Vehicle Logo Retrieval Based on Hough Transform and Deep Learning

Li Huan Qin Yujian Wang Li
College of Electronic Science, National University of Defense Technology
Changsha, Hunan, P. R. China. 410073
lhwx888@126.com 13548599496@163.com wangli08a@126.com

Abstract

Vehicle logo retrieval is an important problem for the intelligent traffic systems, which is still not reliably accurate for practical applications due to the mutable site conditions. In this paper, a new algorithm based on Hough transform and Deep Learning is proposed. The main steps are as follows: First, the logo region is located according to the prior knowledge for the location of vehicle logo and vehicle license plate. Then, typical shapes in vehicle logos, such as circle and ellipse are detected based on optimized Hough transform; meanwhile the accurate position of the logo can be obtained. Finally, the pattern of logo is classified based on Deep Belief Networks (DBNs). Comparative experiments with the actual traffic monitoring images demonstrate that the algorithm outperforms traditional methods in retrieval accuracy and speed. Moreover, the algorithm is particularly suitable for practical application.

1. Introduction

The video surveillance has been widely applied in the intelligent traffic systems all over the world. As an important research field, the process of vehicle logo retrieval consists of several steps: location of the logo region, feature extraction and classification. The vehicle logo location and retrieval are remarkably influenced by the factors such as vehicle's position, image perspective, ambient lighting, shadow, etc.

Based on Hough Transform (HT) and Deep Learning, a new algorithm for vehicle logo retrieval is proposed in this paper. This algorithm combines shape detection and Deep Belief Networks (DBNs) to realize the feature classification. The general framework is shown in Figure 1.

The whole process can be summarized as follows: the typical methods are used to detect and locate vehicle license plates from videos and images, and then the probable region of vehicle logo is roughly located from the relative positions. HT is used to detect some special shapes of logo contour

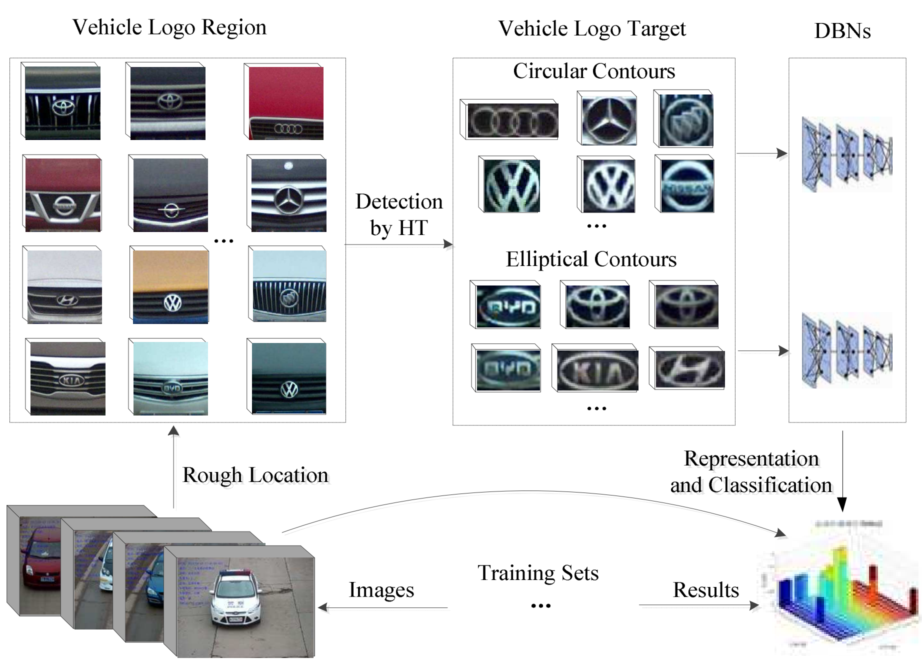

Figure 1: the algorithm framework

which is optimized with geometrical features. Finally, DBNs are designed to represent features of logos and classify them.

The rough region of logos can be located from input videos and images by adopting vehicle license plates detection and prior knowledge about relative positions. Then particular circular and elliptical shapes of logo can be detected by HT, which means the logo is accurately located. Finally, the logo retrieval is achieved based on the classification with DBNs. The contributions of this study are summarized as follows:

(1) The accuracy of traditional methods is comparatively low under the condition of unknown logo type. This paper realizes the location of vehicle logos with similar shapes by particular shape detection method, and the accuracy is relatively higher.

