Optik 127 (2016) 7941–7951

Contents lists available at ScienceDirect

Optik

journal homepage: www.elsevier.de/ijleo

Original research article

Real-time vehicle detection using histograms of oriented gradients and AdaBoost classification

Gang Yan[a,b], Ming Yu[b,*], Yang Yu[b], Longfei Fan[c]

[a] School of Electronic and Information Engineering, Hebei University of Technology, Tianjin, China
[b] School of Computer Science and Engineering, Hebei University of Technology, Tianjin, China
[c] School of Computer Science and Technology, Civil Aviation University of China, Tianjin, China

ARTICLE INFO

Article history:
Received 26 January 2016
Accepted 26 May 2016

Keywords:
Vehicle detection
AdaBoost
Histograms of oriented gradients
Monocular vision

ABSTRACT

Vehicle detection is a major part of a driver assistant system. However, a complex environment and diverse types of vehicles make real-time detection of vehicles a very challenging task. This paper proposes a real-time vehicle detection system of two steps: hypothesis generation and hypothesis verification. In the first step, potential vehicles are detected using shadows under vehicles. In the second step, hypotheses generated in the first step are classified as vehicles and non-vehicles. The novel aspect of this research is in constructing two types of histogram orientation gradients descriptors to extract vehicle features, and then combining them for their final features. The AdaBoost classifier is trained by the combined histogram orientation gradients features. The Treatment Group of images vehicle dataset is adopted for the classifier training. The experiment results show that the proposed system performs well in accuracy and robustness and can meet real-time requirements.

1. Introduction

With the increase in traffic accidents year by year, human and property safety are threatened. Because of the severe con-sequences from traffic accidents, driver assistant systems (DAS) have been a popular research topic in recent times.

Vehicle detection is an important and challenging part of the DAS. Diverse sensors are available for vehicle detection including camera [1,2], radar [3], ultrasound [4], infrared cameras [5] and stereo sensors [6,7]. Vision-based sensors can obtain more information than other sensors, so traffic detection based on cameras plays a more important role in vehicle detection [8,9]. In addition, a monocular camera is cheap and can be flexibly located. However, the videos captured by a camera need sophisticated processing to detect vehicles because of their various shapes, colors and perspectives. This makes vehicle detection using a camera a more challenging task.

Researchers are engaged in vehicle detection systems using a camera. Typical vehicle detection systems traditionally include two steps: Hypothesis Generation (HG) and Hypothesis Verification (HV). In the HG step, the locations of possible vehicles are hypothesized. In the HV step, the hypothesis generated in the HG step is verified.

There are three main methods for HG [10–12]: (1) knowledge-based, (2) stereo-based, and (3) motion-based. The knowledge-based methods use features of the vehicle's appearance, such as its shadow [13], edge [14], texture, symmetry [15], colors and vehicle lights. There are also fusion methods based on this knowledge. Betke et al. [16] used the combina-

[*] Corresponding author.
E-mail addresses: yangang@scse.hebut.edu.cn (G. Yan), yuming@hebut.edu.cn (M. Yu), yuyang@scse.hebut.edu.cn (Y. Yu), lffan@cauc.edu.cn (L. Fan).

http://dx.doi.org/10.1016/j.ijleo.2016.05.092
0030-4026/© 2016 Elsevier GmbH. All rights reserved.

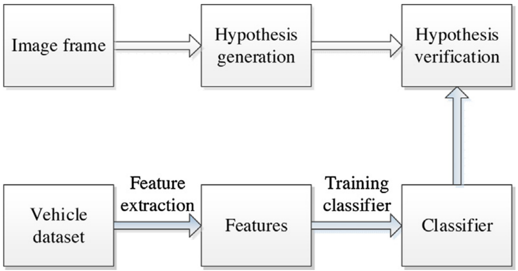

Fig. 1. The flow chart of the system.

tion of color and edge to recognize and track the road boundaries, lane markings and other vehicles on the road. Tsai et al. [17] detected vehicles using color and edges. To deal with the variety of vehicle colors, they transformed the images from the RGB space to a 'vehicle color space' using the Karhunen–Loeve (KL) transform in which vehicle color can be effectively located. Khairdoost et al. [18] adopted a three-step method including shadow detection, texture extraction, and vehicle HG to generate vehicle hypotheses. Stereo vision-based methods mostly used disparity maps and inverse perspective mapping to detect vehicles in Refs. [6,7]. In [19], motion information was used to identify vehicles approaching from the opposite direction.

In the second step, the hypotheses generated in the first step are classified either as vehicles or not. Two major operations are needed in this step: feature extraction and classification. Sun et al. [20] used Gabor filters for vehicle feature extraction and Support Vector Machines (SVMs) for vehicle classification. Gradient features and AdaBoost classification were applied and recorded in the literature [21]. Gradient features are good at describing the appearance and shape of objects. In recent years, some researchers verified vehicles using histograms of oriented gradient (HOG) features which were first used in detecting people [22]. Khairdoost et al. [18] extracted pyramid HOG (PHOG) features and applied principle component analysis (PCA) to reduce the dimension of features, named PHOG-PCA. A Genetic Algorithm and linear Support Vector Machine are then used for vehicle verification based on PHOG-PCA features. However, this method makes the procedure more complex and time-consuming. In [23], vehicle features were extracted by integrating HOG features with symmetrical HOG features that doubled the feature dimensions but resulted in more calculation time. Arróspide et al. used HOG and SVM in Ref. [24] for vehicle detection, simplifying the HOG descriptor by considering only the vertical edges to reduce the computation of HOG features. However, the accuracy was less than for the original HOG features.

