Front Vehicle Detection Method based on Edge Boxes and Adaboost

Peng Gan [1, 2, *], Yu Xiao [1, 2]

[1]School of Electronics and Information Engineering, Tianjin Polytechnic University, Tianjin 300387, China

[2]Tianjin Key Laboratory of Optoelectronic Detection Technology and Systems, Tianjin, 300387, China

* Corresponding author Email: 915342086@qq.ocm.

Abstract

Vehicle detection for streaming video data with complex scenes is an interesting but challenging task for Advanced Driver Assistance Systems. In this paper, we present a faster vehicle detection method, called Edge Boxes and AdaBoost for vehicle detection (EBAV), which effectively combines Edge Boxes and AdaBoost. First, two hundred proposal windows are obtained by Edge Boxes method according to the edge feature. Second, the camera calibration technique and the vehicle color information are used to remove windows which do not contain the vehicle. Third, the normalized cut algorithm is adopted to group the proposal windows after filter operation into various clusters. Finally, the proposal windows of each cluster are ranked according to the predicted objectness scores, and then a few top scored windows of each cluster are selected and fed into the AdaBoost classifiers for judgment. Since there are no publicly related dataset, we collect the front vehicle video as the train and test dataset. Experiments with traffic images in different weather conditions verify the practicability of the proposed method. Comparison with related method in detection speed verifies the superiority of the proposed method.

Keywords

Vehicle detection, Edge Boxes, Proposal windows, AdaBoost classifiers.

1. Introduction

Vehicle detection technology is a hot research topic in intelligent traffic image processing and has been widely used in the field of Intelligent Traffic System (ITS), such as Driver Assistance Systems [1], Traffic Information System and Vehicle Automatic Control System. In addition, the increasing demand on acquiring additional information of vehicles (such as location and size) in intelligent transportation systems drives more attentions on video-based vehicle detection strategy. Vehicle detection methods based on computer vision can be roughly divided into two classes: feature-based and machine learning method [7]. The feature-based method locates vehicles by detecting the local features, such as the symmetric components (wheels, head-lamp, rear-lamp) [3], shadow [2], edge [4]and so on. The advantage of this method is instantaneity. However, the extraction of these features is usually relied on the edge detection, so the accuracy and robustness of the feature-based method are not good. The method referred in [2] locate vehicles by detecting the shadow, which is easily influenced by the illumination and weather. And the method described in [20], which detect vehicles by lamps, would also been interfered by streetlights and city lights in night scene, thereby interfering the detection result. Computational complexity and time consumption of edge-based vehicle detection method [4] is low, which is suitable for the real-time applications. However, false detection is easily caused by noise and background edges in edge map.

Compared with above methods, the method of machine learning possesses advantages in robustness and accuracy. Most of them consist of features and classifiers, Support Vector Machine (SVM) [6], AdaBoost [5] and Neural Network (NN) are usually applied to train popular features such as HOG [12] and Haar-like wavelet [8]. The strategy based on Haar features and the cascade classifier for vehicle detection systems has captured growing attention for its effectiveness and robustness; however, such a vehicle detection strategy relies on exhaustive scanning of an entire image with different sizes sliding windows, which is tedious and inefficient, since a vehicle only occupies a small part of the whole scene. If we first utilize object proposal method to find some candidate locations that may contain objects, in this stage many locations may be false positives can be filtered, and then the AdaBoost classifiers only carried on these selected candidate locations, by this way, the detection speed can be improved.

In this work, we propose a new method to solve these problems based on Edge Boxes and AdaBoost. The proposed method is includes three major steps. The first part extracts the proposal windows by Edge Boxes algorithm. In order to improve the detection speed, the filtering operation of proposal windows is necessary. So the second part performs the filter operation on these proposal windows to filter the nonvehicle windows. The normalized cut algorithm [17] groups the proposal windows into a few clusters to reduce the number of proposal windows for AdaBoost classification. The third part selects a few top scored windows of each cluster for vehicle detection. During the test phase, the method of AdaBoost, BING+AdaBoost for vehicle (BIAV), and EBAV are tested on the dataset respectively. Experimental results show that the EBAV is superior to other methods in detection accuracy and detection speed.

