A Kernel-based l_2 Norm Regularized Least Square Algorithm for Vehicle Logo Recognition

Weiyang Liu†, Yandong Wen‡, Kai Pan†, Hui Li†, and Yuexian Zou†*
†School of Electronic & Computer Engineering, Shenzhen Graduate School, Peking University, P. R. China
‡School of Electronic & Information Engineering, South China University of Technology, P. R. China
Email: wyliu@pku.edu.cn, zouyx@pkusz.edu.cn. *Corresponding Author

***Abstract*—We consider the problem of automatically recognizing the vehicle logos from the frontal views with varying illumination, as well as certain corruption. To better address the problem, a kernel-based l_2 norm regularized least square (RLS) algorithm is proposed in the paper. Kernel technique is smoothly combined with the l_2 norm RLS algorithm to enhance the performance of vehicle logo recognition (VLR). As an extension, the improvement of dictionary is also considered. A simple mechanism of constructing an adaptive online dictionary has been presented and experimented. Experimental results show that our proposed algorithm outperforms the original l_2 norm RLS algorithm and the l_1 norm based algorithms.**

***Index Terms*—Kernel technique, l_2 norm, Regularized least square algorithm, Adaptive online dictionary**

I. Introduction

The applications of computer vision (CV) and machine learning (ML) technique in intelligent transportation systems have drawn more and more attentions from researchers. It becomes an increasingly inevitable trend to apply the CV and ML technique to the future transportation systems. Motivated by rapidly growing CV and ML technique, vehicle logo recognition (VLR) has been stepping into a blooming spring. Various methods for VLR have been proposed and discussed recently in [1]–[5].

Compared to vehicle manufacturer recognition (VMR) [6], [7], VLR refers to a more specific recognition task, namely to find out the vehicle manufacturer by recognizing its brand logo. In general, VLR requires two fundamental steps including detection of candidate vehicle logo in a frame or image and classification of the detected logo region. The detection of logos, also known as vehicle logo location, is a necessary part in any practical VLR system. Consequently, a number of approaches to address the problem of logo detection were proposed in [8]–[12]. As the second step of VLR, the classification of vehicle logos, which is also the main issue we study in this paper, is the most essential part in a VLR system and can directly determine the performance of the VLR system. Note that, when VLR is referred in the following content, it mainly represents the classification of vehicle logos.

To address the VLR problem, the neural network is adopted in [13], [14] to perform the logo recognition. From another perspective, Wang et al. presented a fast coarse-to-fine vehicle logo recognition method via template matching and edge orientation histograms with a good recognition performance. Then Psyllos et al. [1] reported a SIFT-based enhanced matching scheme. They have tested the scheme on a 1200 samples database and obtained a promising recognition rate with a fast processing time.

Different from the previous researches on VLR, this paper dedicates to addressing the problem from a novel insight. Inspired by the collaborative representation classification (CRC) model [15], we apply kernel technique to further enhance its classification ability and propose a kernel-based l_2 norm regularized least square (RLS) algorithm for vehicle logo recognition. Specifically speaking, kernel function is introduced to map the original vehicle logo feature to a higher dimensional space in order to overcome the drawback of CRC in dealing with data with the same direction distribution. Moreover, as an extension, we explore to find a feasible method to construct an adaptive online dictionary. First of all, an index is established to measure the similarity between atoms per class in the dictionary. (Each atom is stacked into one column by the columns of a image sample.) Based on the index, we come up with a mechanism to update the dictionary online. Following this idea, we can construct an adaptive online dictionary. Experimental results have shown that the proposed algorithm has a promising recognition rate with a low computation time. Simulations verify that the online dictionary can outperform the original fixed dictionary under certain conditions.

The outline of this paper is as follows. Section II elaborates our proposed kernel-based l_2 norm RLS algorithm for VLR. In Section III, a feasible construction method of an adaptive online dictionary for our proposed algorithm is briefly discussed. Section IV gives the results and analysis of experiments. Finally, we provide some concluding remarks in Section V.

II. Proposed Method

A. Systematic Perspective

The proposed method contains four modules that are surveillance camera, vehicle logo location, vehicle logo preprocessing and vehicle logo recognition respectively. For better illustration, the systematic framework of the proposed method is shown as follows.

