VEHICLE VERIFICATION IN TWO NONOVERLAPPED VIEWS USING SPARSE REPRESENTATION

Shih-Chung Hsu* and Chung-Lin Huang†
* Dept. of Electrical Eng., National Tsing-Hua University, Hsin-Chu, Taiwan
E-mail: chvjohnff@gmail.com Tel: +886-3-5742585
†Dept. of M-Commerce and Multimedia Application, Asia University, Tai-Chung, Taiwan
E-mail: clhuang@asia.edu.tw Tel: +886-4-23323456

***Abstract*—Vehicle verification in two different views can be applied for Intelligent Transportation System. However, object appearance matching in two different views is difficult. The vehicle images captured in two views are represented as a feature pair which can be classified as the same/different pair. Sparse representation (*SR*) has been applied for reconstruction, recognition, and verification. However, the *SR* dictionary may not guarantee feature sparsity and effective representation. In the paper, we propose Boost-KSVD method without using initial random atom to generate the *SR* dictionary which can be applied for object verification with very good accuracy. Then, we develop a discriminative criterion to decide the *SR* dictionary size. Finally, the experiments show that our method can generate better verification accuracy compared with the other methods.**

***Keywords*—Vehicle verification, Sparse Representation(*SR*), Boost KSVD.**

I. INTRODUCTION

Vehicle identification has been applied for vehicle tracking and traffic flow analysis in Intelligent Transportation System (ITS). There are three different types of vehicle identification: detection, recognition and verification. Vehicle detection [1, 2] identifies the object as a vehicle. Vehicle recognition [7, 9, 30, 31] identify of the vehicle by finding best match in the gallery set. Vehicle verification [3, 4, 6, 8, 15-17, 24-26] differentiates whether the object pair is the same vehicle or not by using a binary classifier without the gallery set. It can also be used for multi-view vehicle tracking.

The vehicle image representation is always a major issue for vehicle identification. Many researchers demonstrate different approaches by using the designated features. Ballesteros *et al.* [28] use HOG feature to describe the vehicle because the typical appearance of vehicle has clear vertical and horizontal edges. Some researchers [1, 2] apply Gabor features to describe the vehicle because they are similar to human perception. Ying *et al.* [3, 4] construct input feature vectors for SVM classifier by using the nonmetric distance between the input vehicle and representative vehicles. For the object verification, dictionary representation such as non-negative matrix factorization (NMF) [22] or sparse coding [16, 24, 26] present the excellent verification results. However, these dictionary methods have not been widely applied on vehicle verification. Therefore, one of our major works is to test whether these dictionary features and other typical features are effective for verification.

Feature selection encounters two conflict situations: feature representation and noise exhibition. For Principle Component Analysis(PCA), Discrete Cosine Transform(DCT) and Gabor features [21], the noise can only be described by high frequency components or smaller principle components. The background or non-target objects are represented by major principle components or low frequency components. To include high frequency component, Principle Component Regression [27] and Sparse Representation (SR) [9] are proposed based on the regularization constraint. Furthermore, In [10, 11, 23], the sparsity of the sparse vector provides excellent face reconstruction and recognition results[9]. SR has been successfully applied for face verification [16, 24, 25].

With regularization, the feature set may over-describe the training objects. Over-representation with over-complete dictionary does not work very well for classification. Zhang *et al.* [13] show SR is not adaptive to the classification very well because the over-completing coding may include the redundant atoms into the dictionary learning. Wong *et al.* [26] also point out the over-complete dictionary is not an appropriate model to represent the object with the insufficient trainings samples. One explanation to over-representation problem is that the dictionary generation does not follow the Restricted Isometry Property [12] (RIP) which is an influence factor of feature representation. Beside anti-noise and redundant problems, there is another bottleneck for variant object verification. The aligned features from pairs of object in two different views can be applied for face verification [16, 24]. With the appearance variations of the vehicle across disparate views, direct matching may not provide reliable similarity likelihood.

Here, we propose SR dictionary generation for object verification. Different from direct matching methods, we do not warp the object from one view to the other view. First, we propose Boost K-SVD algorithm and consider the RIP for each view. We integrate the features from two views as the testing pair and check the similarity. To avoid over-presentation, we propose the discriminative criteria to determine the size of dictionary that makes the sparse matrix more compact for representation.

This paper is organized as follows. In Section 2, we explore the basic concepts of RIP and introduce the least square solution with different kinds of regularization. In Section 3, we develop the system with sparse coding, K-SVD dictionary learning, initial atom selection, and particle sample selection. In Section 4, we show that our method is better than the other methods, and discuss the amount of atoms and the verification accuracy. Then, we compare the results by using several

different feature representations. We also apply our learning system on the different types of objects. Finally we have the conclusion in Section 5.

II. Related Work

Feature selection is essential for object representation and recognition. The orthogonal component [1, 2, 36] feature is a good solution to represent the input image. However, the unrelated components (e.g., the noise or background) existing in high frequency components or minor principle components may also appear in the major principle components or low frequency components. To approximate the input and reduce the unrelated components or error, there are two strategies: forcing each component with more object details and increasing the size of the feature set.

A. Regularization

To eliminate the small or unrelated component, we decompose the input with the regularization or penalty to ignore the weak components. L2-norm penalty is also called Rigid Regression of which the component truncated version is called PCR. Although L2-norm penalty suppresses the weak components inside the circle, the circle constraint implies the unrelated input may still be constructed by any two or more of major orthogonal components. Furthermore, L1-norm penalty not only constraints the weak components into the diamond range, but also inhabits the unrelated input at the edge in contrast with the circle constraints. Unfortunately, the diamond constraint is not differentiable so that there is no close from to solve the L1-norm penalty problem and the solutions are only resolved by linear programing such as OMP [18], Lasso [19] and LARS [20]. However, L1-norm penalty cannot eliminate the unrelated input. Therefore, L0-norm penalty sticks the input into the exact components and the unrelated input is eliminated by the cross constraint. Unfortunately, L0-norm penalty is a NP-Hard problem. If the feature transform matrix has RIP condition, the L0-norm penalty can be approximated to L1-norm penalty problem and NP-Hard consumption can be reduced to the linear programing problem.

