Fast Vehicle Re-Identification via Ranked Semantic Sampling Based Embedding

Feng Zheng[1], **Xin Miao**[2], **Heng Huang**[1*]
[1] Department of Electrical and Computer Engineering, University of Pittsburgh
[2] Department of Computer Sciences & Engineering, University of Texas at Arlington
{feng.zheng, heng.huang}@pitt.edu, xin.miao@mavs.uta.edu

Abstract

Identifying vehicles across cameras in traffic surveillance is fundamentally important for public safety purposes. However, despite some preliminary work, the rapid vehicle search in large-scale datasets has not been investigated. Moreover, modelling a view-invariant similarity between vehicle images from different views is still highly challenging. To address the problems, in this paper, we propose a Ranked Semantic Sampling (RSS) guided binary embedding method for fast cross-view vehicle Re-IDentification (Re-ID). The search can be conducted by efficiently computing similarities in the projected space. Unlike previous methods using random sampling, we design tree-structured attributes to guide the mini-batch sampling. The ranked pairs of hard samples in the mini-batch can improve the convergence of optimization. By minimizing a novel ranked semantic distance loss defined according to the structure, the learned Hamming distance is view-invariant, which enables cross-view Re-ID. The experimental results demonstrate that RSS outperforms the state-of-the-art approaches and the learned embedding from one dataset can be transferred to achieve the task of vehicle Re-ID on another dataset.

1 Introduction

Vehicle re-identification (Re-ID) aims at identifying whether a pair of vehicle images collected from different conditions (sensors, views or environments) belong to the same object (Identity) or not. In recent years, a few initial works have been made [Zapletal and Herout, 2016; Liu *et al.*, 2016b; Shen *et al.*, 2017; Wang *et al.*, 2017]. However, most existing methods work on the Euclidean space in which computing similarities is computationally expensive, especially because a large-scale gallery is inevitable in the traffic surveillance system [Zheng and Shao, 2016]. Secondly, most methods either consider the task of cross-camera Re-ID [Liu *et al.*, 2016b] (treating all the views equally) or focus solely on one view. For example, in [Liu *et al.*, 2016a], only the front view is mainly investigated. Virtually, the most difficult task of vehicle Re-ID is in the cross-view setting such as from side view to front view.

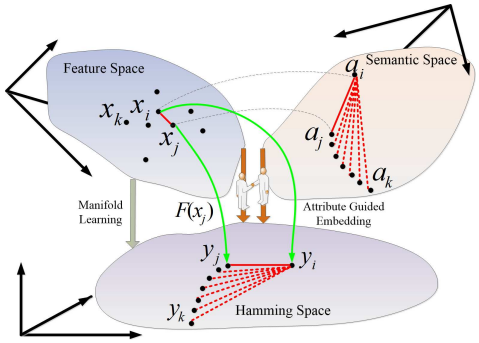


Figure 1: Ranked semantic sampling and embedding. The ranked semantic sampling will facilitate to speed up the training while the preserved ranked semantic distance in embedding can help us discover the identity features via a series of comparisons.

To this end, we focus on learning binary embedding for tackling the challenging task of fast cross-view vehicle Re-ID (see Fig. 1). With success of deep learning [Szegedy *et al.*, 2015], we also adopt a deep architecture as the function of embedding. Generally, the challenging task can be tackled by embedding images from different views into a common code space. Given a sample, the sample of the same identity would be the one that has the minimum Hamming distance in the learned space to it. However, most existing deep embedding methods are insufficient to address the challenging task of cross-view Re-ID, partially because they are specifically designed for the tasks of recognition and categorization.

To address above problems, we propose a Ranked Semantic Sampling (RSS) guided binary embedding for fast vehicle Re-ID. In this method, according to the semantic hierarchies, tree-structured attributes are first constructed to define the semantic distance. Due to the view-invariant properties of attributes [Frome *et al.*, 2013; Amid and Ukkonen, 2015], the relative semantic distance is also view-invariant. Then, to improve the convergence of SGD optimizer, we adopt the attribute tree to guide the mini-batch sampling, in which the samples can be ranked according to the relative semantic distance. Owning to the ranked samples, more relative relationships can be exploited to reduce the frequencies of accessing samples. Furthermore, a probability inequality is derived to smoothly transfer the discrete optimization into a smooth

*To whom all correspondence should be addressed.

problem, in which the SGD optimizer can be used without risk. The theoretical analysis guarantees that the learned Hamming distance can directly preserve the relative semantic distance. Consequently, the proposed RSS enables to effectively measure cross-view similarities and efficiently search the matched samples in a cross-view setting.

In summary, our main contributions are in four-fold: 1) We propose a novel deep binary embedding model which enables fast cross-view vehicle Re-ID. 2) The ranked semantic distance can be preserved so that the learned distance is view-invariant (shown in Fig. 1). Instead employing random sampling as existing methods, to improve the convergence and reduce the frequencies of accessing samples, we introduce a ranked semantic distance guided sampling method. 4) A probability inequality guarantees the transfer from a discrete problem to a smooth objective which SGD can be used.