The rest of this paper is organized as follows. Section 2 reviews the related works. Section 3 introduces the process of EBAV in detail. In order to verify the effectiveness and superiority of the proposed method, Section 4 provides experiments and validation, shows detection results of the proposed method detecting in the traffic images under different weather conditions, and compares the detection results with the other methods. Section 5 offers some concluding remarks and discussions of future works.

2. Related Work

2.1 Haar features and Cascade of Classifiers

The Haar features, which can be viewed as real-time feature for object detection, are adopted in our method for its simple calculation process. It is a simple rectangular feature, and has faster extraction speed. During the training process, all computable features are extracted from which the AdaBoost algorithm can select features by means of the lowest weighted classification error. Fig. 1 shows the basic two, three, and four rectangle template sets employed as Haar filters.

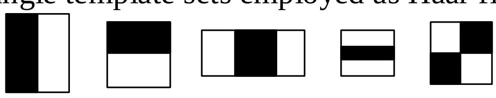


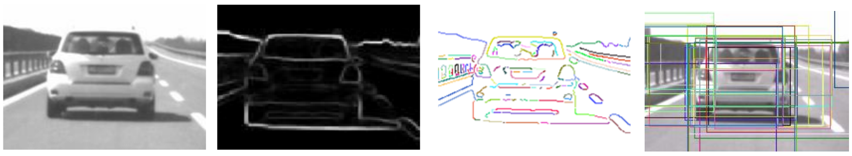
Fig. 1 Basic two, three, and four rectangle template sets used for Haar filters

The selected features are combined as simple weak classifiers to a strong classifier *H*. A few of these strong classifiers are combined in a cascaded structure, where sub-windows that do not contain vehicle-like objects are rejected very quickly using just a few features.

By testing on the dataset, it is found that the AdaBoost algorithm based on Haar feature is very robust. But the AdaBoost algorithm is pretty complex and utilizing a sliding window scheme which results in slow detection speed, and has high false detection rate a when the road scene is complex or the light is weak.

2.2 Edge Boxes

This method generates object proposals based on edges. The main idea is the number of contours that are fully contained in an arbitrary size box is indicative of the probability of an object in the box. It obtains the initial edge map by using the fast and publicly available Structured Edge detector [18], where each pixel has an edge magnitude and orientation. For the efficiency, adjacent edge pixels of similar orientation are clustered together to form a group. It computes the affinities between the edge groups based on their relative positions and orientations such that groups that have high affinity forming long continuous contours. It computes the score of a bounding box by summing the edge strength of all edge groups within the box, then minus the strength of edge groups that are part of a contour that straddles the boundary of the box. It adopts a sliding window paradigm to evaluate proposals in every scale and position and generates a score which denoting the likelihood of an object exists in a sub-window of the image. Then, it finds the top proposals from millions of possible candidates. The results of the Edge Boxes are as follows:



(a) (b) (c) (d)

Fig. 2 Illustrative examples showing from left to right (a) original image, (b) Structured Edges, (c) edge groups, (d) object proposals

2.3 BING

Binarized Normed Gradients for the possibility of object generates the detection proposals by using the idea that generic object with well-defined boundary looks alike strongly when we observe the norm of the gradient, after resizing the image windows to a small fixed size. Hence, BING resizes a sub-window which includes an object to $8*8$ and learns a generic possibility of the object using the norm of the gradients, a simple 64D feature vector, in a cascaded SVM [21](support vector machine) framework. It predefines quantized window sizes $\{(w_0, h_0)\}$, where $w_0, h_0 \in \{10,\ 20,\ 40,\ 80,\ 160,\ 320\}$, the image is resized to all of these 36-size, then it employs a $8*8$ size window to scan over these 36-size images and extracting NG (Normed Gradients) features from them for detection.