(2) Particular shape detection based on optimized HT achieves both accurate location and rough classification of vehicle logos, and enhances the system efficiency.

(3) The complex function can be represented with less parameter by using multiple hidden layers of DBNs. This model is more effective than traditional neural networks when extracting high-level features from micro image pixels in classification.

2. Related work

A huge amount of prior relative work on vehicle logo detection and retrieval has been done. Wang Yunqiong et al. introduced a method which can locate vehicle logo by two steps, they recognized the logo by template matching [1]; the strategy is useful, but the process of recognition only works for a fixed size, so the robustness is not strong in actual circumstances. SIFT-based enhanced matching scheme is used to recognize the logo, and the correct recognition rate of 10 types can reach higher than 84%. But its shortcoming lies in that the reflection of light in logo region will influence the feature points of SIFT method [2]. HOG features of the logo and the neighborhood of the logo are extracted and then classified by SVM [3]. This method is highly influenced by the accuracy of location. Some methods based on Adaboost are introduced, such as Modest Adaboost in conjunction with radial tchebichef moments [4], and wavelet moment and improved Adaboost [5]. Only two kinds of HD images are mainly tested in such research, and the interference factors in actual application are not discussed.

Recently, deep learning theory has been applied in object recognition and classification. Deep learning refers to a large class of machine learning techniques, and is developing rapidly since 2006. The theory adopts large amounts of nonlinear hierarchical architectures to realize pattern classification and feature recognition [6]. Typical models based on deep learning have been developed such as DBN, Restricted Boltzmann machine (RBM), Deep Boltzmann machine (DBM), DNN as well as related unsupervised learning algorithms such as auto encoders [7] and sparse coding[8]. The above methods have been used to represent high-level features from unlabeled data, making them suitable for classification [9, 10]. Overall, deep learning performs excellently in abstract feature extraction and recognition. Thus it becomes a hot topic for target recognition in computer vision [11~15].

Aiming at logo location, the traditional methods are easily affected by factors such as rotation and light reflection in actual traffic surveillance images and videos. Similarly in recognition, current methods such as template matching, feature points matching, also can’t achieve strong robustness.

Different from the traditional treatment, the commonality of different logos is taken into account in this paper. Different kinds of logos have been divided into several typical shapes. The detection of typical shapes can achieve both logo location and preliminary classification. Further classification for logos with the same typical shape can actualize the retrieval of vehicle logo.

3. Vehicle logo location

3.1 Rough Location

With the efforts of researchers, nowadays the license plate location has excellent effect. Because the position and scale between logo and plate are relatively fixed, firstly we can locate the plate by the existing mature algorithms, and then calculate the approximate region of logo. Furthermore, we can normalize the scale of rough region by the size of plate. The images of logo regions after rough location and normalization are shown in Figure 2.

Figure 2: Examples of rough logo region

3.2 Accurate Location

Generally speaking, plentiful logos have circular or elliptical contours. So the logos which have such contours can be accurately located by detecting these shapes. In this paper, HT and geometrical properties are combined to implement this process.

HT calculates the maximum accumulated local results in a parameter space through voting algorithm. The collection of distinctive forms can be obtained. HT is highly reliable and adaptive to noise, transform, deformity and so on [16]. Therefore, the HT can be used to detect objects with particular shapes.

(1) Circle detection by optimized HT

The basic concept of Hough circle detection is to map the edge pixels from image space to parameter space, then to sum up all coordinate points in parameter space and judge the radius and center. The calculated amount will increase quickly, and require a large amount of storage space. In practice, researchers brought up some improved methods [17~19]. This paper uses the geometrical features and scale prior knowledge to realize fast process.

A circle has a geometrical feature that perpendicular bisectors of two strings will meet at the center, as shown in Figure 3.