2 Related Work

Vehicle Re-ID: Recently, a few initial works have been made, including a linear regression model [Zapletal and Herout, 2016], a coarse-to-fine framework [Liu *et al.*, 2016b], a two-branch deep convolutional network [Liu *et al.*, 2016a], orientation invariant features [Wang *et al.*, 2017] and visual-spatio-temporal path proposals [Shen *et al.*, 2017]. Moreover, the two recent works [Zheng and Shao, 2016; Zheng *et al.*, 2016] focus on improving the efficiency of person Re-ID.

Attribute Learning: Recent works [Ferrari and Zisserman, 2007; Hwang and Sigal, 2014] explicitly demonstrate that attribute is essentially beneficial to various computer vision tasks. In [Frome *et al.*, 2013; Hwang and Sigal, 2014], the semantic knowledge learned in the text domain is transferred to train a model for visual object recognition. In [Amid and Ukkonen, 2015], a multi-view triplet embedding is proposed to produce a number of low-dimensional maps, each corresponding to one of the attributes. [Kukliasnky and Shamir, 2015] can choose and observe a small subset of the attributes of each training example.

Relative Distance Loss: In earlier years, distance based loss including contrastive loss [Hadsell *et al.*, 2006] and Kullback-Leibler divergences over all data points [van der Maaten and Hinton, 2008] could be used to dimensionality reduction and visualization. Beyond pair-wise constraints, recently, various contrastive embedding methods such as triplets [Schroff *et al.*, 2015] and quadruplets [Song *et al.*, 2016] etc. are proposed to capture the high-order relative distance. To explore more contrastive information in mini-batches, $(N+1)$-tuplet loss [Sohn, 2016] and histogram loss [Ustinova and Lempitsky, 2016] are also proposed recently.

3 Proposed Method

Intuitively, in order to learn the optimal binary embedding, several questions can naturally be asked: Q1) How to make learning convergence faster? Q2) What types of relationships (similarities) need to be kept in the learned space?

3.1 Cross-View Binary Embedding

Given a sample $x_i^v \in X^v, v = 1, \cdots, V$ of the vth view, we assume that vector $a_i^v \in A^v \in R^{N_a}$ can be used to describe its corresponding semantic attributes, where V and N_a are the number of views and attributes, respectively. X^v and A^v are the sample set and attribute set of the vth view, respectively. In our setup, two samples x_i^u and x_j^v belong to the same object (identity), only if all the corresponding items in the two attribute vectors a_i^u and a_j^v are the same.

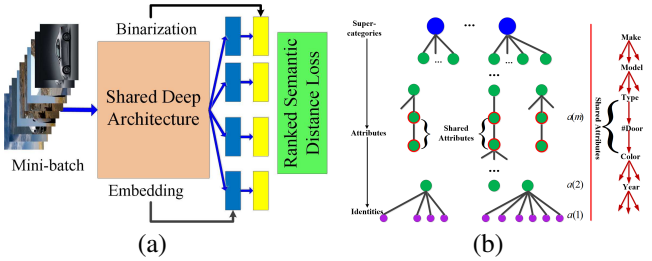


Figure 2: (a) The data flowchart of binary embedding in which GoogLeNet [Szegedy *et al.*, 2015] is used as the shared deep architecture. (b) Attribute tree.

The basic requirement for embedding is that samples collected from any view will be projected onto similar binary codes if they have similar attributes. Assume that F^v is a hash function from a hypothesis space, then the binary codes of x_i^v can be obtained by using $y_i^v = \mathbf{sign}(F^v(x_i^v)), y_i^v \in \{-1, 1\}^K$, where K is the number of binary codes. Obviously, the ideal objective is that, $\forall u, v$, if $a_i^u \equiv a_j^v$, then we have $y_i^v \equiv y_j^u$ and vice versa[1].

Once the hash functions $F^v : v = 1, \cdots, V$ have been learned, given a sample x_i^u of the uth view in the test stage, we can obtain the samples of the same object collected from the vth view by ranking the Hamming distance $D_h(y_i^u, y_j^v)$ between the binary codes of them:

$$\begin{aligned}(x_j^v)^* &= \min_{x_j^v} D_h(y_i^u, y_j^v) \\ &= \min_{x_j^v} D_h(\mathbf{sign}(F^u(x_i^u)), \mathbf{sign}(F^v(x_j^v))), \quad (1)\end{aligned}$$

For simplicity, we denote $F^u(x_i^u)$ as $F(x_i^u)$. Therefore, the basic consideration of this paper is to learn a set of hash functions $F^v : v = 1, \cdots, V$, one for each view, to achieve cross-view ranking. The architecture (hypothetical space) shown in Fig. 2 (a) has a shared deep architecture and V separate fully connected hierarchy will be considered hash functions.

3.2 Ranked Semantic Sampling

In order to ensure that the learned binary code can represent a large number of relationships between semantic entities, a distance measure $D_s(a_i^u, a_j^v)$ needs to be established to describe the difference between two attribute vectors a_i^u and a_j^v of any two samples.