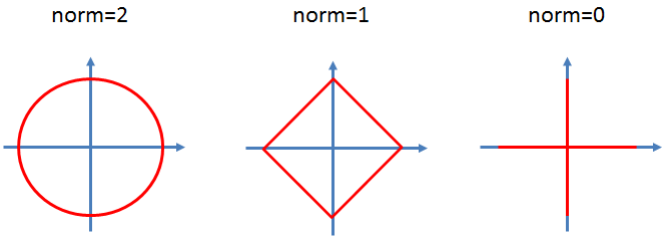


Figure 1. The ranges of different norm penalty functions

B. Dictionary Representation

According to L1-norm penalty and L0-norm penalty, sparse dictionary representation allows more flexibility to represent the data. These methods do not impose that the basis vectors be orthogonal and focus on de-noising by regularization. For SRC, we train the dictionary with enough training samples to span each object space. The classifying dictionary just needs to construct each test sample by a part of dictionary and does not need to consider the uniqueness of feature presentation. K-SVD Dictionary leaning and its online version also devote the learned dictionary to noise removal or image inpainting. However, these dictionary cannot guarantee the fidelity of presentation.

C. Restricted Isometry Property (RIP)

Regularization may show noiseless feature presentation but not guarantee the inverse transform. The representation can be confirmed by RIP. To derive the RIP inequality, we assume there are all unit column vectors in the transform matrix $\boldsymbol{X}$. Then, to find the bound of RIP, we define β as the reconstruction coefficient vector of testing inputs which can be decomposed as $\beta = \beta_1 + \beta_2$ satisfying $X\beta_1 = \pm X\beta_2$. The energy ratio of the reconstruction vector $X\beta$ over the transform coefficient vector β is written as bellow.

$$\frac{\|X\beta\|^2}{\|\beta\|^2} = \frac{\|X\beta_1 + X\beta_2\|^2}{\|\beta_1 + \beta_2\|^2} = \frac{\|\beta_1\|^2 + \|\beta_2\|^2 + 2\sum_i^T \sum_j^{T_c} \beta_i x_i x_j \beta_j}{\|\beta_1\|^2 + \|\beta_2\|^2}$$

$$= \begin{cases} 0, & X\beta_1 = -X\beta_2 \\ 2, & X\beta_1 = +X\beta_2 \end{cases} \quad (1)$$

Therefore, if the transform function is independent, the range of *RIP* inequality is written as bellow.

$$0 < \frac{\|X\beta\|^2}{\|\beta\|^2} < 2 \text{ or } 1 - \delta \le \frac{\|X\beta\|^2}{\|\beta\|^2} \le 1 + \delta, 0 < \delta < 1 \quad (2)$$

Once matrix **X** is over-determined, each component in the matrix becomes more related. Hence, if δ in eq. (2) is small enough, the L1-norm penalty solution is enough close to the L0-norm penalty solution. There are some tight RIP bound [12] to restrict L1-norm penalty into L0-norm penalty. Although finding the RIP condition of the over-determined matrix is a strongly NP-Hard problem, RIP is still considered as a factor to design the near-orthogonal feature transform.

The dictionary-based representation may not provide fully reconstruction property. There is a trade-off between representation and reconstruction. Effective representation property provides effective classification or verification but may not provide effective reconstruction. Therefore, the cleanness and noiseless feature presentation should be constrained with RIP and L1-norm regularization. It is difficult to develop the dictionary with strict RIP, however, it is possible to develop a dictionary with relaxed RIP for the specific object set. The tight *RIP* bound ($\delta \approx 0$) [29] is applied to restrict L1-norm penalty into L0-norm penalty. RIP is an important factor to design the near-orthogonal feature transform. In the following section, we show how to build up the dictionary under RIP condition and demonstrate sparse representation for verification.

III. Vehicle Pair Verification

We develop the vehicle object pair verification by using classification technology. Radial base function support machine vector (RBF SVM) offers a nonlinear decision hyper-plane for many practical classification applications.

Based on the RBF kernel, Euclidean distance among the test instance and support vectors can be used for classification. Euclidean distance is the special case of Mahalanobis distance of which the basis vectors are independent of each other. Therefore, we need to find a set of independent basis vectors to ensure that the vector distance measurement can be evaluated appropriately. Besides the orthogonal PCA basis, sparse coding is another option for component-independent representation satisfying *RIP*. If the *RIP* ratio of the sparse transform is bounded to unity, the *SR* vectors may not be mutual independent. Therefore, discriminative *SR* representation is essential for RBF SVM classifier.

Our object verification system, as shown in Figure 2, consists of (1) Boost K-SVD-based *SR* dictionary training, (2) pair-based RBF SVM object verification, and (3) *SR* dictionary size decision. The dictionary for each view is obtain from Boost K-SVD learning algorithm which has the better *RIP* than the original K-SVD. Boost K-SVD improves the RBF SVM accuracy. However, some atoms obtained from Boost K-SVD may has less representative property, and the verification accuracy may not increase with more atoms. More atoms in dictionary require more computation for decomposing. Therefore, object verification accuracy depends on the appropriate size dictionary. Before RBF SVM training, we need to decide the appropriate size dictionary so that we can avoid exhausting size testing and then apply sparse vector in each view as a SVM input vector for object verification.