For the object proposal methods, the document [16] shows that Edge Boxes [14] and BING [15] have advantages in speed and accuracy. Given just 1,000 proposals, Edge Boxes achieves over 96% detection rate and about 87% recall at overlap threshold of 0.5. Particularly worth mentioning is that Cheng adopts the model binary approximation method and a series of optimized mechanisms make the speed achieves 300 frames per second on CPU, which is very fast.

3. Edge Boxes+AdaBoost Method for Vehicle Detection

In the proposed method, the Edge Boxes method is used to get 200 proposal windows as the candidate windows B_n. If all proposal windows are entered into the AdaBoost classifiers for vehicle detection, the computation is burden and the speed is slow. Therefore, it is necessary to filter out part of the nonvehicle proposal windows in the premise of ensuring the vehicle windows are not be deleted.

3.1 Proposal window filter

The camera calibration technology and the vehicle color information are used to remove proposal windows which do not contain the vehicle. The camera is calibrated by the method [22], and the actual distance of a pixel in the image can be obtained. So we can estimate the width and height range of each row's vehicle window in the image, and the corresponding relationship between the size of vehicle windows and the Y axis of image axes can be established. Fig. 3 shows that the size of proposal windows is different when the vehicle located different rows of image. The greater the vehicle distance, the smaller the size of vehicle window, on the contrary, the smaller the vehicle distance, the larger the size of vehicle window. According to the corresponding relationship between the size of vehicle windows and the Y axis of image axis, the partial nonvehicle windows are filtered from the B_n. The result of filter operation is B'_n and the proposal window filtering formula is as follows:

$$B'_i = \begin{cases} B_i & if\ T_{rW_{min}} < W_i < T_{rW_{max}},\ T_{rH_{min}} < H_i < T_{rH_{max}} \\ 0 & others \end{cases} \tag{1}$$

$$i = 1, 2, 3, ..., n$$

Fig. 3 Calibration information is used to estimate the size of each row's vehicle windows, the Fig. 3 example showing the proposal window's size of different rows in the image.

Where, $T_{rW_{min}}, T_{rW_{max}}, T_{rH_{min}}, T_{rH_{max}}$ are the minimum and maximum values of proposal window's width, the minimum and maximum values of proposal window's height(the Y axis coordinate value is r), W_i, H_i are width, height of proposal window. Using the calibration information to filter the proposal windows can reduce the number of windows, while ensuring that the vehicle windows will not be missed. The result of this process is show in Fig. 4.

(a) (b)

Fig. 4 (a) The original results of proposal windows (b) the filter results of proposal windows based on camera calibration information.

The color feature is an important feature of vehicle, vehicle area color is uniform and the quantity of vehicle color category is limited. According to the color information, the proposal windows B_n'' can be obtained. Fig 5 (a) is the filter result of the proposal windows B_n' based on camera calibration information, and Fig 5 (b) is the filter result of the proposal windows B_n'' based on color information.

(a) (b)

Fig. 5 Using color information to filter the proposal windows

3.2 Vehicle classification for the proposal window

Because of containing the same objects, proposal windows has a similarity on the location and size information. If we subdivide the proposal windows B_n'' into various clusters, and classify a part of windows selected from each cluster using AdaBoost classifiers, the calculation of subsequent classification can be reduced. In the proposed method, the normalized cut method is adopted to group the proposal windows B_n'' into m clusters $(m=10)$, and the k top scored windows of each cluster are selected and fed into AdaBoost classifiers. The clustering result is show in Fig. 6 (b).

(a) (b)

Fig. 6 Clustering examples showing from left to right (a) the filter results of proposal windows by calibration information and color information, (b) proposal windows generated by Edge Boxes. Different colors indicate different clusters, which are produced by the normalized cut method.