Tree-structured attributes

Generally, we can create structures of attributes with the help of WordNet [Fellbaum, 2000], which provides the semantic hierarchy of nouns. When building large-scale dataset such as ImageNet [Deng *et al.*, 2009], we can also learn attributes from datasets. In order to accomplish the task of cross-view

[1] $\equiv$ means all corresponding items are the same.

Re-ID, we construct an attribute tree as shown in Fig. 2 (b) based on the semantic hierarchy and working scope of attributes. $a(1)$ represents the attribute of a leaf node (lowest level) in the tree. The larger the index l in an attribute $a(l)$ is, the higher the semantic levels of this attribute is. If $l < m$, then we can call attribute $a(m)$ as the parent attribute of the attribute $a(l)$.

Simply, we can divide the attributes into two groups based on the working scope: shared attributes and non-shared attributes. Shared attributes are global variables, so that samples with the same values of these attributes can have different parent attributes, such as type, door number and color. If two samples share a common shared attribute (e.g. color), they may have the same or different parent attributes. However, if two samples share a common non-shared attribute (e.g. model), then they must have the same parent attribute (e.g. car make). In general, the attributes with higher hierarchies are closer to the concept of super-categories (label) whilst the ones with lower hierarchies are closer to the identity. In a word, two samples of the same identity definitely have the identical attributes. Hierarchies can be used to describe the semantic differences between two samples at a high level of understanding. Given two samples x_i^u and x_j^v with attribute vectors a_i^u and a_j^v, we can define the semantic distance between them as:

$$D_s(a_i^u, a_j^v) = \sum_{l \le \mathbf{l}_{ij}} N(a(l)) + \sum_{l \le \mathbf{l}_{ij}} \mathcal{I}(a_i^u(l) \neq a_j^v(l)) \mathcal{I}(a(l)), \tag{2}$$

where $\mathbf{l}_{ij}$ ($1 \le \mathbf{l}_{ij} \le N_a$) is the index of lowest hierarchy where the two samples have different $\mathbf{l}_{ij}$th attributes but share all the same parent attributes above the $\mathbf{l}_{ij}$th hierarchy. $N(a(l))$ denotes that $a(l)$ has $N(a(l))$ child nodes (sub-tree). $\mathcal{I}$ is an indicative function where $\mathcal{I}(a_i^u(l) \neq a_j^v(l)) = 1$ if $a_i^u(l)$ and $a_j^v(l)$ are not the same and $\mathcal{I}(a(l)) = 1$ if $a(l)$ is a shared attribute. Obviously, we have $D_s(a_i^u, a_j^v) = 0$ when they have the same value at leaf node.

Semantic sampling

Given a training data set X with corresponding semantic attribute set A, a small batch of samples can be selected according to the semantic structure T, so that the complex relationships of samples in this mini-batch can be fully explored to guide the learning of embedding.

The sampling process is as follows: First, we randomly select a pair of samples x_1^u and x_2^v with the same attributes from two views u and v at random. Obviously, there is $a_1^u = a_2^v$ and $\mathbf{l}_{12} = 0$. In general, sample x_1^u is considered as an anchor (reference) and x_2^v is considered as a positive sample. Next, we randomly select a sample x_3^v from the view v as the first negative sample by adding one step $\mathbf{l}_{13} = 1$. This example is somewhat similar to an anchor, but with only a different attribute. Then, in order to select more negative samples, we can perform the sampling step to the root of the tree by gradually incrementing $\mathbf{l}_{1j}$. At high hierarchies, when we increase $\mathbf{l}_{1j}$ each time, all the shared attributes of the lower hierarchies should be reconsidered. By changing one at a time in $\sum_{l \le \mathbf{l}_{1j}} \mathcal{I}(a_1^u(l) \neq a_j^v(l)) \mathcal{I}(a(l))$, we select samples which has exactly the same shared attributes to the anchor sample, up to a sample with a completely different shared attribute. Finally, we obtain a mini-batch X_B, where the first sample is an anchor, second one is a positive sample and all others are sorted negative samples.

The characteristics of the sampled mini-batch are distinctive. On the one hand, the adjacent two samples are the hard pairs (To answer the first question Q1: considering hard pairs will make faster convergence). On the other hand, the semantic distance between the anchor and samples from the second to the last one in the mini-batch is monotonically non-decreasing.

Ranked semantic distance loss

Most existing contrastive methods [Schroff *et al.*, 2015; Huang *et al.*, 2016] have some potential limitations in sample sampling: 1) Triplets can only be defined by labels, so fine-gained categories or attributes are not modelled and therefore can not handle more challenging issues, such as identification and verification. 2) Hard samples can only be selected from mini-batches, so selectivity is limited. 3) In most models, mini-batches are randomly generated. Then, in order to learn more triplets or quads, most models must add mini-batch sampling, which can be computationally expensive.