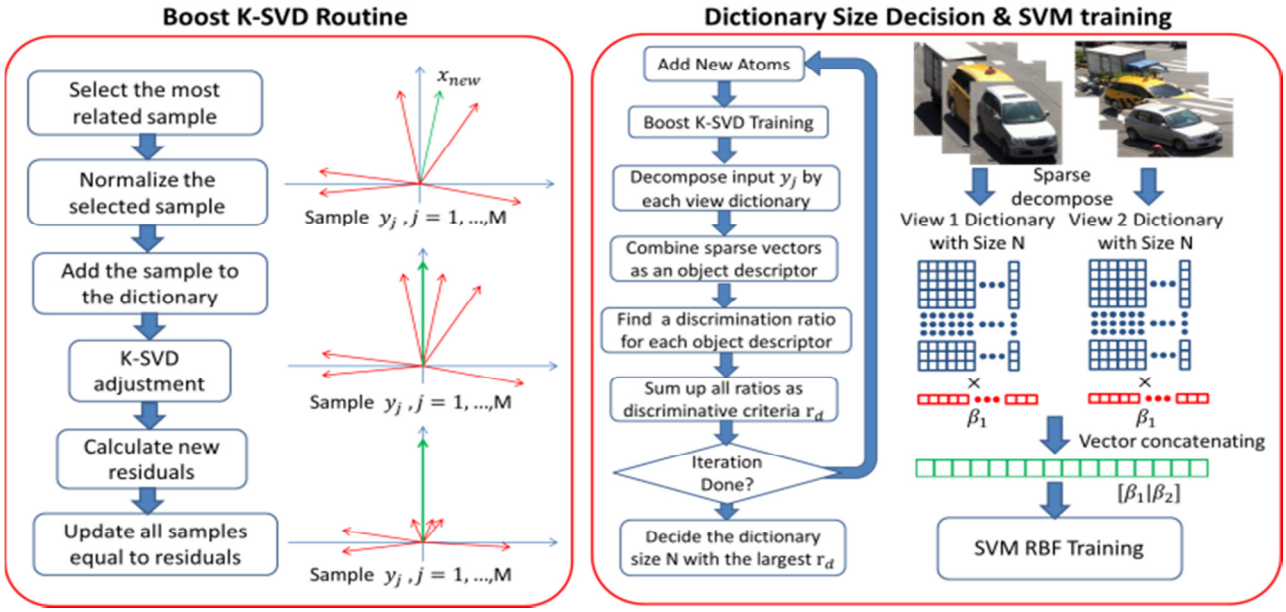


Figure 2. The construction of pair verification.

A. *Boosting initial atoms of K-Singular Value Decomposition (K-SVD)*

Dictionary of appropriate size is required for reconstruction with minimal reconstruction error. K-SVD method with the typical dictionary training algorithm is usually applied on sparse feature extraction [9, 16, 17, 26] and image de-noising [10, 11]. K-SVD algorithm starts iteration with K random selected initial atoms. However, K-SVD algorithm only guarantees the sparsity of the dictionary not RIP. K-SVD may not produce the best sparse dictionary. If the sparse feature vector is generated by using the one-to-one mapping dictionary. In our experiments, the dictionary with RIP shows more accurate verification.

To develop the sparse dictionary, we propose Boost K-SVD to increase the representation accuracy for object verification. Similar to [18], which fits the samples with the most related atoms, our algorithm tunes each initial atom that is most related to all samples. The atom selection finds one sample as the initial atom with maximum summation of correlation with the rest of samples as

$$\boldsymbol{x}_{new} = \text{argmax}_{\mathrm{y}_j} \sum_{i=1}^{N} \left| Corr(\boldsymbol{y}_j, \boldsymbol{y}_i) \right| \qquad (3)$$

and then we add $\boldsymbol{x}_{\text{new}}$ to dictionary $\mathbf{X}$ as $\mathbf{X}=[\mathbf{X}, \boldsymbol{x}_{new}]$, where $\boldsymbol{y}_{\text{i}} \in \mathbf{Y}=\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{\text{N}}\}$ is the set of training samples, and $\boldsymbol{x}_{new}$ is the initial atom of the dictionary. If the initial x_{new} obtained from eq.(3) is closest to the principle atom with the maximum absolute correlation for all training samples, then the iteration of K-SVD can converge at the principle atom faster.

However, the iteration algorithm requires high computation cost to examine each sample to see whether it is closest to the principle atom. To replace the computation, we propose the *initial atom selection algorithm* by using particle filtering. The particle filtering (or random particle sampling process) is similar to the training sample selection process. The particle sampling is equivalent to the training sample selection. The probability of observation $\boldsymbol{z}$ is related to the expect value of the selected samples E[$\boldsymbol{y}$] of which each sample $\boldsymbol{y}_i$ has a weight ω^i. The selected training sample set can be denoted as M particle samples. The identity of the training samples is

modeled by the state variable $\boldsymbol{s}$. The initial weight ω^i of each sample is assigned with $1/N$, where N is the number of training samples. The prior probability distribution of the observation $\boldsymbol{z}$ at iteration k is defined as

$$\mathrm{P}(\mathrm{z_k}|\mathrm{s_k}) = \mathbf{Y} \times \mathrm{E}[\boldsymbol{y}]_k \tag{4}$$

where $\mathbf{Y} \in \mathrm{R}^{\mathrm{D}\times\mathrm{N}}$ is the matrix composed of all training samples, D is the sample dimension, and the expect value of the selected sample set at the k^{th} iteration is denoted as

$$\mathrm{E}[\mathbf{y}]_k = \sum_{\mathrm{i}=1}^{\mathrm{M}} \omega_k^i \times \mathrm{y_i} \tag{5}$$

The k^{th} iterative weight update of the i^{th} sample is defined as

$$\omega_k^i = \omega_{k-1}^i \times \mathrm{P}(\mathrm{z_{k-1}}|s_{k-1} = i) \tag{6}$$

The iteration stops when the reconstruction error is less than certain threshold ε as

$$\left\| \mathbf{Y} - \frac{\mathrm{P}(\mathrm{z_k}|\mathrm{s_k}) \cdot \mathrm{E}[\mathrm{y}]_k}{\|\mathrm{E}[\mathrm{y}]_k\|} \right\|_M < \varepsilon \tag{7}$$

The initial atom searching is described in the following algorithm.

Algorithm 1: Initial Atom Searching:

Denotations: Y is the training set, N is the number of training samples, M is the number of particle samples, and $\mathrm{y_p}$ is a selected sample. Input: $\mathbf{Y}=\{\boldsymbol{y}_1,\ldots\boldsymbol{y}_\mathrm{N}\}$; Output: $\boldsymbol{x}_{\text{new}}$.