Surveillance Camera. It is usually a video camera and used to take videos of vehicles. In fact, the video will be divided into frames to process in the following modules

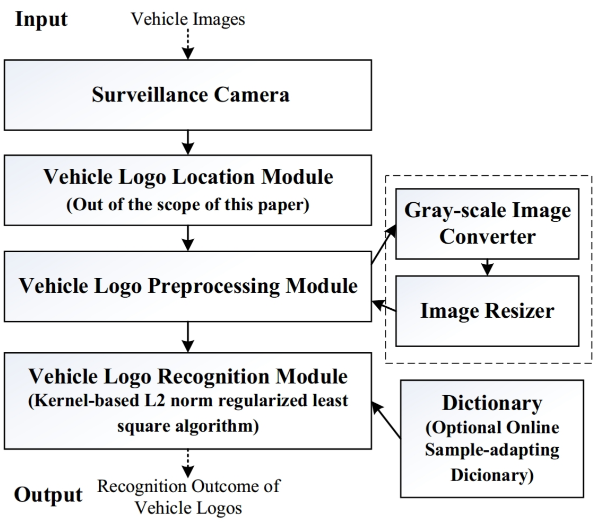

Fig. 1. Framework of the proposed VLR method

Vehicle Logo Location Module. This module is a crucial part in VLR because it decides whether the latter recognition module can work well or not. If the location module can not locate the logo region, the whole VLR task will definitely fail. However, the discussion of this module is out of the scope of the paper, so we just skip this module in the experiments by locating the vehicle logo region manually.

Vehicle Logo Preprocessing Module. This module contains two sub-modules that are gray-scale image converter and image resizer respectively. Gray-scale image converter is used to convert a RGB image to a gray-scale image. Image resizer serves to resize the input vehicle logo image to the same size as the atom in dictionary.

Vehicle Logo Recognition Module. This module plays a core role in VLR since it directly decide the system performance. Our proposed kernel-based l_2 norm RLS algorithm serves as the vehicle logo recognition classifier.

B. Kernel-based l_2 Norm RLS Algorithm

The recognition task can be summarized as using labeled training vehicle logos from k different classes to accurately determine the class to which a input vehicle logo belongs. Suppose the given set of training samples contains total i classes, each with n_i samples, the dictionary associated to the ith class can be written as $\boldsymbol{D}_i{=}[\boldsymbol{v}_{i1},\boldsymbol{v}_{i2},\cdots,\boldsymbol{v}_{in_i}]{\in}\mathbb{R}^{m\times n_i}$ where $\boldsymbol{v}_{ij}, 1{\le}j{\le}n_i$, also called atom, represents a single training sample of vehicle logo. $m{=}wh$ denotes the dimension of each $w\times h$ sample.

The matrix $\boldsymbol{D}$, namely the training dictionary, is defined to represent all training samples. So $\boldsymbol{D}$ can be expressed as

$$\boldsymbol{D} = [\boldsymbol{D}_1, \boldsymbol{D}_2, \cdots, \boldsymbol{D}_k] \in \mathbb{R}^{m\times n} \tag{1}$$

where $n = \sum_{j=1}^{k} n_j$.

The classification is accomplished by solving the following l_2 norm minimization problem [15]:

$$\hat{\boldsymbol{x}} = \arg\min_{x} \|\boldsymbol{x}\|_{l_2}^2 \text{ subj. to } \|\boldsymbol{y} - \boldsymbol{D}\boldsymbol{x}\|_{l_2}^2 \le \varepsilon \tag{2}$$

where $\boldsymbol{x}$ denotes the coding vector over the dictionary $\boldsymbol{D}$ and $\boldsymbol{y}$ represents the test sample of vehicle logo. ε refers to a small error. By the Lagrangian formulation, the l_2 norm problem can be formulated as

$$\hat{\boldsymbol{x}} = \arg\min_{x} \left(\|\boldsymbol{y} - \boldsymbol{D}\boldsymbol{x}\|_{l_2}^2 + \mu\|\boldsymbol{x}\|_{l_2}^2\right) \tag{3}$$

where, μ is the regularization parameter. The problem can be solved by the regularized least square algorithm [15].