Therefore, in order to improve the efficiency of sampling, this paper proposes a ranked semantic distance loss on the mini-batch to guide the leaning of embedding (To answer the second question Q2: ranked semantic distance shown in Fg. 1). Given a mini-batch X_B sampled according to the semantic structure, we define the ranked semantic distance loss as:

$$\begin{aligned} \mathcal{R}(X_B) \quad = \quad & \sum_i \sum_{j>i} [D_h(y_1^u, y_i^v) - D_h(y_1^u, y_j^v) \\ & + D_s(a_1^u, a_j^v) - D_s(a_1^u, a_i^v)]_+, \end{aligned} \tag{3}$$

where $[\cdot]_+$ operation indicates the hinge function. Minimizing the above loss can guarantee that, in the learned Hamming space, the relative semantic distances between the anchor, the positive and negative samples are preserved. Due to $j > i$, we have $D_s(a_1^u, a_j^v) - D_s(a_1^u, a_i^v) > 0$ according to the semantic structure. Here are a few simple conclusions to be drawn: 1) If $i = 2$, x_i^v is a positive sample and $D_s(a_1^u, a_i^v) = 0$. 2) Furthermore, if there are only two negative samples in the mini-batch, then the proposed loss is the quadruplet loss [Huang *et al.*, 2016]. 3) If there is only one negative sample in the mini-batch and the semantic distance is fixed by a value as well, then it becomes a triplet loss [Schroff *et al.*, 2015].

The above loss is defined when the anchor is immobilized on the first sample of the mini-batch. In fact, when we use the following samples as anchors, we can also explore the indirect relationships implied in the semantic structure.

Theorem 1. *Given three samples a_i^v, a_j^v and a_k^v in the mini-batch X_B in which samples are sorted and sampled according to the semantic tree, if $\mathbf{l}_{1i} < \mathbf{l}_{1j} < \mathbf{l}_{1k}$, then the following distance inequality[2] holds:*

$$D_s(a_i^v, a_j^v) < D_s(a_i^v, a_k^v). \tag{4}$$

This theory means that more comparative relationships would be discovered based on the ranked semantic sampling.

[2] All proofs will be provided in supplementary materials.

Hence, the additional information can further facilitate training of the model and mining of comparative features without adding sample access. $\mathcal{R}(X_B)$ considers the explicit relationships in mini-batch sampling. While based on Theory 1, the implied relationships in the mini-batch X_B can be explored as well. Therefore, we need to minimize the following loss:

$$\mathcal{R}_+(X_B) = \sum_{\mathbf{l}_{1i}<\mathbf{l}_{1j}<\mathbf{l}_{1k}} [D_h(y_i^v, y_j^v) - D_h(y_i^v, y_k^v) + D_s(a_i^v, a_k^v) - D_s(a_i^v, a_j^v)]_+. \quad (5)$$

4 Optimization

In order to find the best function to preserve the semantic distance, we need to minimize the quantity $\mathcal{R}_+(X_B)+\mathcal{R}(X_B)$. Unfortunately, however, the Hamming distance is a discrete variable that is defined based on a $\mathbf{sign}$ function that is not differentiable at zero. The most straightforward way is to replace the $\mathbf{sign}$ function directly with the auxiliary continuous variable $F(x)$, regardless of the difference between y and $F(x)$. The basic problem of this strategy is that the gap is likely to destroy the properties preserved by $F(x)$. In this paper, we solve this problem by minimizing the difference between the Hamming distance and the Euclidean distance in the learning space $F(x)$.

4.1 Quantization Loss

An auxiliary Euclidean distance between x_i^u and x_j^v is defined as $D_e^2(F(x_i^u), F(x_j^v)) = ||F(x_i^u) - F(x_j^v)||_2^2$ in the learned Euclidean space. Obviously, $D_e(F(x_i^u), F(x_j^v))$ is differentiable. The following theory provides an upper bound of a quantization loss between the Hamming distance $4D_h(y_i^u, y_j^v)$ and the auxiliary distance $D_e^2(F(x_i^u), F(x_j^v))$.

Theorem 2. *Given a functional hypotheses space $\mathcal{F}$ and a small value ε, for any two samples x_i^u and x_j^v, the following probability inequality holds:*

$$P(|D_e^2(F(x_i^u), F(x_j^v)) - 4D_h(y_i^u, y_j^v)| > \varepsilon) < \frac{\frac{2}{3K}\mathbf{E}_{F\in\mathcal{F}}(\mathcal{L}(F(x_i^u)) + \mathcal{L}(F(x_j^v))) + C}{\ln \varepsilon^2}, \quad (6)$$

where $C = \frac{2}{3}\ln 2 + 2\ln 3 + 2\ln K$ and $\mathcal{L}(F(x)) = \mathbf{e}_K^T \ln(F(x) \circ F(x) - \mathbf{e}_K)^2$. The symbol $\circ$ denotes the Hadamard product of entry-wise multiplication, $\ln(\cdot)^2$ is an element-wise operator on each entry of x and $\mathbf{e}_K$ in which all items are one is a column vector of length K.

Obviously, searching for a function in the hypothesis space by minimizing $\mathcal{L}(F(x_i^u))$ and $\mathcal{L}(F(x_j^v))$ can reduce the right term of the probability inequality. The smaller the items on the right, the greater the probability that the difference between the two distances $D_e^2(F(x_i^u), F(x_j^v))$ and $4D_h(y_i^u, y_j^v)$ will be within a smaller value ε. This means that minimizing the right item makes $D_e^2(F(x_i^u), F(x_j^v))$ closer to $4D_h(y_i^u, y_j^v)$. In fact, $\mathcal{L}(F(x_i^u))$ is a quantization loss defined on F which projects x_i^u into y_i^u. When all the items of $F(x_i^u)$ are either 1 or -1, $\mathcal{L}(F(x_i^u))$ will reach its minimum.