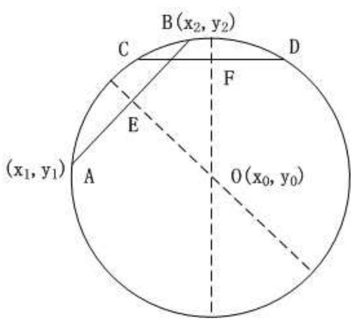

Figure 3: One geometrical feature of circle

Points (x_1, y_1) and (x_2, y_2) are two endpoints of a string, r_{max} is the upper limit of radius, they must obey:

$$|x_1 - x_2| \le 2r_{\max}, |y_1 - y_2| \le 2r_{\max} \tag{1}$$

$$(x_1 - x_2)^2 + (y_1 - y_2)^2 \le 4r_{\max}^2 \tag{2}$$

Then the perpendicular bisector of this segment can be expressed as:

$$y = \frac{x_1 - x_2}{y_1 - y_2} x + \frac{x_1^2 + y_1^2 - x_2^2 - y_2^2}{2(y_1 - y_2)} \tag{3}$$

Besides, the points in this perpendicular bisector must satisfy:

$$r_{\min}^2 \le (x - x_1)^2 + (y - y_1)^2 \le r_{\max}^2 \tag{4}$$

Firstly, according to the prior knowledge, the radius of a circular can be limited to the scale of $(r_{\min}, r_{\max})$, then in order to improve the detecting speed and accuracy, we use the method proposed in [20] to filter the length of a curve: Preset the threshold of curve length as $Th_{\min}$, which is sextant of circumference with the radius $r_{\min}$. If a curve's length is longer than $Th_{\min}$, the curve is set as valid. The steps are introduced as follows:

Step1: Search continuous outline curves, mark the positions of curves, and add the curves which have eligible length into collection $Q = (Q_1, Q_2, \cdots, Q_n)$;

Step2: For curve $Q_i (1 \le i \le n)$, mark the strings of which length are at the scale of ($Th_{\min}$, $r_{\max}$). Then calculate the possible centers and their probabilities, and obtain the center collection $C = (C_1, C_2, \cdots, C_t)$, $C_j = (x_j, y_j, r_j), 1 \le j \le t$;

Step3: Calculate the similarity of circles in C and merge them. Then define $O_i = \max(C)$, which has the maximum probability;

Step4: if $i = n$, then $O = (O_1, O_2, \cdots, O_n)$, take it into step3. Then $O_{\max}$ is the circle detected.

(2) Ellipse detection by optimized HT

The parameter equation of any ellipse can be expressed as below:

$$\frac{[(x-p)\cos\theta + (y-q)\sin\theta]^2}{a^2} + \frac{[(y-q)\cos\theta + (x-p)\sin\theta]^2}{b^2} = 1 \tag{5}$$

Here (p, q) is the center coordinate, (a, b) are two axes, and θ is the angle between the long axis and horizontal axis. Compared with circle, the ellipse has four unknown parameters, the amount of calculation is much huger by traditional HT. Consider that ellipse has the following geometric property: if the tangents of two points on the edge of an ellipse have the same direction, then the middle point is the center of the ellipse. Thus, we can find the point pairs whose tangents have the same slope at first. The middle point is taken as center. The detection process is introduced as follows:

Step1: Initialize the 2D accumulation matrix $E(w, h) = 0$, w, h mean the size of logo region C;

Step2: Detect the edge of logo region, and change it to binary image V, where the background is set to 0, and edge is set to 1;

Step3: Calculate the tangent directions of every non zero pixel in V;

Step4: Iterate the pixel pairs with the same tangent direction in V, calculate the coordinate (x, y) of middle point, and add 1 to $E(x, y)$;

Step5: In the accumulation matrix, the coordinate of the largest element is the center (p, q) of the ellipse.

Add the center coordinate (p, q) into elliptic equation (5). Select data from edge array V, then the parameters a, b, θ can be counted by HT in 3D space.

With the method above, we can detect and accurately locate the logos with circular or elliptical shapes. As a special case in circular logos, if there are four circulars detected in the logo region, the brand can be judged as Audi directly. The detection results are shown in Figure 4.

