,W_d,W_e and let $\frac{\partial L}{\partial W_a} = 0$, then we can obtain:

$$\begin{aligned}W_a &= ((4n-8)X_a^i(X_a^i)^T - X_a^j(X_a^j)^T + 4\lambda E(X_a^i)E(X_a^i)^T + \gamma_1 I)^{-1} \\ &[\sum_{u=\{b,c,d,e\}}(n-1)(X_a^i(X_u^i)^T W_u - X_a^i(X_u^j)^T W_u + \lambda E(X_a^i)E(X_u^i)^T W_u)]\end{aligned} \quad (12)$$

If fix W_a,W_c,W_d,W_e and let $\frac{\partial L}{\partial W_b} = 0$, then we can obtain:

$$\begin{aligned}W_b &= ((4n-7)X_b^i(X_b^i)^T - X_b^j(X_b^j)^T + 4\lambda E(X_b^i)E(X_b^i)^T + \gamma_1 I)^{-1} \\ &[(n-1)X_b^i(X_a^i)^T W_a - X_b^j(X_a^i)^T W_a + \lambda E(X_b^i)E(X_a^i)^T W_a + \\ &\sum_{u=\{c,d,e\}}(n-1)(X_b^i(X_u^i)^T W_u - X_b^i(X_u^j)^T W_u + \lambda E(X_b^i)E(X_u^i)^T W_u)]\end{aligned} \quad (13)$$

If fix W_a,W_b,W_d,W_e and let $\frac{\partial L}{\partial W_c} = 0$, then we can obtain:

$$W_c = ((4n-6)X_c^i(X_c^i)^T - 2X_c^j(X_c^j)^T + 4\lambda E(X_c^i)E(X_c^i)^T + \gamma_1 I)^{-1}$$
$$[\sum_{u=\{a,b\}}(n-1)X_c^i(X_u^i)^T W_u - X_c^j(X_u^i)^T W_u + \lambda E(X_c^i)E(X_u^i)^T W_u + \tag{14}$$
$$\sum_{u=\{d,e\}}(n-1)(X_c^i(X_u^i)^T W_u - X_c^i(X_u^j)^T W_u + \lambda E(X_c^i)E(X_u^i)^T W_u)]$$

If fix W_a, W_b, W_c, W_e and let $\frac{\partial L}{\partial W_d} = 0$, then we can obtain:

$$W_d = ((4n-5)X_d^i(X_d^i)^T - 3X_d^j(X_d^j)^T + 4\lambda E(X_d^i)E(X_d^i)^T + \gamma_1 I)^{-1}$$
$$[\sum_{u=\{a,b,c\}}(n-1)X_d^i(X_u^i)^T W_u - X_d^j(X_u^i)^T W_u + \lambda E(X_d^i)E(X_u^i)^T W_u + \tag{15}$$
$$(n-1)(X_d^i(X_e^i)^T W_e - X_d^i(X_e^j)^T W_e + \lambda E(X_d^i)E(X_e^i)^T W_e]$$

If fix W_a, W_b, W_c, W_d and let $\frac{\partial L}{\partial W_e} = 0$, then we can obtain:

$$W_e = (4(n-1)X_e^i(X_e^i)^T - 4X_e^j(X_e^j)^T + 4\lambda E(X_e^i)E(X_e^i)^T + \gamma_1 I)^{-1}$$
$$[\sum_{u=\{a,b,c,d\}}(n-1)X_e^i(X_u^i)^T W_u - X_e^j(X_u^i)^T W_u + \lambda E(X_e^i)E(X_u^i)^T W_u] \tag{16}$$

where $X_u^i = [x_u^1, x_u^2, \cdots, x_u^n], E(x_u^i)$ represents the mean of different pictures of the same view.