We apply the kernel function [16] to the features of logo samples in above problem in order to make training samples more separable and spontaneously enhance the performance of recognition. In essence, kernel function is a nonlinear mapping mechanism for the original feature space. It maps the original feature to a much higher dimensional space so that the features in the higher dimensional space can be linearly separable, which helps to overcome the weakness in processing samples with the same direction distribution.

The nonlinear mapping mechanism via transform Φ can specifically be expressed as

$$\boldsymbol{y} \in \mathbb{R}^n \mapsto \boldsymbol{\phi}(\boldsymbol{y}) = [\phi_1(\boldsymbol{y}), \phi_2(\boldsymbol{y}), \cdots, \phi_s(\boldsymbol{y})] \in \mathbb{R}^s \tag{4}$$

where $\boldsymbol{\phi}(\boldsymbol{y}){\in}\mathbb{R}^s$ is the high dimensional feature associated to the sample $\boldsymbol{y}$ and therefore $s{\gg}n$. $\boldsymbol{d}_i^{[j]}$ is defined to denote the ith atom in $\boldsymbol{D}$ belonging to the jth class, where $1{\le}i{\le}\sum_{j=1}^{k} n_j$, but the class information $[j]$ will be removed for the convenience of description. According to the nonlinear mapping mechanism, the original dictionary $\boldsymbol{D}$ becomes a much higher dimensional one: $\boldsymbol{\Phi}{=}[\boldsymbol{\phi}(\boldsymbol{d}_1), \boldsymbol{\phi}(\boldsymbol{d}_2), \cdots, \boldsymbol{\phi}(\boldsymbol{d}_k)]{\in}\mathbb{R}^{s\times k}$. As a result, the representation form of the test sample becomes $\boldsymbol{\phi}(\boldsymbol{y}) = \boldsymbol{\Phi}\boldsymbol{x}$. So (3) is formulated as

$$\hat{\boldsymbol{x}} = \arg\min_{x} \|\boldsymbol{x}\|_{l_2}^2 \text{ subj. to } \|\boldsymbol{\phi}(\boldsymbol{y}) - \boldsymbol{\Phi}\boldsymbol{x}\|_{l_2}^2 < \varepsilon \tag{5}$$

in which a matrix needs to be constructed to reduce the dimension. By introducing matrix $\boldsymbol{R} \in \mathbb{R}^{s\times c}$, we get

$$\boldsymbol{R}^T\boldsymbol{\phi}(\boldsymbol{y}) = \boldsymbol{R}^T\boldsymbol{\Phi}\boldsymbol{x} \tag{6}$$

where $\boldsymbol{R}$ is related to the samples mapped into the kernel space. Normally speaking, each atom in $\boldsymbol{R}$ is a linear combination of samples in kernel feature space. Namely

$$\boldsymbol{R} = \boldsymbol{\Phi}\boldsymbol{\Psi} = [\boldsymbol{\phi}(\boldsymbol{d}_1), \cdots, \boldsymbol{\phi}(\boldsymbol{d}_k)] \cdot [\boldsymbol{\psi}_1, \cdots, \boldsymbol{\psi}_c] \tag{7}$$

where $\boldsymbol{R}{=}\{\boldsymbol{R}_1, \cdots, \boldsymbol{R}_s\}$ and $\boldsymbol{\psi}_i$ is the linear projection coefficients corresponding to the $\boldsymbol{R}_i{=}\sum_{j=1}^{k} \psi_{i,j}\boldsymbol{\phi}(\boldsymbol{d}_j){=}\boldsymbol{\Phi}\boldsymbol{\psi}_i$. $\boldsymbol{\Psi}{\in}\mathbb{R}^{k\times c}$ is called pseudo-transformation matrix. Then we put (7) into (6) and obtain

$$(\boldsymbol{\Phi}\boldsymbol{\Psi})^T\boldsymbol{\phi}(y) = (\boldsymbol{\Phi}\boldsymbol{\Psi})^T\boldsymbol{\Phi}\boldsymbol{x} = \boldsymbol{\Psi}^T(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\boldsymbol{x}) \tag{8}$$