Initial:

Set $P(\boldsymbol{z}_k|\boldsymbol{s}_k)$=1/N as an uniform distribution, *k=1*

While eq.(7) is not satisfied do

Begin

Random sample selection using $\mathrm{P}(\mathrm{z_k}|\mathrm{s_k})$

Apply eg. (5) to find $\mathrm{E}[\boldsymbol{y}]_\mathrm{k}$

k=k+1

Apply eq. (4) to compute $\mathrm{P}(\mathrm{z_k}|\mathrm{s_k})$

Update $P(\boldsymbol{z}_k|\boldsymbol{s}_k = i), i = 1, \ldots, N$, (*Importance updating*)

End

Find the initial atom $\boldsymbol{x}_{\boldsymbol{new}} = \mathrm{argmax}_{\boldsymbol{y}_i} |Corr(\boldsymbol{y}_i,\ E[\boldsymbol{y}])|$

The atom training needs to be reformatted to minimize the error by tuning the generated atoms to all training samples. Similarly, Elad *et. al.* [10, 14] proposed K-SVD by using generalized K-means clustering to avoid data overfitting and reweighting each atoms by the sparse vector with L1-norm penalty. The initial atom is randomly selected in K-SVD. Instead, our Boosting K-SVD dictionary construction is based on appropriate initial atom selection so that the spare vectors become more representative and the energy is concentrated on a few components. The dictionary learning is based on minimizing the objective function J defined as,

$$J = min \sum_{i=1}^{N} \frac{1}{2} \left| y_i - \sum_{j=1}^{K} x_j \beta_{ji} \right|^2 + \sum_{j=1}^{K} \lambda |\beta_{ji}|_1 \tag{8}$$

where N is the size of training set, K is the size of the dictionary, β_{ji} is the j^{th} reconstruction coefficient of the i^{th} input sample $\boldsymbol{y}_i$, $\boldsymbol{x}_j$ is the j^{th} atom. To minimize J, we take partial differential of J with respective to x_j and β_{ji} respectively as

$$\frac{\partial J}{\partial x_j} = \sum_{j=1}^{K} \boldsymbol{y_i} \beta_{ji} - \sum_{\mathrm{i}=1}^{N} {\beta_{ji}}^2 x_j = 0$$

$$\text{and} \qquad \frac{\partial J}{\partial \beta_{ji}} = {x_j}^T (y_i - x_j \beta_{ji}) - \lambda sign(\beta_{ji}) = 0 \tag{9}$$

The K-SVD can be obtained by solving eq.(9), where the projection on atom $\boldsymbol{x}_j$ must be higher than the Lagrange multiplier λ if the β_{ji} partial differential holds for each sample i. Here, we modify K-SVD by iterative updating $\boldsymbol{x}_j$ by removing the other components and then averaging the residues in the training samples. We rewrite eq.(9) in the iteration form as

$$x_j(n+1) = \frac{1}{\sum_{\mathrm{i}=1}^{N} {\beta_{ji}}^2} \sum_{\mathrm{i}=1}^{N} \left(\boldsymbol{y_i} - \sum_{\mathrm{k}=1, k\neq j}^{K} x_k(n) \beta_{ki} \right) \cdot \beta_{ji} ,$$

$$\text{subject to } \beta_{ji} = \begin{cases} {x_j}^T(n) y_i - \lambda sign(\beta_{ji}), & |{x_j}^T(t) y_i| - \lambda > 0 \\ 0, & otherwise \end{cases} \tag{10}$$

Third, we minimize J by adjusting the atoms, however, the residues of the training samples may still have other representative components. To obtain the residues, we need to remove the atom component from all training samples and ensure that the dictionary with *RIP* constraints indicating that the atoms for reconstruction are mutual uncorrelated. Then we can pick up a new atom from the set $X_\perp = \mathrm{Y} - \mathrm{X}\beta$, where $X_\perp$ is near orthogonal to X. The process of Boost K-SVD with RIP is described in algorithm 2.

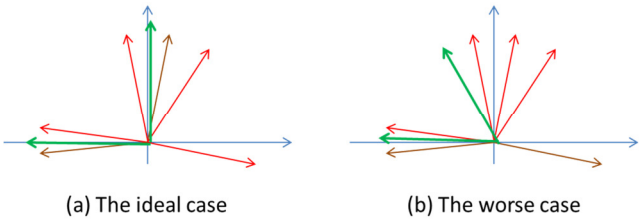


Figure 3. Two different cases in K-SVD algorithm, vectors in red are training samples.

Figure 3 shows the original K-SVD using different initials. In the ideal case, if there are good initial atoms (vectors in brown) selected for K-SVD, the angles among the final atoms (green vectors) will be large after the iteration. Unfortunately, if the improper initials are selected, the final atoms will not be orthogonal after the iteration. The improper initials not only cause larger reconstruction errors, but also generate the dictionary with no RIP property. However, our Boost KSVD keeps the atoms less mutual correlated because the initial atom is iteratively learned from updating training samples satisfying RIP and the atom generation converges faster than K-SVD.

Algorithm 2: Boost K-SVD

Denotations Y is the training set, X is dictionary with final size K, Initial X={ϕ}

Add the initial atom x_{init} to dictionary as **X**=[**X**, x_{init}], atom $\boldsymbol{x}\in\mathbf{X}$

For k = 1 to K

Begin

While $\sum_{j=1}^{k}|x_j(n+1)-x_j(n)|>\varepsilon$

Begin

Use eq. (9) to solve the sparse vectors $\boldsymbol{\beta}$ of all training samples

Update the atoms by eq.(10)

Normalize $x_j(n+1)=x_j(n+1)/norm(x_j(n+1))$

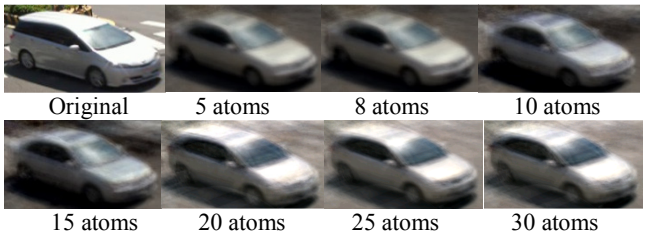
End

Generate a new atom x_{new} from $\boldsymbol{X}_{\perp}=\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}$

Add x_{new} to dictionary **X** as **X**=[**X**, x_{new}]

End

As shown in Figure 4, the original K-SVD reconstructs the input using the holistic vehicle dictionary. Because of the initial random atom selection, the original K-SVD cannot reconstruct the samples from the dictionary of small size. K-SVD requires more atoms for reconstruction.