3.6 Realizing vehicle re-identification

First, the optimal projections including W_a, W_b, W_c, W_d and W_e are obtained by MVBI training using the above constraint equations. Second, for the gallery sample set X_u, we can obtain the binary codes in advance by using the corresponding projections $Y_u = sign(W_u^T X_u)$. Then, test sample X_v can be also converted into the a set of hash codes, i.e $y_v = sign(W_v^T x_v)$. Finally, the Hamming distance Eq. 17 between the gallery set and the test sample can be calculated and sorted.

$$D_h(y_v^i, y_u^j) = 1[sign(W_v^T x_v^i) \neq sign(W_u^T x_u^j)] \tag{17}$$

where Dh indicates the Hamming distance and xju represents the jth sample in the dataset X_u ·1[.] is an indicator function, that the total number of unequal records.

Algorithm 1	MVBI training
Input:	Data matrix X_u, $u = a, b, c, d, e$, parameters λ, γ_1, n
Output:	

1. Initialize W_a, W_b, W_c, W_d, W_e by random matrices and center sample set Xu by the mean of samples.

2. repeat

3. Fix W_b; W_c, W_d, W_e, update W_a by Equation 12;

4. Fix W_a, W_c, W_d, W_e, update W_b by Equation 13;

5. Fix W_a, W_b, W_d, W_e, update W_c by Equation 14;

6. Fix W_a, W_b, W_c, W_e, update W_d by Equation 15;

7. Fix W_a, W_b, W_c, W_d, update W_e by Equation 16;

8. **until** convergence.

4. Experiments

MVBI is validated for cross-view vehicle re-identification on two public dataset-CompCars [15] and VeRi [8]. Some example images of the two datasets are shown in Fig.2.

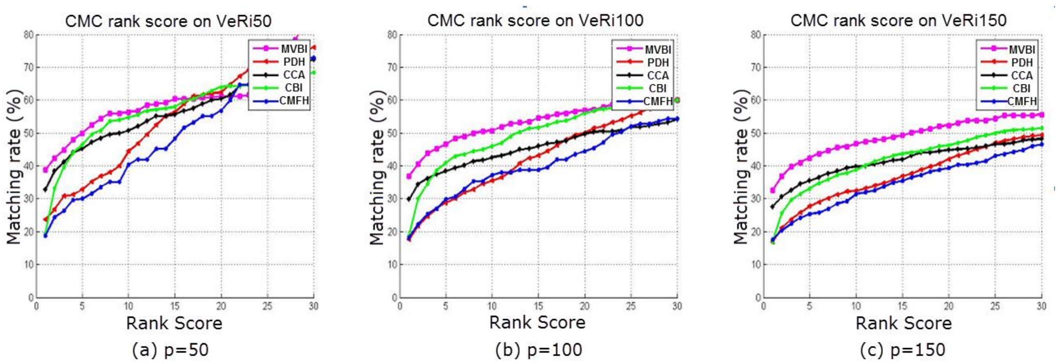

Fig. 2. Using the CMC Curve to VeRi the Google net Feature of the compcars Data Set in Each Hash Method

In order to illustrate the performance and efficiency of MVBI, this paper compares MVBI with three cross-modal hashing methods and one statistical method: canonical correlation analysis (CCA) [9].

Datasets: The CompCars dataset contains data from two scenarios, including images from web-nature and surveillance nature that are widely used in the real-world applications. In particular, the web-nature data contains 163 car makes with 1, 716 car models, covering most of the commercial car models in the recent ten years. There is a total of 136, 727 images capturing the entire cars and 27, 618 images capturing the car parts, where most of them are labeled with attributes and viewpoints.

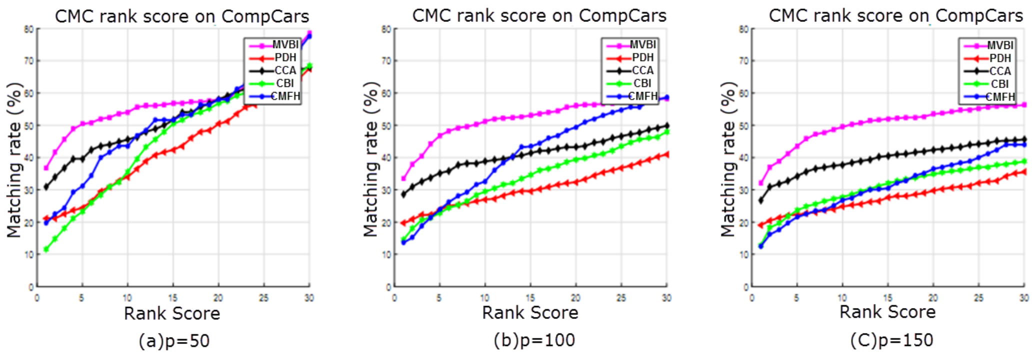

Fig. 3. Using the CMC Curve to Compare the Google net Feature of the VeRi Data Set in Each Hash Method