from which, (9) can be derived by introducing the kernel function $K(\boldsymbol{d}_i, \boldsymbol{d}_j){=}\langle\boldsymbol{\phi}(\boldsymbol{d}_i), \boldsymbol{\phi}(\boldsymbol{d}_j)\rangle{=}\boldsymbol{\phi}(\boldsymbol{d}_i)^T\boldsymbol{\phi}(\boldsymbol{d}_j)$. In our proposed algorithm, we use Gaussian radial basis function (RBF) kernels $K(\boldsymbol{y}_i, \boldsymbol{y}_j){=}\exp(-\beta\|\boldsymbol{y}_i^T - \boldsymbol{y}_j\|_2^2)$.

$$\boldsymbol{\Psi}^T\boldsymbol{K}(\boldsymbol{D}, \boldsymbol{y}) = \boldsymbol{\Psi}^T\boldsymbol{G}\boldsymbol{x} \tag{9}$$

where $\boldsymbol{K}(\boldsymbol{D},\boldsymbol{y})=[K(\boldsymbol{d}_1,\boldsymbol{y}),\cdots,K(\boldsymbol{d}_k,\boldsymbol{y})]^T$. $\boldsymbol{G}$ ($G_{ij} = K(\boldsymbol{d}_i,\boldsymbol{d}_j)$) is defined as the kernel Gram matrix that is symmetric and positive semi-definite according to Mercer's theorem. Since both $\boldsymbol{G}$ and $K(\boldsymbol{d}_i,\boldsymbol{d}_j)$ are known, we should find $\boldsymbol{\Psi}$ instead of finding $\boldsymbol{R}$.

Due to the low complexity of the regularized least square algorithm, the algorithm still works well without performing dimensionality reduction in the feature space. So we simply let $\boldsymbol{\Psi}$ be an identity matrix. Thus, (5) can be formulated to

$$\begin{aligned}\hat{\boldsymbol{x}} &= \arg\min_x \left(\|\boldsymbol{\Psi}^T\boldsymbol{K}(\boldsymbol{D},\boldsymbol{y})-\boldsymbol{\Psi}^T\boldsymbol{G}\boldsymbol{x}\|_{l_2}^2+\mu\|\boldsymbol{x}\|_{l_2}^2\right)\\ &= \arg\min_x \left(\|\boldsymbol{K}(\boldsymbol{D},\boldsymbol{y})-\boldsymbol{G}\boldsymbol{x}\|_{l_2}^2+\mu\|\boldsymbol{x}\|_{l_2}^2\right)\end{aligned} \tag{10}$$

to which the regularized least square algorithm is applied so as to solve the l_2 norm problem. Specifically speaking, we give the new form of dictionary $\boldsymbol{D}'=\boldsymbol{\Psi}^T\boldsymbol{G}$ and define $\boldsymbol{P}'$ as the coding basis in our proposed algorithm. Namely

$$\begin{aligned}\boldsymbol{P}' &= \left((\boldsymbol{\Psi}^T\boldsymbol{G})^T(\boldsymbol{\Psi}^T\boldsymbol{G})+\mu\cdot\boldsymbol{I}\right)^{-1}\left(\boldsymbol{\Psi}^T\boldsymbol{G}\right)^T\\ &= (\boldsymbol{G}^T\boldsymbol{G}+\mu\cdot\boldsymbol{I})^{-1}\boldsymbol{G}^T\end{aligned} \tag{11}$$

which can be directly used in the regularized least square algorithm. The test sample $\boldsymbol{y}$ is expressed as $\boldsymbol{y}=\boldsymbol{\Psi}^T\boldsymbol{K}(\boldsymbol{D},\boldsymbol{y})=\boldsymbol{K}(\boldsymbol{D},\boldsymbol{y})$. The specific algorithm flow is shown as follows.