4.2 Overall Objective

Therefore, we can optimize $\mathcal{R}_+(X_B) + \mathcal{R}(X_B)$ by substituting the auxiliary distance $D_e/4$ for the Hamming distance D_h. Hence, we obtain $\mathcal{R}(X_B) = \sum_i \sum_{j>i}[D_e(F(x_1^u), F(x_i^v))/4 - D_e(F(x_1^u), F(x_j^v))/4 + D_s(a_1^u, a_j^v) - D_s(a_1^u, a_i^v)]_+$ and $\mathcal{R}_+(X_B) = \sum_{\mathbf{l}_{1i}<\mathbf{l}_{1j}<\mathbf{l}_{1k}}[D_e(F(x_i^v), F(x_j^v))/4 - D_e(F(x_i^v), F(x_k^v))/4 + D_s(a_i^v, a_k^v) - D_s(a_i^v, a_j^v)]_+$. To guarantee the learned Hamming distance, the quantization loss $\mathcal{L}(F(x))$ of all samples in the mini-batch X_B should be minimized, simultaneously. Totally, our overall objective can be defined as:

$$\{\mathcal{F}\}^* = \arg\min_{\mathcal{F}} \mathcal{O}(\mathcal{F}) = \arg\min_{\mathcal{F}} \sum_{X_B} (\lambda\mathcal{L}(X_B) + \mathcal{R}_+(X_B) + \mathcal{R}(X_B)), \quad (7)$$

where $\mathcal{L}(X_B) = \mathcal{L}(F(x_1^u)) + \sum_{j\geq 2}\mathcal{L}(F(x_j^v))$ and λ is a balance parameter. Importantly, the objective function is differentiable, so an optimal hash function can be searched directly using a stochastic gradient descent (SGD) based on the mini-batch of structural sampling. The derivatives of $\mathcal{O}$ w.r.t F are discussed in the supplementary material. Finally, the derivative of objective $\mathcal{O}$ w.r.t the parameter θ of function F can be obtained using the chain rule: $\frac{\partial\mathcal{O}}{\partial\theta} = \frac{\partial\mathcal{O}}{\partial F}\frac{\partial F}{\partial\theta}$. θ will be updated during the training stage by using the derivatives on the mini-batches.

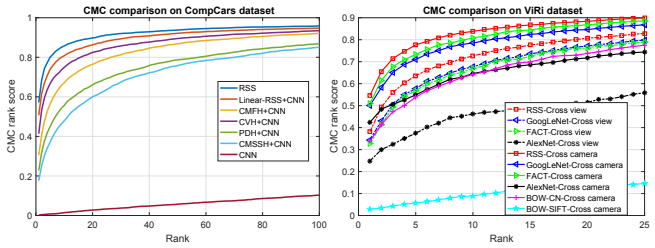
Figure 3: Two datasets: first and second rows - CompCars [Yang *et al.*, 2015]; Third and fourth rows - VeRi [Liu *et al.*, 2016b].

5 Experimental Results

To evaluate the proposed RSS for cross-view vehicle Re-ID, we test RSS on the two recently published vehicle datasets: CompCars [Yang *et al.*, 2015] and VeRi [Liu *et al.*, 2016b] shown in Fig. 3. The balance parameter in the RSS model is set to 0.1. As with most Re-ID methods, Cumulative Match Characteristics (CMC) curves are used to assess performance. The Area Under Curve (AUC) is used to rank the performance of the different methods.

The binary deep architecture shown in Fig. 2 consists of three components: a shared deep architecture based on the GoogLeNet [Szegedy *et al.*, 2015] style Inception models, a view-specific fully connected layer, and a binarization layer. The view-specific embedding layer consists of 640 cells, which are fully connected to the previous layer. The 640 units are divided into 5 groups, each of 128 units corresponds to a view. In the learning phase, the first two components

need to be updated using objective in 7. Batch normalization is used for each mini-batch. In the testing phase, the binary code is obtained by binarizing the embedded values. With effective Boolean operations, efficient vehicle search can be achieved in the learning Hamming space.



(a) (b)

Figure 4: (a) CMC comparison between 7 methods at ranks from 1 to 100. (b) CMC comparison on VeRi. 'RSS-Cross view' denotes that the scores are the results of cross-view Re-ID by RSS.

5.1 Cross-View Vehicle Re-ID

CompCars [Yang *et al.*, 2015] is originally collected for the tasks of fine-grained categorization and verification. This dataset contains a total of $135,846$ images capturing the entire cars from $2,004$ car models. All images have been labelled as one of views including *front* (V1), *rear* (V2), *side* (V3), *front-side* (V4) and *rear-side* (V5). Fortunately, three hierarchies of attributes including *make*, *model* and *year* and the shared attributes between different models including *maximum speed*, *displacement*, *door number*, *seat number* and *type* are given. To make the dataset more suitable to the task of Re-ID, we carefully label each image using the 12 kinds of *colors* which are not offered but important for identification. In total, we select six attributes including *make*, *model*, *type*, *door number*, *color* and *year* to construct the tree shown as 2 (b). $2,000$ images of each view are randomly selected for testing and the remaining samples are used for training.