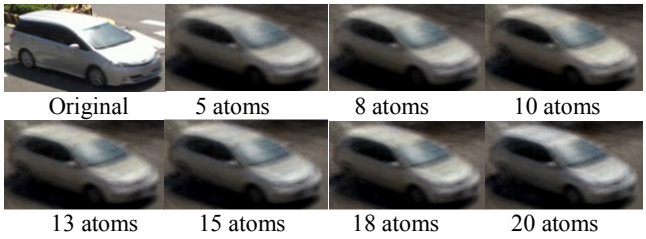


Original 5 atoms 8 atoms 10 atoms

15 atoms 20 atoms 25 atoms 30 atoms

Figure 4. The K-SVD reconstruction

To eliminate the appearance variance from K-SVD, our boost K-SVD is applied to reconstruct the input image so that we can make sure all atoms fitting the training data. In contrast with K-SVD, our boost K-SVD has less shape distortion and less reconstructed background. Figure 5 shows that our boost K-SVD finds more compact dictionary representation, and the reconstruction is more effective than the original K-SVD.



Original 5 atoms 8 atoms 10 atoms

13 atoms 15 atoms 18 atoms 20 atoms

Figure 5. Boost K-SVD reconstruction

B. Pair based object verification

The larger sparsity among the decomposed components will make the reconstructed representation more precise. Our object verification is based on SR representation that demonstrates successful outcome. In face verification [8, 16, 17, 24], the aligned input pairs are represented by the same transform model. However, vehicles usually have different appearance in various views. It is hard to warp the vehicles to the same viewpoint. The scale and illumination normalization are also difficult because the shape of different vehicles and illumination in two views may be totally different.

Here, we simplify the object pair verification problem to a classification problem. The viewing direction is fixed for each camera, the distance between two object pairs in views V_1 and V_2 is defined as

$$\mathrm{P}(t=o|V_1,V_2)=e^{\left(\frac{-\left\|x_{t,V_1}-x_{o,V_1}\right\|^2}{\sigma}\right)}\cdot e^{\left(\frac{-\left\|x_{t,V_2}-x_{o,V_2}\right\|^2}{\sigma}\right)} \quad (11)$$

where x_o is the object vector in the model and x_t is the testing vector and σ is the variance of input. If x_t is enough close to object x_o in views V_1 and V_2, then x_t is similar to x_o. We assume that the object vector pair in two views are mutual independent and the vector dimension is not fixed. The distance between the pair of vectors which are merged into one single vector is described as

$$\mathrm{P}\left(t=o|p_{V_1,V_2}\right)=e^{\left(\frac{-\left\|x_{pt}-x_{po}\right\|^2}{\sigma}\right)} \quad (12)$$

where $x_{pt}=\left[\boldsymbol{x}_{\mathrm{t},V_1}^{\mathrm{T}}|\boldsymbol{x}_{\mathrm{t},V_2}^{\mathrm{T}}\right]^{\mathrm{T}}$ and $x_{ot}=\left[\mathrm{x}_{\mathrm{o},V_1}^{\mathrm{T}}|x_{\mathrm{o},V_2}^{\mathrm{T}}\right]^{\mathrm{T}}$. If the set includes all object pairs, the pair similarity can be examined by inspecting each pair in the set using eq.(12). However, the exhausting pair examining costs too much computation when the data set is enormous. The object verification is developed by training the decision margin function constructed by similar/different object pairs near the decision margin similar to RBF kernel SVM.

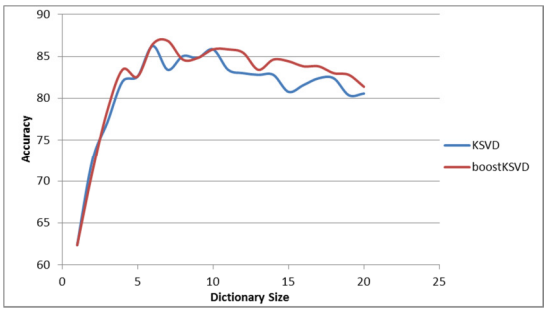
RBF kernel SVM is effective for the object verification. However, when the dimension of the pair vector increases, it incurs the curse of dimensionality that means although two data are very similar, the distance between them is still large. Therefore the less noise on each dimension affects the input with obvious difference and influence the classification results of RBF kernel SVM. Inspired by [3, 4, 6], we represent the pair inputs from two separate views by two *SR* feature vectors which are combined as input x_{pt} as eq.(12). The dimension of the pair of inputs may be different so that warping and alignment distortion cannot be avoided.

In [16], the testing input pair is warped to the frontal view for verification, however, the warped image may cause some shape or context distortion. According to [3, 4, 5], the good object representation will assure the verification accuracy without any warping or fixing the dimensions between the testing object of the input pair. Moreover, nonmetric distance has the similar shrinkage as L1-penalty, so we train each dictionary with the holistic images without warping for each view and use *SR* measurement in that designated view. Here, we present the SR feature vector using boost K-SVD dictionary, and rewrite eq.(9) in RBF SVM classifier form as

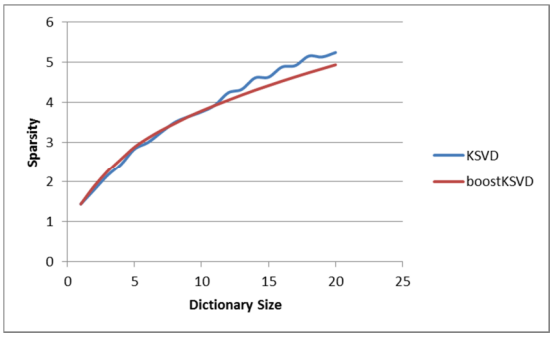
$$\mathrm{f}(\beta_p)=\sum_{s=1}^{N_s}w_s\cdot\exp\left(\frac{-\left\|\beta_p-\beta_s\right\|^2}{\gamma}\right)+\mathrm{b},$$

$$\beta_p=\langle\beta_1|\beta_2\rangle\ \forall\beta_1,\beta_2 \text{ from different dictionaries.} \quad (13)$$

We compare our boost K-SVD and original K-SVD by the sparsity of the dictionary and the verification accuracy using our database which includes the images of two views of 250 individual vehicles. Here, we pick up the samples from the near frontal view as shown in Figure 4 and near side view as shown in Figure 5 to examine the algorithm performance without any alignment or warping. Our boost K-SVD not only has better reconstruction than K-SVD, but also generates the sparser dictionary and achieves better object verification accuracy. Figure 6 (a) shows the accuracy between K-SVD and boost K-SVD and Figure 6(b) shows the sparsity of them.