The surveillance-nature data contains 44,481 car images captured in the front view. Each image in the surveillance-nature partition is annotated with bounding box, model, and color of the car. Overall, the CompCars dataset offers four unique features in comparison to existing car image databases, namely car hierarchy, car attributes, viewpoints, and car parts. To further investigate the attributes of car re-identification, CompCars dataset is reclassified according to the color RGB threshold and randomly selected 3750 images are considered as data for future training and testing. It can capture the high-level semantic information of the vehicle as a semantic feature by extracting the feature from images. Another data set used in this paper, the VeRi, contains over 40,000 bounding boxes of 619 vehicles captured by 20 cameras in unconstrained traffic scene. Moreover, each vehicle is captured by cameras in different viewpoints, illuminations, and resolutions to provide high recurrence rate for vehicle Re-Id. To enable cross-view re-identification, we also label the views of images in datasets VeRi [8], which contains 619 vehicles captured by 20 cameras, with the same setting of CompCars dataset [15].In order to facilitate the operation, the experimental data consisted of 750 sample pairs containing 3,570 images, each pair containing five images from different views of the same vehicle.

In this paper, we randomly divided the datasets into two parts according to a certain percentage. The probe set consists of a picture of each car and the others will be considered as a training set. The experiment will be conducted 10 times and the results are averaged finally. We randomly divided the datasets into two parts according to a certain percentage. The probe set consists of a picture of each car and the others will be considered as a training set. The experiment will be conducted 10 times and the results are averaged finally.

In our experiment, for each data set, we randomly select all the images of the p number of cars for testing, the rest are used to train our model. For example, we can select p = 150 pairs (750 pictures) as a test set and 600 pairs (3000pictures) as a training set. We use the average Cumulative Match Characteristic (CMC) curve over 10 trials to show the accuracy rate. The accuracy rate at ranks: r = 1,5,10, 20 indicates the percentage of the test image with the correct match in the p training set which are ranked in the top r list. Actually, the first rank r = 1 is the most important factor to indicate the performance of a method.

Comparison with other hashing methods, We compared our MVBI with CCA (The method used by CCA is to linearly transform the multidimensional ‘X’ and ‘Y’ into one-dimensional ‘X’ and ‘Y’, and then use the correlation coefficient to see the correlation between ‘X’ and ‘Y’. Changing the data from multi-dimensional to one-bit can also be understood as CCA is performing dimensionality reduction, reducing high-dimensional data to one-dimensional, and then using correlation coefficients for correlation analysis.) [9] and recently proposed multi-modal binary codes learning methods including PDH (The PDH algorithm embeds the proximity of the data samples into the original space, and by creating a cross-view Hamming space, compares the information in the previously incomparable domain with the concept of “predictability”.) [12], CBI (It learns two sets of discriminant hash functions for two different views by simultaneously minimizing their distance in Hamming space and maximizing cross covariance and margin. It embeds the image in the Hamming space and can find similar binary images of the same person captured in different views.) [17] and CMFH (It learns the unified hash code through collective matrix decomposition and a potential factor model of different modalities of an instance. It can not only support cross-view search, but also improve search accuracy by combining multiple view information sources.) [2] on the CompCars and VeRi datasets. Most of the compared methods are used for cross-model search, such as cross retrieval between texts and images.

Table 1 :

Top ranked matching rate (%)on CompCars. p is size of the gallery set (larger p means larger training set) and r is the rank.