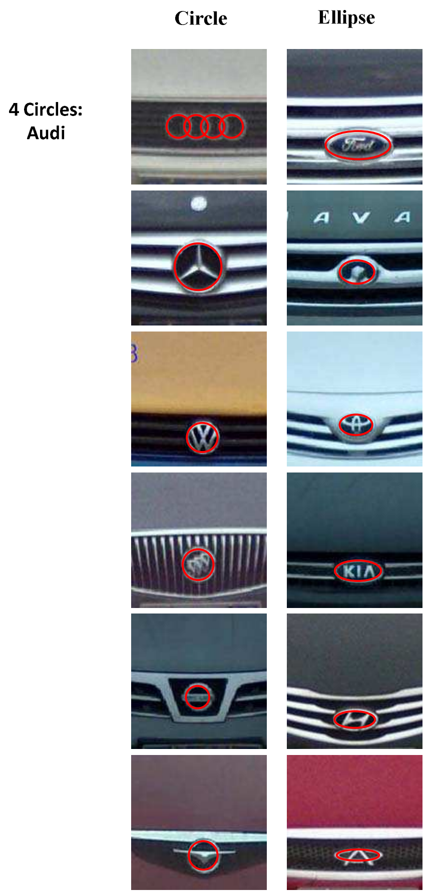

Figure 4: Accurately location of logo and preliminary classification (part of the brands)

4. Vehicle logo retrieval based on DBNs

DBN is a kind of random deep neural network，which can be used for statistical modeling. It can represent abstract features and statistical distribution of a target. The network structure of DBNs is stacked by multiple RBMs.

4.1 RBMs

Supposing that every node in a network has the value of 0 or 1 randomly, and the full probability distribution $p(v,h)$ meets the Boltzmann distribution, then a RBM model is set up. RBM is a typical random neural network, and it consists of three parts:

(1)Visible units

$v=[v_1,v_2,\cdots,v_n]$, which is the input of network, such as pixels of images, voice data, etc.

(2)Hidden units

$h=[h_1,h_2,\cdots,h_m]$， abstract features or categories gained by learning.

(3)Bias unit

Bias unit is always in active state，and is used to adjust every hidden unit.

The neurons of a RBM generate the status and outputs by probability. Their connections have limits and there aren't any connections between visible units and hidden units. Suppose that the input is an image, and then a RBM can be shown in Figure 5.

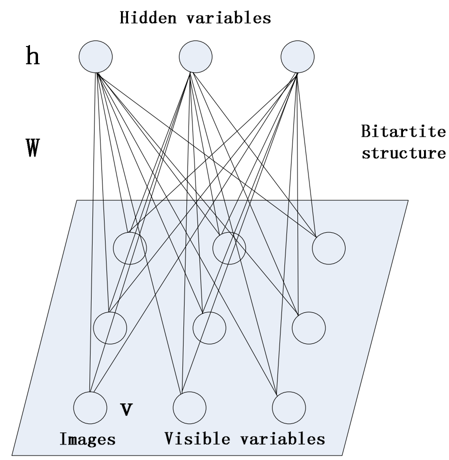

Figure 5: Structure of RBMs

From the probability model of energy theory, the conditional probability distribution of visible unit v and hidden unit h can be defined as

$$P(h/v)=\frac{e^{-Energy(v,h)}}{\sum_{y} e^{-Energy(v,h)}} \tag{6}$$

And the joint probability distribution is：

$$P(v,h)=\frac{e^{-Energy(v,h)}}{Z} \tag{7}$$

The definition of energy function is：

$$Energy(v,h)=-b'v-c'h-h'Wv \tag{8}$$

In which b is the offset of v, c is the offset of h, W is the weight matrix.

Taking Equation (8) into (6)， the $P(h/v)$ can be calculated as:

$$\begin{aligned} &P(h/v)\\ &=\frac{exp(b'v+c'h+h'Wv)}{\sum_{\tilde{h}} exp(b'v+c'\tilde{h}+\tilde{h}'Wv)}\\ &=\frac{\prod_i exp(c_i h_i+h_i W_i v)}{\prod_i \sum_{\tilde{h}_i} exp(c_i\tilde{h}_i+\tilde{h}_i W_i v)}\\ &=\prod_i \frac{exp(h_i(c_i+W_i v))}{\sum_{\tilde{h}_i} exp(\tilde{h}_i(c_i+W_i v))}\\ &=\prod_i P(h_i/v) \end{aligned} \tag{9}$$