As described in Sect. 2.1, the AdaBoost algorithm gets the vehicle area by sliding window with different sizes traversing the whole image, which results in a large amount of calculations and poor detection speed. Because the $m*k$ proposal windows obtained by the clustering algorithm contain all of vehicle regions in the image, so we calculate the Haar feature value of each proposal window and feed it into a cascade of strong classifiers, FB_n' are the results of the cascade classifiers. The process of vehicle classification for proposal windows is show in Fig. 7.

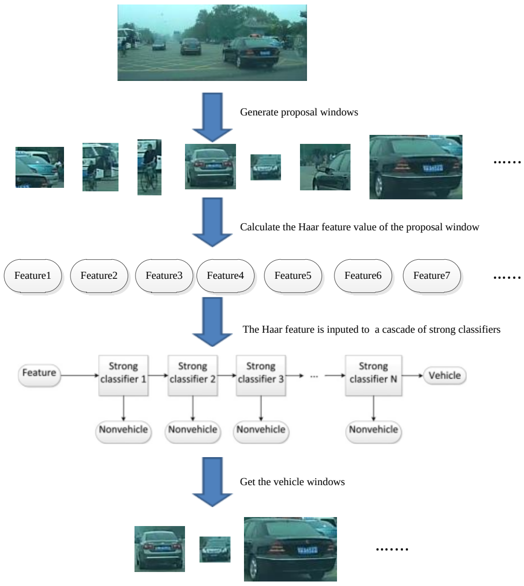


Fig. 7: Vehicle classification for the proposal windows

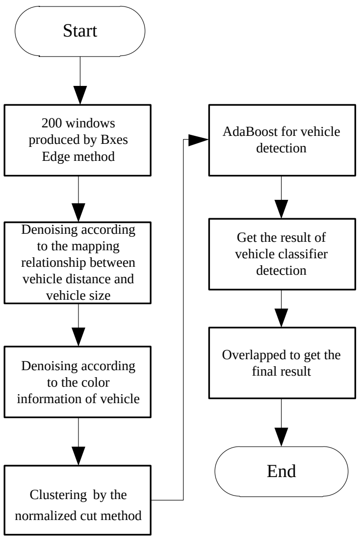


Fig. 8 The steps of using the EBAV to detect the vehicle

We experimentally vary $k = 1,2,3,4,5...n$ to vehicle detection, when $k < 20$, $k = 5$ is work best for balancing detection rate and detection speed, and when $k > 20$ the number of proposal windows has a great influence on the detection speed. So $5*m$ proposal windows are extracted from each image. The result FB'_n of AdaBoost classifiers is eliminated the overlapping to get the final result FB''_n by using the center distance and overlapping area information between adjacent windows. The formula of eliminate the overlapping is as follows:

$$FB''_i = \begin{cases} FB'_{i+1} & S_{i+1} > TS_i, d < d_T \\ FB'_i & others \end{cases} \qquad (2)$$
$$i = 1, 2, ..., n$$

Where, S_i, S_{i+1} are the area of adjacent windows, d, d_T are center distance and center distance threshold of adjacent windows, $T = 0.75, d_T = 5$(the best threshold value is obtained through many experiments).

The active learning method [10]is used to train AdaBoost classifiers, the active learning involves two steps. In the first step which is called passive learning, manually annotated positive samples and random negative samples are used to train the classifiers. In the second step, true and false positive samples are manually queried from the outputs generated by applying the passively trained classifiers on test images. The method of active learning not only reduces the false detection rate but also keeps higher detection rate in the vehicle detection.