We compare our proposed RSS based binary embedding with the original work in [Yang *et al.*, 2015] based on CNN [LeCun *et al.*, 1989] and 4 state-of-the-art cross-modal hashing methods, including CMFH [Ding *et al.*, 2014], CVH [Kumar and Udupa, 2011], PDH [Rastegari *et al.*, 2013] and CMSSH [Bronstein and Bronstein, 2010].

Our method can learn the embeddings for all views at the same time. However, since all other hashing methods can handle only two view problems, we implement these methods separately for all pairs of views. The CNN features will be considered as input to all other hashing methods. In addition to learning directly from images (ie, RSS), we also investigate the performance of our model with linear embeddings and CNN features [Yang *et al.*, 2015] (Linear-RSS+CNN). From the Fig. 4 (a), we can see that RSS and Linear-RSS + CNN have always outperformed other methods and RSS achieves better results than that of Linear-RSS+CNN. Moreover, we observe that the original model [Yang *et al.*, 2015] can hardly directly address the challenging task of cross-view Re-ID, but performance can be greatly improved by modelling cross-view relationships. This clearly shows that in order to deal with cross-view tasks, it is necessary to model the view-invariant distance.

The detailed results of cross-view Re-ID are shown in Table 1. We arrive at the same conclusion that the proposed RSS performs the best for cross-view Re-ID at 20 different settings. From this table, we can also see that the *rear-side* view (V5) seems to be easy to recognize, and most of the methods get better results in both settings: 1) *rear-side* and *rear* views, 2) *front-side* and *front* views than others.

5.2 Knowledge Transfer for Real-World Re-ID

VeRi [Liu *et al.*, 2016b] is collected from real-world urban surveillance scenes and contains a total of 776 vehicles taken by 19 cameras. 37,778 images from 576 vehicles are used for training while the remaining 13,257 images from the 200 vehicles were used for testing. Our experimental setup is the same as the original report in [Liu *et al.*, 2016b], but we use only images without regard to license plate recognition. In this section, we focus on the task of acquiring the knowledge transfer capability of RSS from the large-scale dataset CompCars to solve the vehicle Re-ID on the real-world dataset VeRi. The basic RSS model is first trained on CompCars and then fine-tuned on the training set of VeRi.

In order to make VeRi suitable for cross-view Re-ID, we carefully label the camera view of the VeRi dataset followed the setting of the CompCars dataset. Then, by using the fine-tuned model, the binary code of the image can be obtained directly. The three methods of GoogLeNet [Szegedy *et al.*, 2015], FACT [Liu *et al.*, 2016b], and AlexNet [Krizhevsky *et al.*, 2012] are used to compare the performance of cross-view Re-ID without considering the same view pairs. In addition to the cross-Re-ID, we also conduct cross-camera Re-ID under the same settings in [Liu *et al.*, 2016b] and select two other models including Bow-SIFT [Lowe, 1999] and Bow-CN [van de Weijer *et al.*, 2007] from the original VeRi paper [Liu *et al.*, 2016b]. From Fig. 4 (b), we can see that RSS consistently achieves better results than other methods in both cross-view and cross-camera setups. In particular, RSS can outperform FACT, which combines three features, by exploiting the ranked semantic distance. The intrinsic reason is that the ranked semantic distance can help us discover identity features through a series of comparisons. Furthermore, we can observe that the cross-camera Re-ID tasks are much easier than cross-view tasks, regardless of the method used. The potential reason is that most cameras have similar views, and samples from two similar views are easily identified. For example, in the second row of Fig. 3, 10 images of the same car were taken by 10 cameras, but the appearances of the first and fourth images were very similar. In conclusion, experiments show that ranked semantic distances do benefit mining of identity features and can be used to implement actual Re-IDs in both cross-view and cross-camera settings.

5.3 Complexity Analysis of Re-ID

Matching efficiency is the most important factor in a real-world system because CCTV cameras can automatically collect millions of images. However, almost all existing vehicle Re-ID algorithms are mainly focused on improving performance by integrating various complex modules. To the best

Table 1: AUC comparison of cross-view Re-ID between different methods. The largest value is $100(\%)$, when the best results are achieved. To better show the comparisons, we use 'M[i]' refers to ith method, in which M[1]-'RSS', M[2]-'Linear-RSS+CNN', M[3]-'CMFH+CNN', M[4]-'CVH+CNN', M[5]-'PDH+CNN', M[6]-'CMSSH+CNN' and M[7]-'CNN'. V1-2 denotes that the probe is from the *front* view (V1) and the gallery is from the *rear* view (V2).