Kernel-based l_2 Norm RLS Algorithm

1. Normalize the columns of $\boldsymbol{D}'=\boldsymbol{G}$ to unit l_2-norm.
2. Represent $\boldsymbol{y}=\boldsymbol{K}(\boldsymbol{D},\boldsymbol{y})$ over dictionary $\boldsymbol{D}'$ by
$$\hat{\boldsymbol{p}}'=\boldsymbol{P}'\boldsymbol{y}$$
where $\boldsymbol{P}'=(\boldsymbol{D}'^T\boldsymbol{D}'+\mu\boldsymbol{I})^{-1}\boldsymbol{D}'^T$.
3. Obtain the regularized residuals
$$r_i=\frac{\|\boldsymbol{y}-\boldsymbol{D}'_i\hat{\boldsymbol{p}}'_i\|_2}{\|\hat{\boldsymbol{p}}'_i\|_2}.$$
where $\hat{\boldsymbol{p}}'_i$ is the coding coefficients associated with class i over $\boldsymbol{P}'$ and $\boldsymbol{D}'_i$ consists of atoms with class label i.
4. Output the identity of $\boldsymbol{y}$ (the class information) as
$$identity(\boldsymbol{y})=\arg\min_i(r_i).$$

III. On the adaptive online Dictionary

This section is to try constructing an adaptive online dictionary for our kernel-based RLS algorithm. First of all, let's define the adaptive online dictionary. Input-adaption refers to the adapting ability of the dictionary when the test sample or the background of sample is slightly changing. For example, the background of the vehicle logos varies from vehicle to vehicle since vehicles have different colors. Moreover, even the same vehicle logo looks different from day to night. If the dictionary stays unchanged in this situation, the recognition performance will drop with no doubt. The online dictionary indicates that the dictionary is updated real-timely after a successful recognition.

Inheriting the previous notation, the dictionary $\boldsymbol{D}$ contains k class with n m-dimensional samples per class. The jth sample belonging to the ith class is denoted by $v_{ij}(1\le i\le k, 1\le j\le n)$. First, we define a correlation-based similarity index Sim_{ij} for atoms of the same class in dictionary. Namely

$$Sim_{ij}=\frac{1}{n-1}\sum_{p=1,p\neq j}^{n}|\boldsymbol{v}_{ij}^T\boldsymbol{v}_{ip}| \tag{12}$$

from which we can know that the index measures the similarity of the atom among the others in the same class, so we only calculate the correlation distance between two atoms in the same class and obtain their average value as the similarity measure.

After each recognition, we obtain a residual vector $\boldsymbol{Res}=[r_1,r_2,\cdots,r_k]$ which is the key to the adaptive online dictionary. Two threshold values Th_{upp} and Th_{low} are built to control the atom updating. The specific flow is as follows.

Dictionary Updating Mechanism

1. Finish the recognition of the test sample $\boldsymbol{y}$. Assume its label output is k.
2. Get the matrix Sim of the dictionary. If an atom in the dictionary is updated, update the related column of Sim
3. Obtain the residual vector from the previous recognition
$$\boldsymbol{Res}=[r_1,r_2,\cdots,r_k]$$
where r_i is the regularized residual.
4. Calculate the ratio Rat of the smallest residual and the second smallest one:
$$Rat=\frac{\min\boldsymbol{Res}}{\min\left(\boldsymbol{Res}-\{\min\boldsymbol{Res}\}\right)}$$
where $\{\boldsymbol{Res}-\{\min\boldsymbol{Res}\}\}$ represent the difference set between $\boldsymbol{Res}$ and $\{\min\boldsymbol{Res}\}$.
5. If the following conditions are both satisfied, go Step 6. If not, do not upate the dictionary.
   Condition 1: Rat is between Th_{low} and Th_{upp}
   Condition 2: $\frac{1}{n}\sum_{p=1}^{n}|\boldsymbol{y}^T\boldsymbol{v}_{ip}|\le\max_j Sim_{kj}$
6. Update the dictionary by using the input atom to replace the atom with the maximum similarity index.

The mechanism of updating the dictionary can better suit the practical recognition scenario because the continuous change of dictionary can better update the dictionary. Our mechanism is just a trial work. It may not be the best way or not even close to the best way to construct an adaptive online dictionary. But it still provides us with a new insight to enhance the VLR performance.