During the proposed method, Edge Boxes method is used to preprocess the image to get a part of proposal windows that may contain objects, which remove most of the background regions in the image. According to filter operation and the clustering algorithm, the number of proposal windows can be reduced. Compared with the AdaBoost algorithm, the proposed method improved the detection speed significantly. Fig. 8 is the flow chart of the proposed method:

4. Experimental Results

In order to verify the performance of the proposed method, the method of AdaBoost, BIAV, and EBAV are tested on the dataset respectively. In the experiment, the road video captured by industrial camera and driving recorder is taken as the test dataset, and the video data is collected in a variety of weather and road environments. The detection recall (Recall) and detection accuracy (Precision) is used to measure the performance of the proposed method. Recall shows the proportion of the number of detected vehicles in the actually total number of vehicles, the higher the value, the more vehicles are detected. Precision shows the ability of the method to overcome the effects due to background clutter, high Precision means that wrong detection rarely happened. Recall and Precision are defined as follows:

$$\mathrm{Re}\,call = \frac{true\ positive}{true\ positive + false\ negative} \tag{3}$$

$$\mathrm{Pr}\,ecision = \frac{true\ positive}{true\ positive + fasle\ positive} \tag{4}$$

Where, true positive is the number of correctly detected vehicles, false negative is the number of missed vehicles, false positive is the number of the detected rectangle not being a vehicle.

The comparison of recall and precision under different weather conditions (sunny, cloudy, night) is show in Fig. 9. The Fig. 9 show that EBAV method's recall and precision is increase with the increase in the number of proposal windows, and it is better than other methods. Edge Boxes algorithms is used to filter the background area in the image, it reduces the false detection caused by background and improves the detection accuracy. The accuracy of Edge Boxes method is better than BING method and mentioned in the document [14], the Fig. 9 proved the conclusion. The PR curve (precision-recall curve) of the three methods in the test dataset is show in Fig. 10, the PR curve of EBAV method is better than AdaBoost method and BIAV method. Fig. 11 shows the results of vehicle detection under different weather conditions (sunny, cloudy, night) and different road environments (urban expressway, highway, city street road).

The experimental results show that the proposed method has a good robustness and accuracy in different road conditions and weather environments. Because the edge features of object in different roads and environments not being lost, Edge Boxes method can always get proposal windows that may contain objects, so the proposed method has a good robustness. The method of active learning improved the accuracy of classifier model.

Now, the running speed of the three methods are tested and compared as followed. The number of each test video is 5000 frames, and the average time $\overline{t}$ is used as the detection time of each method. Table1 and Table2 are the detection time of three methods.

Table 1. The detection time of three methods in video data collected by industrial camera (unit: ms/frame).

TestData	AdaBoost	BIAV	EBAV
TestData1	65.8	27.2	39.2
TestData2	62.4	28.7	38.3
TestData3	75.3	29.0	40.4
TestData4	63.2	27.3	44.5
TestData5	64.5	26.8	43.7
TestData6	74.1	28.9	45.2
TestData7	76.6	29.1	41.1
TestData8	75.7	28.6	39.6
TestData9	80.4	28.2	45.2
Average time($\bar{t}$)	70.9	28.2	41.9

Table 2. The detection time of three methods in video data collected by driving recorder (unit: ms/frame).

TestData	AdaBoost	BIAV	EBAV
TestData10	67.4	28.3	39.3
TestData11	70.6	28.7	41.1
TestData12	72.0	29.1	43.0
TestData13	70.2	27.6	40.7
TestData14	71.6	28.8	45.1
TestData15	71.9	28.7	46.5
TestData16	80.3	30.2	38.7
TestData17	79.8	30.8	44.9
TestData18	79.8	31.4	46.6
Average time($\bar{t}$)	73	29.3	42.9

TestData1-TestData3,TestData10-TestData12 are the sunny urban expressway traffic video, sunny highway traffic video and sunny city street traffic video collected by industrial camera and driving recorder respectively. Test Data4-Test Data6, Test Data 13-Test Data 15 are the cloudy urban expressway traffic video, cloudy highway traffic video, cloudy city street traffic video that collected by industrial camera and driving recorder respectively. Test Data7-TestData9, Test Data13-Test Data 15 are the night urban expressway traffic video, night highway traffic video, night city street traffic video that collected by industrial camera and driving recorder respectively.