(a) Accuracy comparison



(b) Sparsity comparison

Figure 6. Comparison between K-SVD and boost K-SVD by sparsity and verification accuracy.

C. *Discriminative criteria of Verification for Dictionary Size Evaluation*

Although Figure 5 shows that our method also has more accurate presentation than K-SVD on verification but keeping adding atoms into the dictionary may not improve verification apparently. It implies that not all atoms are effective for verification. To find the relationship between the discriminant of the sparse vector pair from two dictionary and decide the appropriate size of the dictionary, the discriminative criteria based on RBF kernel is designed as below.

$$r_d = \sum_{n=1}^{N} \frac{max_{n',n' \neq n} exp(-\left|\beta_n - \beta_{n'}\right|^2) \ ,n' \in [1,N]}{max_k \, exp(-|\beta_n - \beta_k|^2) \ ,k \in [1,K]}, \qquad (14)$$

where N is the size of the positive pair set and K is the size of negative pair set, β_n is one of selected positive pair sparse vectors, $\beta_{n'}$ are the other positive pair sparse vectors, and β_k is the negative pair vector. The overall discriminant should be considered with all training pair vectors, so we take the summation of the ratio with the intra estimation over the inter estimation for all pair of samples.

The classifier with the better accuracy will have larger discriminative ratio r_d, due to the fact the between-class distance as the factor in the denominator is larger than the within-class distance. We estimate the discriminative ratio r_d with the nonlinear function *exp*(.), because the ratio of positive/negative distribution as shown in Figure 7 cannot be separated by a linear decision plane. The samples are not only non-separable but also unbalance distributed. Enormous negative samples bring a large within class distance which is neglected in eq.(14).

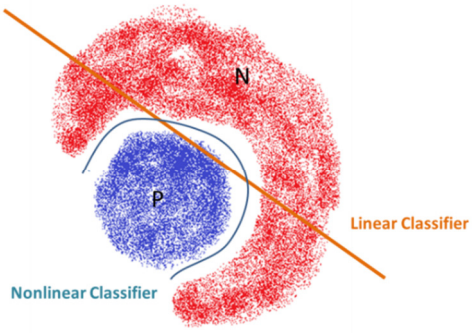
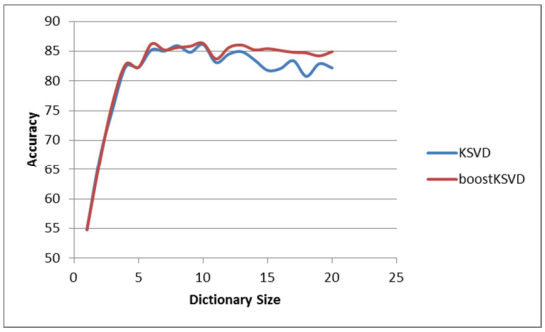


Figure 7. Unbalance distribution of training samples

We apply the discriminative ratio r_d to our database. Figure 8 shows the accuracy vs dictionary size and the discriminative ratio vs dictionary. The highest discriminative ratio of Boost K-SVD occurs at 10 atoms in each view dictionary. Adding more atoms cannot improve the accuracy for object verification. It is because Boost K-SVD has constructed the major components in the dictionary. As shown in Figure 8, although most of K-SVD discriminative ratios are higher than Boost K-SVD, the ratios cannot indicate correct discriminant estimation. It is because K-SVD dictionary has less sparsity without RIP property. It makes the between-class distance in eq.(14) much smaller than the within-class distance. K-SVD cannot be appropriately evaluated by the discriminative criteria which are applied effectively for BoostK-SVD dictionary with RIP condition.



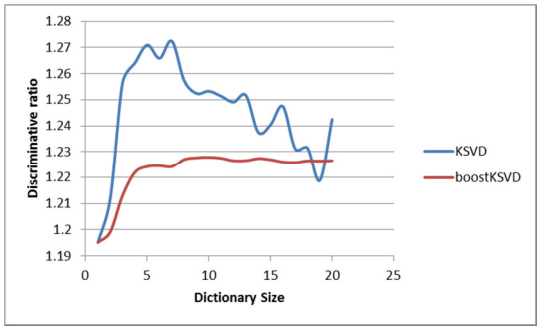


Figure 8. Accuracy and discriminative ratio via dictionary size

IV. EXPERIMENT

In this section, we evaluate the effectiveness of our BoostK-SVD object pair verification among different verification methods and different type of vehicle pairs. Because our Boost KSVD dictionary has good representation under the clustering data, VPTHU (Vehicle pairs captured in-campus or out-campus of Tsing Hua Univ.) is more appropriate to build up side/frontal view dictionaries. We also compare our method performance with some methods on the vehicle database. Meanwhile, we show the over fitting problem in the databases by our discriminative criteria.

In the resampling process, the most related atom to the training sample will be the most likely one to be selected. We adopt Particle filtering to find the initial atoms. To compare the computation complexity, we assume the complexity of exhausting search is $O(N^2)$, and N is the size of training set. The complexity of Particle initializing is $O(N*p*I)$. Here p is the amounts of particle and I is the iteration. If $p*I<N$, then applying Particle filtering requires less computation than the exhausting search. The computation time of atom initialization with 8 particles and 10 time iterations is 0.2179 sec, whereas the exhausting search spends 0.6202 sec.

A. *Database introduction*

The VPTHU consists of in-campus vehicle set and out-campus vehicle set. There are 42 vehicle pairs captured in two-hour video which are recorded at the 20 meter-long road in-campus and 500 pairs captured in the eight-hour video captured at the terminals of the 1.5 km street out-campus. There are many different type of vehicles and various lighting conditions in both video as shown in Figure 9, 10.