IV. Experiments and Results

A. Experiments of Vehicle Logo Recognition

The experiment is to test the performance of our proposed algorithm for VLR. The vehicle logo database is self-built. A large portion of vehicle logos are automatically collected by a video surveillance camera while others are downloaded from the internet. The vehicle logo database as shown in Fig.2 consists of 1300 images of 21 manufacturers. Some popular manufacturers such as TOYATA, HONDA etc. have more logo samples than the other manufacturers, but each manufacturer has at least 30 logo samples.

The simulation method is similar to [17]. We randomly select 10 vehicle logo images in each class as training samples

Fig. 2. Samples of Vehicle logo database

TABLE I. Comparison of computation time. Set the computation time of l_2 norm RLS algorithm as the unit 1 for better comparison.

Method	Computation Time
Kernel-based l_2 RLS algorithm	1.2031
l_2 RLS algorithm	1.0000
l_1 BP algorithm	1257.2

and the remaining as test samples. All training samples constitute a fixed dictionary. We perform the experiment under different dimensions: 16(4×4), 64(8×8), 256(16×16), 1024(32×32). We also use two different features (Downsample and Eigenface) for comparison. Results are averaged across 20 times experiments for accuracy, which means the training samples are different in each time's experiment.

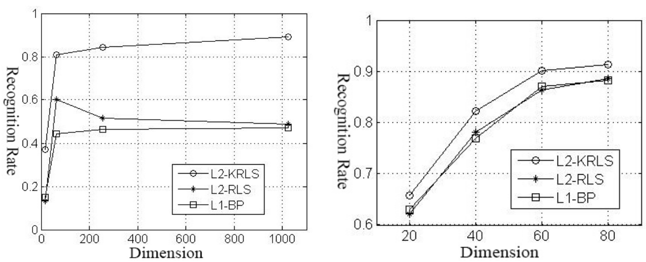

Fig. 3. Comparison of the recognition performance among the proposed kernel-based l_2 norm RLS algorithm, l_2 norm RLS algorithm and the l_1 norm basis pursuit algorithm. (a) The recognition test uses downsample feature. (b) The recognition test uses Eigenface feature. [17].

The recognition performance is shown in Fig.3. Results show our proposed algorithm outperforms l_1 norm basis pursuit (BP) algorithm and l_2 norm RLS algorithm. Next we test the computation time of the three algorithms and results are shown in Table 1. Note that, the computation time of other methods are presented in proportion to the l_2 norm RLS algorithm.

B. Preliminary Simulations of the Adaptive Online Dictionary

The atom updating mechanism of the adaptive online dictionary is tested through a simple simulation. We use three vehicle logo database with different number of samples. There contain 500 samples (10 manufacturers), 900 samples (13 manufacturers) and 1300 samples (21 manufacturers) respectively. We still randomly select 10 samples from each class to construct our dictionary. (For online dictionary, it is set as an initial dictionary) Results in Fig.4 show that the online dictionary works well when the number of samples are small.

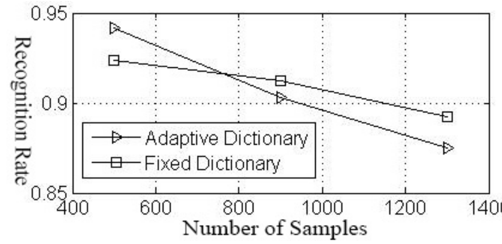

Fig. 4. Comparison of the original dictionary and online adaptive dictionary. All vehicle logos are downsampled to 1024 dimensions. The proposed algorithm is also used in this simulation.

V. Conclusion

In this paper, we elaborated the proposed kernel-based l_2 norm regularized least square algorithm and experimented the algorithm on a self-built database containing 1300 logo samples of 21 manufacturers. Experimental results verified the high recognition accuracy and low computation complexity of the proposed method. Moreover, the mechanism of constructing an adaptive online dictionary was also presented and simulated. Simulations showed the adaptive online dictionary works well under certain conditions.

VI. Acknowledgement

The author Weiyang Liu appreciates the grateful help from Lijia Lu and Fuxing Chen and acknowledges Prof. Zhiheng Zhou for motivating the sparse representation work. This work is supported by Shenzhen Science & Technology Fundamental Research Program (No: JCYJ20130329175141512), and the funded project from Shenzhen HuaZun Technology Co., Ltd.