Suppose that θ is the parameter set of the energy model. The training process is to change θ at the trend of reducing the energy gradient, and to achieve the lowest potential energy finally. That means for the derivative of θ:

$$\frac{\partial \log P(v)}{\partial \theta} = -\sum_{h} P(h/v)\frac{\partial Energy(v,h)}{\partial \theta} + \sum_{v,h} P(v,h)\frac{\partial Energy(v,h)}{\partial \theta} \quad (10)$$

Taking the concrete value of v and h into (6)～(9)，then (10) can be solved. With the data of images, Gibbs sampling can be used for training RBM[21,22], the method is introduced as below：

Algorithm 1: Updating RBM

Input:

v1: Sample from the training distributions for the RBM

e: Learning rate for the stochastic gradient descent in contrastive divergence

Output:

W: The weight matrix of RBM

b : The biases vector for hidden units of RBM

c : The biases vector for input units of RBM

Initialization: Randomly choose W, b and c

for all hidden units i, do

  compute $Q(h_{1i}=1|v_1)$

  sample h_{1i} from $Q(h_{1i}=1|v_1)$

end for

for all visible units j, do

  compute $P(h_{2i}=1|h_1)$

  sample v_{2j} from $P(h_{2i}=1|h_1)$

end for

for all hidden units i, do

  compute $Q(h_{2i}=1|v_2)$

end for

$W \leftarrow W + e(\mathrm{h}_1 v_1' - \mathrm{Q}(\mathrm{h}_{2.} = 1 | v_2) v_2')$

$b \leftarrow b + e(\mathrm{h}_1 - \mathrm{Q}(\mathrm{h}_{2.} = 1 | v_2))$

$c \leftarrow c + e(v_1 - v_2)$

4.2 DBNs

The key problem of stacking multiple RBMs to DBNs is how to train and construct the deep network. The learning method can be divided into two stages: the unsupervised learning and Fine-tune. During unsupervised learning, every RBM sub-network is trained by Algorithm 1. The output of the previous layer is sent to the next layer as input. In order to strengthen the modeling ability of a single target, certain discrimination training is needed. For example, the common features of several logo images of the same brand under different illumination conditions can be extracted in the unsupervised learning. However, the special features need to be extracted at the Fine-tune stage. Only if the distinctiveness of individual logo has been learned, the network structure can have better modeling and classification ability.

The training process of DBNs is introduced as follows:

step1: Train the first layer of RBMs, taking the initial observation samples as input data v;

step2: After the training of the first layer, the abstract presentation of original data can be obtained, which is the output of the first layer;

step3: Make the output of the first layer as the new observation v_1, and use it to train the second layer of the RBMs. Similarly, all the layers can be trained in order.

step4: Fine-tune: Conduct supervised training of all parameters of the DBNs.

Step1~step3 can be realized by the method introduced in section 4.1. As to step4，how to attain the best Fine-tune learning is one of the research emphasizes in deep learning.

Deep learning can be interpreted as calculating a function of $Y = F(Z,W)$, whose dependent variables are input samples and parameter sets. Here Z is the input sample, while W is the parameter set of the system or model. In pattern recognition, the output Y is a pattern of samples, and it can be named as tabs, or probability value which is related to patterns. The loss function $E = ER(D,F(Z,W))$ is defined as the difference between the actual output D and the calculation result of function F. The mean loss function $E_{average}$ is the average of all Es produced by every training collection $[(Z_1,D_1),(Z_2,D_2),\cdots,(Z_p,D_p)]$. The best Fine-tune learning question can be concluded as how to find the proper W, through which the minimum of $E_{average}$ can be achieved.

In this paper, Fine-tune learning is carried out by gradient [23]. Every kind of samples is singed to be supervised training at the base of gradient descending. If $Eaverage$ is a derivable function, the changing of W can be provided as

$$W_k = W_{k-1} - e\frac{\partial E(W)}{\partial W} \quad (11)$$

In this method, W is continuously changed to find the minimum gradient direction of $E_{average}$, and finally W can achieve the minimum value.

After the process above, the DBNs designed in this paper for logo recognition can produce the classification result of different logos.