Figure 9. Samples of in-campus vehicle set.

Figure 10. Samples of out-campus vehicle set.

B. *Improvement of BoostKSVD*

We estimate our dictionary representation and others by ROC curves as shown in the figure bellow. The threshold of the decision value from the SVM classifier is varied to generate ROC curves. Our Boost K-SVD has the better performance on positive pairs and the similar performance on negative pairs.

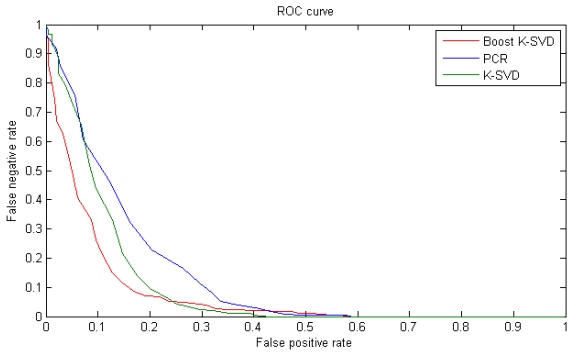
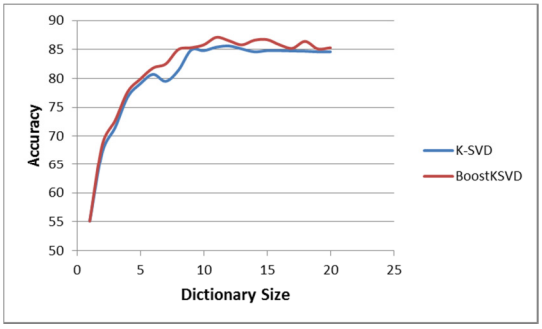


Figure 11. ROC curve.

Boost K-SVD obtaining the sparser components than KSVD has been proven by the experiment on the database in last section. We show that it offers the similar results on the VPTHU out-campus set. In this vehicle experiment, we pick up 250 the same positive pairs and random 750 the different negative pairs as the training samples, and the rest of the positive pairs and other 750 random negative pairs as testing samples. The sparsity comparison between Boost K-SVD and K-SVD is shown as Figure 12. In Figure 12, Boost K-SVD can offer the sparser vectors constrained by minimize L1-norm than K-SVD. It also show the better accuracy of vehicle verification results using our sparse coding Boost K-SVD.



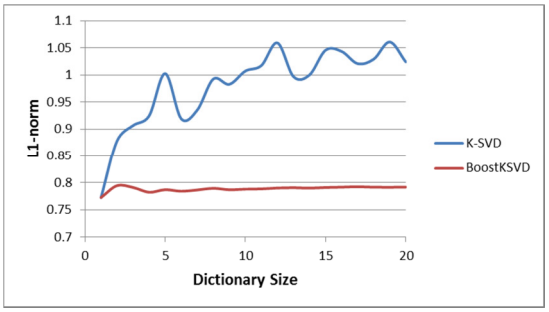


Figure 12. Accuracy and sparsity between K-SVD and BoostKSVD

C. *Other comparisons*

Boost KSVD is applied on the VPTHU out-campus data set as shown in Table 2. In [29], the calibration for vehicles does not work very well because recovering the side appearance of the vehicle to the frontal view by using the warping technique is not accurate.

Table 1. Vehicle verification with/without calibration.

Vehicle verification	Calibration	Non-Calibration
Boost K-SVD	87.2	89.6%
K-SVD	84.2	88.2%
PCR	85%	87.2%

Besides PCR, K-SVD, Boost K-SVD, nonmetric distance[3] is another type of feature applied on vehicle verification. We test our Boost K-SVD and nonmetric distance with the training samples in VPTHU in-campus set consisting of 42 positive pairs and 360 negative pairs. In Table 2, using Boost K-SVD demonstrates better accuracy than nonmetric distance under the same representative components.

Table 2. The Accuracy of Boost KSVD and nonmetric distance

Accuracy		
4 atoms in dictionary	6 atoms in dictionary	8 atoms in dictionary
93.3%	93.8%	94.7%
4 embedded pairs	6 embedded pairs	8 embedded pairs
91%	93%	94%

However, nonmetric distance is not an efficient measure feature for VPTHU out-campus data set as shown in Figure 13. It shows that the embedded pairs cannot represent the testing data effectively and each embedded pair is not unique. The vehicle verification with nonmetric distance is not effective. The other three methods indicate that the redundant representative component and the verification accuracy are no good either. Overall, our method using Boost KSVD demonstrates the best accuracy and stability.

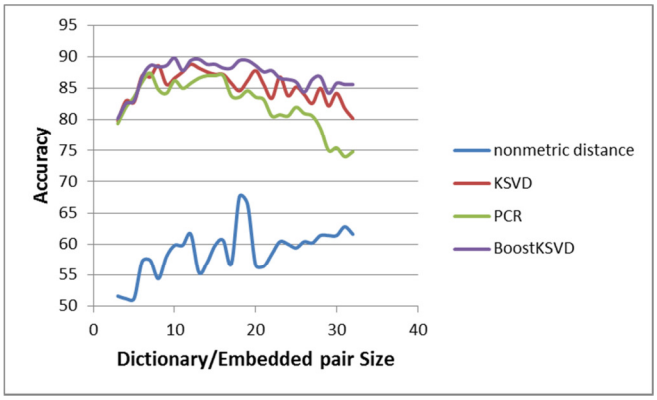


Figure 13. The verification accuracy of using BoostKSVD, KSVD, PCR and nonmetric distance

V. CONCLUSIONS

We develop the object verification system by classifying the object pairs and demonstrate a reliable accuracy. We propose the Boost KSVD to improve the dictionary representation which demonstrate better accuracy and stability of vehicle verification process. Furthermore, we introduce the discriminative criteria to obtain the compact dictionary without the redundant representative components.

REFERENCES

[1] J.-M. Guo , H. Prasetyo and K. Wong, "Vehicle Verification Using Gabor Filter Magnitude with Gamma Distribution Modeling," *IEEE Signal Processing Letters,* vol. 21, pp. 600-604, 18 3 2014.