References

[1] Apostolos P Psyllos, C-NE Anagnostopoulos, and Eleftherios Kayafas, “Vehicle logo recognition using a sift-based enhanced matching scheme,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 11, no. 2, pp. 322–328, 2010.

[2] Kam-Tong Sam and Xiao-Lin Tian, “Vehicle logo recognition using modest adaboost and radial tchebichef moments,” in *International Conference on Machine Learning and Computing (ICMLC 2012)*, 2012.

[3] Apostolos Psyllos, C-N Anagnostopoulos, and Eleftherios Kayafas, “M-sift: A new method for vehicle logo recognition,” in *Vehicular Electronics and Safety (ICVES), 2012 IEEE International Conference on*. IEEE, 2012, pp. 261–266.

[4] Zhang Nan, “Vehicle-logo recognition based on classifier combinations,” in *System Science, Engineering Design and Manufacturing Informatization (ICSEM), 2012 3rd International Conference on*. IEEE, 2012, vol. 2, pp. 278–281.

[5] Shuyuan Yu, Shibao Zheng, Hua Yang, and Longfei Liang, “Vehicle logo recognition based on bag-of-words,” in *Advanced Video and Signal Based Surveillance (AVSS), 2013 10th IEEE International Conference on*. IEEE, 2013, pp. 353–358.

[6] Vladimir S Petrovic and Timothy F Cootes, “Analysis of features for rigid structure vehicle type recognition.,” in *BMVC*, 2004, pp. 1–10.

[7] Hiroki Maemoto, Shigeru Okuma, and Yoshikazu Yano, “Parametric vehicle recognition using knowledge acquisition system,” in *Systems, Man and Cybernetics, 2004 IEEE International Conference on*. IEEE, 2004, vol. 4, pp. 3982–3987.

[8] Yang Liu and Shutao Li, "A vehicle-logo location approach based on edge detection and projection," in *Vehicular Electronics and Safety (ICVES), 2011 IEEE International Conference on*. IEEE, 2011, pp. 165–168.

[9] Xiang Pan, Xiang-Nian Huang, Shao-Hui Zhang, Xiang Li, and Tian-Ai Wu, "Wavelet moment and improved adaboost application to vehicle-logo location," in *Systems, Man and Cybernetics (SMC), 2009 IEEE International Conference on*. IEEE, 2009, pp. 5174–5177.

[10] Wenwen Yang, Guowei Yang, Xu Zheng, and Lubo Zhang, "An improved vehicle-logo localization algorithm based on texture analysis," in *Computer Science and Information Processing (CSIP), 2012 International Conference on*. IEEE, 2012, pp. 648–651.

[11] Songan Mao, Mao Ye, Xue Li, Feng Pang, and Jinglei Zhou, "Rapid vehicle logo region detection based on information theory," *Computers & Electrical Engineering*, vol. 39, no. 3, pp. 863–872, 2013.

[12] Kai Zhou, Karthik Mahesh Varadarajan, Markus Vincze, and Fuqiang Liu, "Hybridization of appearance and symmetry for vehicle-logo localization," in *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*. IEEE, 2012, pp. 1396–1401.

[13] Hyo Jong Lee, "Neural network approach to identify model of vehicles," in *Advances in Neural Networks-ISNN 2006*, pp. 66–72. Springer, 2006.

[14] A Psyllos, CN Anagnostopoulos, and E Kayafas, "Vehicle authentication from digital image measurements," in *16th IMEKO TC4 Symposium*, 2008.

[15] D Zhang, Meng Yang, and Xiangchu Feng, "Sparse representation or collaborative representation: Which helps face recognition?," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 471–478.

[16] Bernhard Scholkopf, "The kernel trick for distances," *Advances in neural information processing systems*, pp. 301–307, 2001.

[17] John Wright, Allen Y Yang, Arvind Ganesh, Shankar S Sastry, and Yi Ma, "Robust face recognition via sparse representation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 2, pp. 210–227, 2009.