Cross-view	M[1]	M[2]	M[3]	M[4]	M[5]	M[6]	M[7]	Cross-view	M[1]	M[2]	M[3]	M[4]	M[5]	M[6]	M[7]
V1-2	90.7	81.4	76.6	80.3	67.3	53.4	3.6	V3-4	95.7	92.9	89.6	91.9	87.8	78.9	9.0
V1-3	86.6	84.5	76.8	82.1	58.2	53.5	4.9	V3-5	97.7	93.0	92.3	92.5	88.4	86.4	6.2
V1-4	98.8	97.5	96.4	96.4	94.6	93.2	9.8	V4-1	95.8	96.0	93.0	93.9	90.3	89.8	11.4
V1-5	88.6	88.5	73.2	78.7	61.5	52.5	6.6	V4-2	89.3	88.8	76.1	84.5	66.4	59.5	4.9
V2-1	86.3	84.2	76.5	82.9	65.6	55.7	4.7	V4-3	92.0	90.0	84.0	86.2	76.5	76.0	9.5
V2-3	86.0	80.3	77.7	80.2	67.3	62.2	3.7	V4-5	94.7	91.8	85.7	85.5	79.7	75.8	7.6
V2-4	88.8	89.7	83.0	82.4	70.9	61.4	1.2	V5-1	88.7	85.5	71.7	79.3	63.2	56.2	4.2
V2-5	96.6	95.8	94.7	94.9	93.9	92.3	2.5	V5-2	95.0	95.6	91.8	92.8	88.8	88.7	4.4
V3-1	86.4	83.4	71.2	81.5	51.4	57.1	4.4	V5-3	93.2	90.8	86.9	88.8	79.7	82.3	6.0
V3-2	91.1	86.9	74.8	83.1	61.7	59.0	2.7	V5-4	95.1	92.7	79.9	88.0	73.6	78.0	5.6

Table 2: The comparison results of CMC scores $(\%)$ at 1 and 5. 'Additional' denotes some additional processes which are required by the corresponding methods, $\times$ in this table denotes the real-value multiplication, '-' means no module is used in the methods and N is the number of samples in gallery. Other abbreviations: 'ST' refers to the spatial-temporal modules, 'AT' means average time of matching for each pair, 'XOR' refers to the boolean operation, 'Bow' mentions the bag of words method for quantization and 'CB' means codebooks.

Methods	CMC@1	CMC@5	ST	Plate	Additional	Projection	Time-s	Matching	AT-$e^{-5}s$	Storage
RSS	54.6	77.6	-	-	-	GoogLeNet	0.106	N*128 XOR	0.003	1
Bow-SIFT	2.81	5.82	-	-	SIFT	Bow(10000 CB)	1.228	N*10000$\times$	4.634	2500
Bow-CN	46.56	61.88	-	-	CN	16*Bow(350 CB)	11.213	N*5600$\times$	2.214	1400
AlexNet	42.39	55.09	-	-	-	AlexNet	0.090	N*4096$\times$	1.525	1024
GoogLeNet	49.82	71.16	-	-	-	GoogLeNet	0.106	N*1024$\times$	0.402	256
FACT	50.95	73.48	-	-	SIFT+CN	GoogLeNet+2Bow	12.502	N*16624$\times$	8.182	4156
FACT++	61.44	78.78	STR	SNN	Plate Detection	FACT+SNN	12.635	N*17624$\times$	9.704	4406
OIFE+ST	68.3	89.7	ST	-	20*Key Points	5*CNN	4.735	N*256$\times$	0.125	64
CNN+LSTM	83.49	90.04	LSTM	-	Chain MRF	2*VGG16	1.894	$O(MK^2)$	1600	500

of our knowledge, we are the first efficient algorithm to implement fast vehicle Re-ID and achieve competitive results.

In order to study the complexity of matching, we compared RSS with the above five methods and the other three models: FACT++ [Liu *et al.*, 2016b], OIFE+ST [Wang *et al.*, 2017] and CNN+LSTM [Shen *et al.*, 2017][3]. This is almost all of the validations on the VeRi dataset, which we can find. With the exception of RSS, GoogLeNet and AlexNet these end-to-end algorithms do not have additional modules, the other methods are very complex systems that use multiple CNNs and spatial-temporal regularization. These extra modules are often very computationally expensive. For example, plate detection and recognition using tens of thousands of sliding windows is a daunting task in itself. In order to focus on Re-ID, the position of the tablet in the VeRi dataset is manually annotated. In general, given an unseen probe image, the matching consists of two steps: sample projection and similarity calculation to the samples in the gallery.

Table 2 gives a comparison of the time complexity of projection and matching as well as a comparison of storage requirements. It is worth noting that the computation time for the hand features of the BOW-SIFT and BOW-CN is included, but for simplicity, the calculation time of additional models required by other methods is excluded. From this table, first, we can see that RSS can be much faster than other methods[4] except for two models with similar deep architecture. Especially for those using multiple CNNs, the benefits of RSS are even clearer. Second, more importantly, most of them are even hundreds of times more than RSS, except OIFE+ST, which has a matching time of at least 42 times. In fact, the overall efficiency depends mainly on the number of samples N in the gallery, but it is usually huge in practice. In short, if Re-ID tasks can be done in less than an hour via RSS, it takes nearly two days or more by other means. Finally, the advantages of RSS in storage are also significant, as all other methods require at least 64x capacity to store the features.