5. Experimental results and analysis

The new algorithm has been tested on a PC which has an Intel Core-i5 CPU with 3.33 GHz and 8GB DDR3 memory, using 80000 surveillance image samples captured from intelligent transportation system. The image samples contain 10 kinds of automobile brands. For comparison, we also employed three other methods: SIFT-based feature-matching [2], HOG features by SVM [3], and AdaBoost [4]. The operation effect and processing time for each method are recorded and analyzed.

(1) Dataset

To study the vehicle logo retrieval in intelligent transportation, we construct the dataset named

"VAP-RoadFig". The source of the dataset is real natural pictures captured from hundreds of monitors in a Chinese city. The pictures are stored as jpg files with the size as 1360 ×1024 and the color depth of 24 degrees. The content of these images are scenes of vehicles, which show plates, logos and figures clearly. Since the pictures are wild images of actual monitor scenes, the dataset have abundant external conditions of lighting, angle, distance, etc. The dataset includes 30 kinds of different vehicle brands with the total number more than 200000, and it contains enormous number of diversified vehicle images.

(2)Operation effect

For experiment, we choose 80000 pictures of 10 kinds of logos with circular or elliptical shapes. We randomly choose half of them per brand for training and the rest 40000 images for testing. We perform vehicle logo retrieval tests from image samples according to the four algorithms mentioned above, and count the accuracy rate (%) for each method as shown in Table 1.

Table 1: Retrieval Effects

	New algorithm	Sift Match	HOG + SVM	AdaBoost
Audi	93.3	90.2	80.7	85.3
Benz	85.8	88.4	85.6	83.2
Buick	88.1	85.5	83.9	70.2
BYD	89.0	86.3	82.8	81.6
Great Wall	92.6	82.3	85.8	85.2
Volkswagen	95.5	88.7	90.3	88.6
Haima	93.1	85.2	86.7	86.6
Nissan	85.4	80.3	75.8	76.4
Toyota	96.6	91.4	89.3	85.1
Ford	90.5	83.6	82.4	79.4

From Table 1 we find that, for the whole impact, all of the four algorithms can achieve the accuracy better than 75%. Generally speaking, the logos which have distinct characteristics are striking to background and they can be retrieved correctly at higher level, such as Audi, Volkswagen, Toyota, etc. The texture of some logos is simple and similar to the color of the background. For these logos, the accuracy rate falls down, such as Buick, Nissan, etc. The algorithm of this paper has higher accuracy.

(3)Processing time

We deal with 40000 images in these four methods. The processing time for each is shown in Figure 6. It can be found that, if the brand of a possible automobile is unknown, the contrastive three algorithms need to search logo from the total graph. So they require more time, especially the Sift Match, which costs the most time as the calculation amount to extract SIFT key points is huge. The new algorithm presented in this paper can shrink the region of image classification at the fastest processing speed.

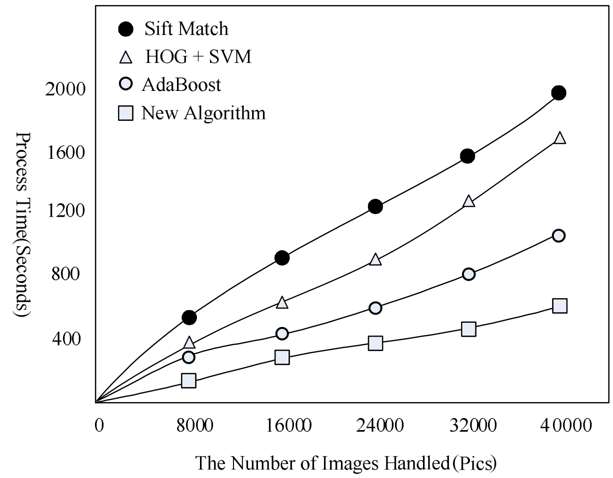

Figure 6: The processing time of several algorithms

6. Conclusion

In order to solve the major problems in vehicle logo retrieval including location and classification, a framework consisting of HT and Deep Learning is proposed. First the rough region of logo is located; then circular and elliptical shapes in the region are searched by HT; finally DBNs are separately trained for the two kinds of shapes, while logos are classified by DBNs. The new algorithm resolves the problems of logo location by comprehensive search, and optimizes the retrieval effect of logos which have typical shapes. With this method, not only the accuracy of vehicle logo retrieval could be improved but also the system expense could be reduced, and the whole efficiency could be strengthened. For future work, we plan to expand the types of typical logo shapes, and optimize the speed of HT further.