Methods	Methods	**MVBI**	PDH	CCA	CBI	CMFH
	r =1	36.80	21.20	30.80	11.60	19.60
p =50	*r* =5	50.40	24.40	39.60	23.20	31.20
	r =10	54.00	34.00	45.60	35.60	43.60
	r =20	58.00	50.40	58.00	56.80	58.00
	r =1	33.40	19.60	28.60	14.60	13.60
p =100	*r* =5	46.80	23.80	35.20	22.80	23.80

	r =10	51.20	27.00	38.80	29.40	32.60
	r =20	56.00	32.40	43.20	39.40	49.40
p =150	*r* =1	32.00	19.07	26.80	12.80	12.53
	r =5	43.60	22.27	34.27	23.60	21.60
	r =10	49.47	24.80	37.73	27.73	26.80
	r =20	53.47	29.73	42.40	34.80	36.40

Table 2:

Top ranked matching rate (%) on VeRi. p is size of the gallery set (larger p means larger training set) and r is the rank.

Methods	Methods	**MVBI**	PDH	CCA	CBI	CMFH
p =50	*r* =1	38.80	23.60	32.80	19.60	18.80
	r =5	50.00	32.80	45.20	46.40	30.00
	r =10	56.40	44.40	50.80	54.80	40.40
	r =20	61.20	62.40	60.40	64.00	56.80
p =100	*r* =1	36.80	17.60	29.80	18.80	18.20
	r =5	46.60	28.80	38.40	41.00	29.80
	r =10	50.80	35.60	42.60	46.00	37.20
	r =20	57.00	50.00	49.80	56.00	44.40
p =150	*r* =1	32.67	16.93	27.60	16.80	17.60
	r =5	42.40	27.73	35.60	33.07	25.33
	r =10	46.80	32.40	39.87	38.80	31.47
	r =20	52.27	42.00	44.80	46.27	39.33

Firstly, we can see that our proposed MVBI is related to the method of cross-view binary identities (CBI). To some extent, our method can be considered as extension of CBI in a setting of multiple views. However, it can preserve the semantic structure of instances better since the mean of binary codes of all samples is minimized, thus resulting in highly correlated binary codes. At the same time, MVBI increases the constraint of Hamming distance between different views and makes the mean of the hash matrix elements similar to those of samples in different views. In contrast, CBI do not take into account the relationship between the pairs of different identities in the different modalities and solutions of CBI may be affected by highly relevant but unimportant variables. Moreover, in the method of PHD, samples have also been projected by maximizing distance differences, but it is significantly different from MVBI from the two following perspectives. On the one hand, both covariance and variance of MVBI have been maximized, but PDH do not consider them. On the other hand, PDH obtains the projection by using the classical SVM directly but the MVBI obtain the solutions by an alternative optimization method in which an extreme value of the dual problem can be obtained to learn the projection. While PHD cannot improve the performance by increasing the number of bits,CMFH with MVBI are similar with each other since the projections of the both methods are obtained by loop iteration.

We compare MVBI with three hash methods including CBI, PDH and CMFH and one statistical method CCA which can produce real representations. Due to the finite level of the variance matrix, the characteristic dimension of the CCA is limited, so the best performance is poor. We compare the proposed method on the two datasets: CompCars and VeRi datasets when p=50, p=100 and p=150.The rank performance of the compared methods is shown in Fig. 2 and 3. From these figures, we can see that 305 the proposed method achieves better results than other methods. No matter what the number of test sets is, our algorithm can get the highest accuracy.

On both the CompCars and VeRi datasets, the results in Table 1 and 2 clearly show that the recognition rate of the proposed MVBI increases significantly by reducing the number of test sets (The number of training sets will also be increased accordingly). This conclusion can be obtained for all ranks from r = 1 to r = 20. Moreover, from the tables, it is clear that our approach is superior to other hash methods in all cases. Finally, it is worth mentioning that these advantages are not very obvious in these tables, but they can also be clearly displayed in the figures.

5. Conclusions

In this paper, we propose a novel multi-view hash algorithm for cross-view vehicle re-identification problem. This method can save storage space (about 87.5% saved.) as well as speed up (upto 10 times) the procedure of identification by converting matching in Euclidean space to binary XOR operations. In the proposed framework, three constraints are defined, which make it possible to simultaneously preserve the intra-class similarity between the pair of the same identity and the inter-class similarity between the pair of different identities in the learned hash space as much as possible. The experimental results of the two datasets: Compcars and VeRi show that the accuracy of the vehicle re-recognition can be improved using MVBI algorithm and the speed of identification is faster than CCA、PDH、CBI、CMFH algorithms.