Compared with the AdaBoost method, the proposed method does not need to use the multi-scale sliding window to repeat traverse and calculation and the number of proposal windows that fed into AdaBoost classifiers is reduced by filter operation and clustering algorithm, so it reduces the computation and improves the detection speed. Because the running speed of Edge Boxes algorithm is lower than BING algorithm, the speed of the proposed method is not the fastest in the three methods, but it still ensures the instantaneity requirements of vehicle detection. Table 1 and Table 2 show that the detection time of the proposed method can achieve 41.9ms/frame and 42.9 ms/frame on the test dataset collected by the industrial camera and driving recorder.

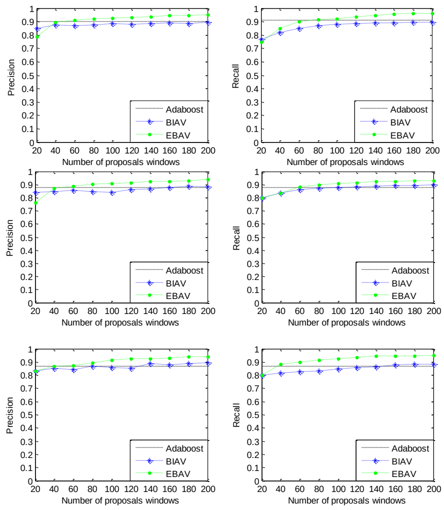


Fig. 9 Illustrative examples showing top to bottom (first row) In sunny weather condition, the results of recall and precision under different number of proposal windows; (second row) In cloudy weather condition, the results of recall and precision under different number of proposal windows; (third row) In night weather condition, the results of recall and precision under different number of proposal windows.

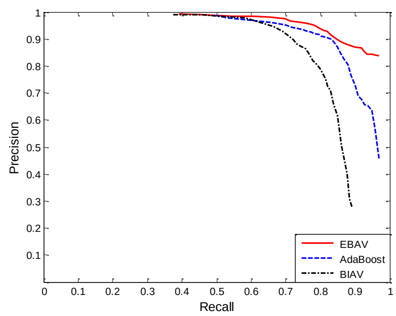


Fig. 10 Illustrative examples showing the PR cure of three methods in the test dataset.

Fig. 11 The results of vehicle detection under different weather conditions (sunny, cloudy, night) and road environments. The three columns on the left are the results of the industrial camera dataset, and the three columns on the right are the results of the driving recorder dataset.

5. Conclusion

In this paper, we present an improved method for vehicle detection, which combines Edge Boxes with AdaBoost classifiers. Firstly, a fast object detection algorithm is used to get a part of proposal windows that may contain an object. Then camera calibration information and vehicle color information are used to remove the windows which not contain vehicle, the normalized cut algorithm is adopted to group the proposal windows after filter operation into various clusters. Finally, a part of proposal windows are selected from each cluster and fed into AdaBoost classifiers to vehicle detection.

Experiment results show that the proposed method can correctly detect the vehicles in different weather conditions (sunny, cloudy, night), and the running time is 41.9ms/frame. The performance comparison shows that the proposed method has a higher detection rate (95%) and accuracy (96%). Because the final detection accuracy of the method is limited by the accuracy of Edge Boxes algorithm, when the vehicle occlusion is very serious, the accuracy of vehicle detection will decrease, this will be the focus of the follow-up study.

References

[1] Nuevo, Jesus, et al. Estimating surrounding vehicles' pose using computer vision. Conference Record-ieee Conference on Intelligent Transportation Systems. 2010, p. 1863-1868.

[2] Chong, Yanwen, et al. Integrated Real-Time Vision-Based Preceding Vehicle Detection in Urban Roads. Neurocomputing. Vol. 116 (2013) No. 10, p. 144-149.

[3] TANG Chun-Ming, CAO Zhi-Sheng,LIN Xiang-Qing. Headlights Detection in Traffic Videos Based on Atmospheric Reflection-scattering Model via Reconstructing Restoration Images. Acta Automatica Sinica. Vol. 42(2016) No. 4, p. 605-616.