References

[1] Wang Yunqiong, Liu Zhifang, Xiao Fei. A Fast Coarse-to-Fine Vehicle Logo Detection and Recognition Method. Proceedings of the 2007 IEEE International Conference on Robotics and Biomimetics, 691-696, 2007.

[2] Apostolos P. Psyllos, Christos-Nikolaos E. Anagnostopoulos, and Eleftherios Kayafas. Vehicle Logo Recognition Using a SIFT-Based Enhanced Matching Scheme. IEEE Transactions on Intelligent Transportation Systems. 11(2), 322-328, 2010.

[3] D. F. Llorca, R. Arroyo, M. A. Sotelo. Vehicle logo recognition in traffic images using HOG features and SVM. Proceedings of the 16th International IEEE Annual Conference on Intelligent Transportation Systems, 2229-2234, 2013.

[4] Kam-Tong Sam, Xiao-Lin Tian. Vehicle Logo Recognition Using Modest AdaBoost and Radial Tchebichef Moments. Proceedings of 4th International Conference on Machine Learning and Computing, 25: 91-95, 2012.

[5] Xiang Pan, Xiang-Nian Huang, Shao-Hui Zhang. Wavelet Moment and Improved Adaboost Application to Vehicle-logo Location. Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics, 5174-5177, 2009.
[6] Li Deng, Three Classes of Deep Learning Architectures and Their Applications: A Tutorial Survey. Microsoft Research, 2013.
[7] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle. Greedy layer-wise training of deep networks. In NIPS, 2007.
[8] H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient sparse coding algorithms. In NIPS, 2007.
[9] M. Zeiler, D. Krishnan, G. Taylor, and R. Fergus. Deconvolutional networks. In CVPR, 2010.
[10] J. Yang, K. Yu, Y. Gong, and T. S. Huang. Linear spatial pyramid matching using sparse coding for image classification. In CVPR, 2009.
[11] Yu, K., Xu, W., Gong, Y.: Deep learning with kernel regularization for recognition. In: Neural Information Processing Systems, In NIPS , 2009
[12] Ranzato, M., Huang, F., Boureau, Y., & LeCun, Y. Unsupervised learning of invariant feature hierarchies with applications to object recognition. In CVPR 2007.
[13] Y.-L. Boureau, F. R. Bach, Y. LeCun, and J. Ponce. Learning mid-level features for recognition. In CVPR, 2010.
[14] M. Ranzato, J. Susskind, V. Mnih, and G. Hinton. On deep generative models with applications to recognition. In CVPR, 2011.
[15] J. Susskind, G. Hinton, R. Memisevic, and M. Pollefeys. Modeling the joint density of two images under a variety of transformations. In CVPR, 2011.
[16] Chi-Han Chung, Shyi-ChyiCheng, Chin-ChunChang. Pattern Recognition, 243: 3219-3232, 2010.
[17] Rozenn Dahyot. Statistical Hough Transform. IEEE Trans on Pattern Analysis and Machine Intelligence, 31(8): 1502-1509, 2009.
[18] N. Aggarwal and W.C. Karl. Line Detection in Image through Regularized Hough Transform [J]. IEEE Trans. on Image Processing, 15(3): 582-591, 2006.
[19] A. Bandera, J.P.B.J.M. Pe?rez-Lorenzo, and F. Sandoval. Mean Shift Based Clustering of Hough Domain for Fast Line Segment Detection. Pattern Recognition Letters, 27: 578-586, 2006.
[20] Lin C, Face detection in complicated backgrounds and different illumination conditions by using YCbCr color space and neural network, Pattern Recognition Letters, 28 (16): 2190-2200, 2007.
[21] YoshuaBengio. Learning deep architectures for AI. Foundations and Trends in Machine Learning. 1(2): 1-127.
[22] Stuart Geman, Donald Geman. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images, 6(6): 721-741, 1984.
[23] Léon Bottou. Large-Scale Machine Learning with Stochastic Gradient Descent In Proceedings of the 19th International Conference on Computational Statistics, 177-187, 2010.