[2] J. Arróspide and L. Salgado, "Log-Gabor Filters for Image-Based Vehicle Verification," *IEEE Trans. on Image Processing,* pp. 2286-2295, 26 2 2013.

[3] S. Ying, H. S. Sawhney and R. Kumar, "Vehicle Identification between non Overlapping Cameras without Direct Feature Match," *IEEE ICCV*, Beijing, 2005.

[4] Y. Shan, H. S. Sawhney and R. Kumar, "Unsupervised Learning of Discriminative Edge Measures for Vehicle Matching between Nonoverlapping," *IEEE Transactions on PAMI,* pp. 700-711, vol. 4, no.4, 2008.

[5] D. W. Jacobs, D. Weinshall and Y. Gdalyahu, "Class representation and image retrieval with non-metric distances," *IEEE Trans. on PAMI*, pp. 583-600, vol.6, 2000.

[6] Y. Guo, Y. Shan, H. Sawhney and R. Kumar, "PEET: Prototype Embedding and Embedding Transition for Matching Vehicles over Disparate Viewpoints," *IEEE Conf. on Computer Vision and Pattern Recog.*, 2007.

[7] K. Oliba, Y. Sato and K. Ikeuclii, "Visual Learning and Object Verification with Illumination invariance," in *Proceedings of the 1997 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Grenoble, 1997.

[8] X. Wang, C. Zhang and Z. Zhang, "Boosted Multi-Task Learning for Face Verification With Applications to Web

Image and Video Search," *IEEE Conf. on Computer Vision and Pattern Recognition*, Miami, 2009.

[9] J. Wright, A. Y. Yang, A. Ganesh, S. Sastry and Y. Ma, "Robust Face Recognition via Sparse Representation," *IEEE Trans. on PAMI,* pp. 210-227, 30 12 2008.

[10] R. Rubinstein, T. Peleg and M. Elad, "Analysis K-SVD: A Dictionary-Learning Algorithm for the Analysis Sparse Model," *IEEE Trans. on Signal Processing,* pp. 661-677, 24 10 2012.

[11] J. Mairal, F. Bach, J. Ponce and G. Sapiro, "Online Dictionary Learning for Sparse Coding," *Proc. of the Int. Conf. on Machine Learning*, New York, 2009.

[12] E.J Candes, "The restricted isometry property and its implications for compressed sensing," *Comptes Rendus Mathematique*, Vol. 346, Issues 9–10, p.p. 589-592, May 2008.

[13] L. Zhang, M. Yang and X. Feng, "Sparse representation or collaborative representation: Which helps face recognition?," in *IEEE Int. Conf. on Computer Vision*, Barcelona, 2011.

[14] M. Elad, Sparse And Redundant Representations, New York: Springer, 2010.

[15] L. E. Shafey , R. Wallace and S. Marcel, "Face Verification using Gabor Filtering and Adapted Gaussian Mixture Models," *Proc. of Int. Conf. of the Biometrics Special Interest Group*, Darmstadt, 2012.

[16] C.-H. Duan, C.-K. Chiang and S.-H. Lai, "Face Verification With Local Sparse Representation," *IEEE Signal Processing Letters,* pp. 177-180, 3 1 2013.

[17] H. Guo, R. Wang, J. Choi and L. S. Davis, "Face Verification Using Sparse Representations," *IEEE Conf. on Computer Vision and Pattern Recognition Workshops*, Providence, 2012.

[18] S. G. Mallat and Z. Zhang , "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing,* pp. 3397-3415, 6 12 1993.

[19] R. Tibshirani, "Regression Shrinkage and Selection Via the Lasso," *Journal of the Royal Statistical Society,* pp. 267-288, 1994.

[20] B. Efron, T. Hastie, I. Johnstone and R. Tibshirani, "Least Angle Regression," *The Annals of Statistics,* vol. 32, no. 2, pp. 407-499, 2004.

[21] Z. Zhou, T. Deng and X. Lv, "Study for Vehicle Recognition and Classification Based on Gabor Wavelets Transform & HMM," *Int. Conf. on Consumer Electronics, Comm. and Networks,* XianNing, 2011.

[22] D.D. Lee and H.S Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature* 401 (6755), p.p. 788-791, 1999.

[23] R. Rubinstein, M. Zibulevsky and M. Elad, "E cient Implementation of the K-SVD Algorithm using Batch Orthogonal Matching Pursuit," *Computer Science Department – Technical Report*, New York, 2008.

[24] J. Lu, W. Gang, D. Weihong and J. Kui, "Reconstruction-Based Metric Learning for Unconstrained Face Verification," *IEEE Trans. on Information Forensics and Security,* pp. 79-89, 1 2015.

[25] M. Kan, D. Xu, S. Shan and W. Li , "Learning Prototype Hyperplanes for Face Verification in the Wild," *IEEE Trans. on Image Processing*, pp. 3310-3316, 8 2013.

[26] Y. Wong, M. T. Harandi, C. Sanderson and B. C. Lovell, "On robust biometric identity verification via sparse encoding of faces: Holistic vs local approaches," *Int. Joint Conference on Neural Networks*, Brisbane, 2012.

[27] I. T. Jolliffe, "A Note on the Use of Principal Components in Regression," *Journal of the Royal Statistical Society*, pp. 300-303, 1982.

[28] G. Ballesteros and L. Salgado, "Optimized HOG For On-road Video Based Vehicle Verification," *European Signal Processing Conference*, Lisbon, Sept. 2014.

[29] L. Wolf, T. Hassner, and Y. Taigman, "Similarity scores based on background samples," ACCV (2), 2009, pp. 88–97.

[30] Y.-N. Chen, C.-C. Han, C.-T. Wang and K.-C. Fan, "Face Recognition Using Nearest Feature Space Embedding," *IEEE Trans. on PAMI*, pp. 1073-1086, 6 2011.

[31] J. Lu, Y.-P. Tan and G. Wang, "Discriminative Multimanifold Analysis for Face Recognition from a Single Training Sample per Person," *IEEE Trans. on PAMI*, pp. 39-51, 2013.