[4] Mu K, Hui F, Zhao X, et al. Multiscale edge fusion for vehicle detection based on difference of Gaussian. Optik - International Journal for Light and Electron Optics. Vol. 127(2016) No. 11, p. 4794-4798.

[5] Baek Y M, Kim W Y, et al. Forward vehicle detection using cluster-based AdaBoost. Optical Engineering. Vol. 53(2014) No. 10, p. 2468-2468.

[6] Zhang Z, Warrell J, Torr P H S. Proposal generation for object detection using cascaded ranking SVMs. Computer Vision and Pattern Recognition. Vol. 42(2011) No. 7, p. 1497-1504.

[7] Kim J, et al. Improved Vehicle Detection Method Using Feedback-Ada Boost Learning. International Journal of Computer Theory & Engineering. Vol. 5(2013) No. 1, p. 77-80.

[8] Wen X, Shao L, Xue Y, et al. A rapid learning algorithm for vehicle classification. Information Sciences. Vol. 295(2015) No. 2, p. 395-406.

[9] Satzoda R K, Trivedi M M. Multipart Vehicle Detection Using Symmetry-Derived Analysis and Active Learning. IEEE Transactions on Intelligent Transportation Systems. Vol. 17(2016). No. 4, p. 926-937.

[10] Sivaraman S, Trivedi M M. A General Active-Learning Framework for On-Road Vehicle Recognition and Tracking. IEEE Transactions on Intelligent Transportation Systems. Vol. 11(2010) No. 2, p. 267-276.

[11] Kim J, Baek J, Kim E. On-road precise vehicle detection system using ROI estimation. IEEE 17th International Conference on Intelligent Transportation Systems (ITSC). Qin Dao,China, 2014, p. 2251-2252.

[12] Yuan Q, Thangali A, Ablavsky V, et al. Learning a Family of Detectors via Multiplicative Kernels. Pattern Analysis & Machine Intelligence IEEE Transactions on. Vol. 33(2010) No. 3, p. 514-530.

[13] Wang H S, Luo Y, Zhang B. A Hybrid Method of Vehicle Detection based on Computer Vision for Intelligent Transportation System. International Journal of Multimedia & Ubiquitous Engineering. Vol. 9(2014) No. 6, p 105-118.

[14] Zitnick C L, Dollár P. Edge Boxes: Locating Object Proposals from Edges. Computer Vision – ECCV 2014. Zurich, Switzerland, 2014, p. 391-405.

[15] Cheng M M, Zhang Z, Lin W Y, et al. BING: Binarized Normed Gradients for Objectness Estimation at 300fps. IEEE Conference on Computer Vision and Pattern Recognition. Ohio, USA, 2014, p. 3286-3293.

[16] Hosang J, Benenson R, Schiele B. How good are detection proposals, really?. Computer Science. 2014.

[17] Shi J, Malik J. Normalized cuts and image segmentation. IEEE Trans.pattern Anal.mach.intell.

Vol. 22(2000) No. 8, p. 888-905.
[18] Dollar P, Zitnick C L. Structured Forests for Fast Edge Detection. IEEE International Conference on Computer Vision. Sydney, Australia, 2013, p. 1841-1848.
[19] Chang W C, Cho C W. Real-time side vehicle tracking using parts-based boosting. Systems, Man and Cybernetics. 2008. p. 3370-3375.
[20] Idler C, Schweiger R, Paulus D, et al. Realtime Vision Based Multi-Target-Tracking with Particle Filters in Automotive Applications. Intelligent Vehicles Symposium. 2006. p. 188-193.
[21] Fan R E, Chang K W, Hsieh C J, et al. LIBLINEAR: A Library for Large Linear Classification. Journal of Machine Learning Research. Vol. 9(2008) No. 9, p. 1871-1874.
[22] Han Y X, Zhang Z S, Dai M. Monocular vision system for distance measurement based on feature points. Optics & Precision Engineering. Vol. 19(2011) No. 5, p. 1082-1087.