6 Conclusion

In this paper, a new binary deep embedding method is proposed for the challenge task of cross-view vehicle re-identification. Its significant advantage is that through a series of comparisons, the ranked semantic distance is view-invariant, which helps us to discover identity features that can be preserved in the learned Hamming space. The validation results show that the preserved semantic distance enables to achieve better results and can transfer the deep architecture learned on one dataset to achieve a real-world vehicle Re-ID. In the future, the ranked semantic distance can be applied to many other areas of computer vision, such as object classification and validation. Moreover, theoretically, one can derive more compact upper bound of inequality in Theorem 2.

[3]We refer the complexity analysis to [Shen *et al.*, 2017].

[4]Quantization in handicraft features is very computationally expensive when the codebook is huge.

Acknowledgement

This work was partially supported by the following grants: NSF-IIS 1302675, NSF-IIS 1344152, NSF-DBI 1356628, NSF-IIS 1619308, NSF-IIS 1633753, NIH R01 AG049371.

References

[Amid and Ukkonen, 2015] Ehsan Amid and Antti Ukkonen. Multiview triplet embedding: Learning attributes in multiple maps. In *ICML*, 2015.

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

[Deng *et al.*, 2009] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

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

[Fellbaum, 2000] Christiane Fellbaum. Wordnet: An electronic lexical database. *Linguistic Society of America*, 76(3):706–708, 2000.

[Ferrari and Zisserman, 2007] Vittorio Ferrari and Andrew Zisserman. Learning visual attributes. In *NIPS*, 2007.

[Frome *et al.*, 2013] Andrea Frome, Greg S. Corrado, Jonathon Shlens, Samy Bengio, Jeffrey Dean, MarcAurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In *NIPS*, 2013.

[Hadsell *et al.*, 2006] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, 2006.

[Huang *et al.*, 2016] Chen Huang, Chen Change Loy, and Xiaoou Tang. Local similarity-aware deep feature embedding. In *NIPS*, 2016.

[Hwang and Sigal, 2014] Sung Ju Hwang and Leonid Sigal. A unified semantic embedding: Relating taxonomies and attributes. In *NIPS*, 2014.

[Krizhevsky *et al.*, 2012] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[Kukliasnky and Shamir, 2015] Doron Kukliasnky and Ohad Shamir. Attribute efficient linear regression with data-dependent sampling. In *ICML*, 2015.

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

[LeCun *et al.*, 1989] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541 – 551, 1989.

[Liu *et al.*, 2016a] Hongye Liu, Yonghong Tian, Yaowei Wang, Lu Pang, and Tiejun Huang. Deep relative distance learning: Tell the difference between similar vehicles. In *CVPR*, 2016.

[Liu *et al.*, 2016b] Xinchen Liu, Wu Liu1, Tao Mei, and Huadong Ma. A deep learning-based approach to progressive vehicle re-identification for urban surveillance. In *ECCV*, 2016.

[Lowe, 1999] David G. Lowe. Object recognition from local scale-invariant features. In *iccv*, 1999.

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

[Schroff *et al.*, 2015] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015.

[Shen *et al.*, 2017] Yantao Shen, Tong Xiao, Hongsheng Li, Shuai Yi, and Xiaogang Wang. Learning deep neural networks for vehicle re-id with visual-spatio-temporal path proposals. In *ICCV*, 2017.

[Sohn, 2016] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *NIPS*, 2016.

[Song *et al.*, 2016] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *CVPR*, 2016.

[Szegedy *et al.*, 2015] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, 2015.

[Ustinova and Lempitsky, 2016] Evgeniya Ustinova and Victor Lempitsky. Learning deep embeddings with histogram loss. In *NIPS*, 2016.

[van de Weijer *et al.*, 2007] Joost van de Weijer, Cordelia Schmid, Jakob Verbeek, and Diane Larlus. Learning color names for real-world applications. In *CVPR*, 2007.

[van der Maaten and Hinton, 2008] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

[Wang *et al.*, 2017] Zhongdao Wang, Luming Tang, Xihui Liu, Zhuliang Yao, Shuai Yi, Jing Shao, Junjie Yan, Shengjin Wang, Hongsheng Li, and Xiaogang Wang. Orientation invariant feature embedding and spatial temporal regularization for vehicle re-identification. In *ICCV*, 2017.

[Yang *et al.*, 2015] Linjie Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. A large-scale car dataset for fine-grained categorization and verification. In *CVPR*, 2015.

[Zapletal and Herout, 2016] Dominik Zapletal and Adam Herout. Vehicle re-identification for automatic video traffic surveillance. In *CVPR*, 2016.

[Zheng and Shao, 2016] Feng Zheng and Ling Shao. Learning cross-view binary identities for fast person re-identification. In *IJCAI*, 2016.

[Zheng *et al.*, 2016] Feng Zheng, Yi Tang, and Ling Shao. Hetero-manifold regularisation for cross-modal hashing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1059–1071, 2016.