Acknowledgements

This paper was supported by National Spark Program Project (2015GA701041), Zhejiang Basic Public Welfare Research Program (LGF19G020001), Ningbo Natural Fund (No. 2017A610126), Ningbo Soft Science Fund (2016A10053), 2017 Zhejiang science and technology innovation program for College Students. National Undergraduate Innovative Training

Program (201713001015), Ningbo Science and technology benefiting people project (2017C50024).

REFERENCES

[1]. *M. M. Bronstein, A. M. Bronstein, F. Michel, N. Paragios*, Data fusion through cross-modality metric learning using similarity-sensitive hashing, in: Computer Vision and Pattern Recognition, 2010, pp. 3594-3601.
[2]. *G. Ding, Y. Guo, J. Zhou*, Collective matrix factorization hashing for multimodal data, in: Computer Vision and Pattern Recognition, 2014, pp. 2083-2090.
[3]. *H. Hosseinzadeh, F. Razzazi*, Domain adaptive multiple kernel learning for handwritten digit recognition, in: Electrical Engineering, 2016, pp.1887-1891.
[4]. *B. Kim, J. Pineau*, Maximum mean discrepancy imitation learning, in: Robotics: Science and Systems, 2013.
[5]. *R. W. Kocher*, Personnel and vehicle identification system using three fac-tors of authentication, 2011.
[6]. *A. Krizhevsky, I. Sutskever, G. E. Hinton*, Imagenet classification with deep convolutional neural networks, Communications of the Acm 60 (2013) 2012.
[7]. *H. Liu, Y. Tian, Y. Wang, L. Pang, T. Huang*, Deep relative distance learning: Tell the Difference between similar vehicles, in: Computer Vision and Pattern Recognition, 2016, pp. 2167-2175.
[8]. *X. Liu, W. Liu, H. Ma, H. Fu*, Large-scale vehicle re-identification in urban surveillance videos, in: Multimedia and Expo (ICME), 2016 IEEE International Conference on, IEEE, 2016, pp. 1-6.
[9]. *P. Mazuruse*, Canonical correlation analysis, Journal of Financial Economic Policy 6 (2012) 179-196.
[10]. *M. Norouzi, D. J. Fleet, R. Salakhutdinov*, Hamming distance metric learning, in: Neural Information Processing Systems, 2012, pp. 1061-1069.
[11]. *Y. Peng, Y. Yan, W. Zhu, J. Zhao*, Binary coding-based vehicle image classification, in: International Conference on Signal Processing, 2015, pp. 918-921.
[12]. *M. Rastegari, J. Choi, S. Fakhraei, D. Hal, L. Davis*, Predictable dual-view hashing, in: 30 The International Conference on Machine Learning, 2013, pp. 1328-1336.
[13]. *R. O. Sanchez, C. Flores, R. Horowitz, R. Raj*agopal, P. Varaiya, Vehicle re-identification using wireless magnetic sensors: Algorithm revision, modifications and performance analysis, in: Vehicular Electronics and Safety (ICVES), 2011 IEEE International Conference on, IEEE, 2011, pp. 226-231.
[14]. *Y. Tian, H. H. Dong, L. M. Ji*a, L. I. Si-Yu, A vehicle re-identification algorithm based on multi-sensor correlation, Journal of Zhejiang University, vol. 15, 2014, 372-382.
[15]. *L. Yang, P. Luo, C. L. Chen, X. Tang*, A large-scale car dataset for fine-grained categorization and verification, in: Computer Vision and Pattern Recognition, 2015, pp. 3973-3981.
[16]. *D. Zapletal, A. Herou*t, Vehicle re-identification for automatic video traffic surveillance, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2016, pp. 25-31.
[17]. *F. Zheng, L. Shao*, Learning cross-view binary identities for fast person re-identification., in: IJCAI, 2016, pp. 2399-2406.