In this study, we propose a real-time vehicle detection system based on a monocular camera. This system can run in real-time and guarantee high detection accuracy. In the HG step, the pre-knowledge shadow is used to simplify the processing. In the HV stage, two HOG descriptors are constructed which can extract more vertical and horizontal gradient features but have fewer dimensions than the standard descriptors. SVM and AdaBoost classification were used to train these features.

This paper is organized as follows. In Section 2, the proposed HG and HV steps are introduced in detail. Section 3 shows the experimental results. The conclusions and future work are outlined in Section 4.

2. Proposed vehicle detection system

The proposed vehicle detection system is suitable for a monocular camera mounted on a moving car to detect vehicles in front of it. In this study the system included the two steps: HG and HV. In the HG step, the system generated vehicle hypotheses using pre-knowledge shadows underneath the vehicles. In the second step, integrated HOG features were used to train the SVM and AdaBoost classifiers, and these hypotheses in HV were then separated into vehicles or not. The flow chart of the proposed system is shown in Fig. 1.

2.1. Hypothesis generation using pre-knowledge shadows

Shadows are significant features of on-road vehicles. The research in [10,13,18] adopted shadows underneath vehicles as a clue to determining potential vehicles on the road. When a vehicle is detected using the shadow underneath, its position can be located accordingly. However, illumination significantly affects the shadow. The gray level of the shadow is different on sunny and cloudy days. Consequently it is necessary to set an adaptive threshold to extract the shadow area. An upper bound can be achieved by estimating the gray scale of the road area based on the fact that the shadow under the vehicle is darker than other areas on the road surface. In this study, the free driving space [10] (the road in front of the prototype vehicle) is first obtained to reduce the effect of non-vehicle shadow areas. This operation is performed by extracting the edges of the image using Sobel and then removing the areas directly above the edges (Fig. 2(b)). The remaining region in the image is the free road area (the gray parts in Fig. 2(c)). Second, we assume the distribution of the gray values of the road satisfies a Gaussian distribution [25] as follows:

$$f(x) = 1/\left(\sqrt{2\pi}\sigma\right)\exp\left(-(x-\mu)\hat{}2/\left(2\sigma\hat{}2\right)\right) \tag{1}$$

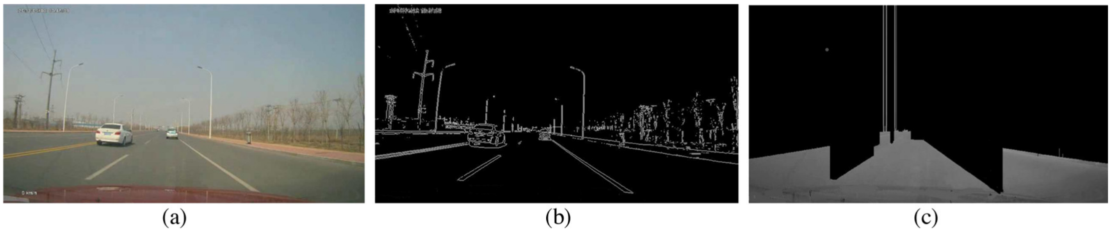

Fig. 2. Detection of the free driving space; (a) the original image, (b) edge image, (c) free driving space.

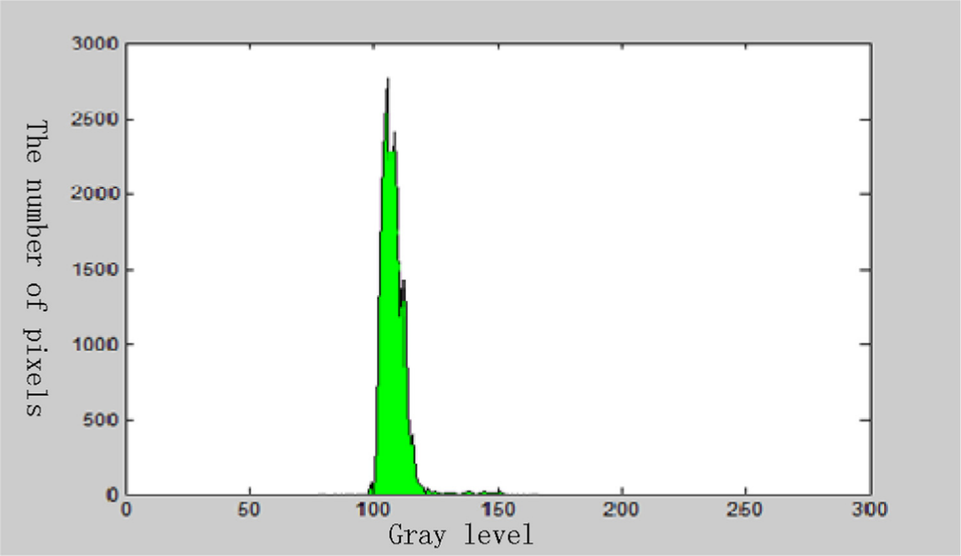

Fig. 3. Gray level distribution of the road region.

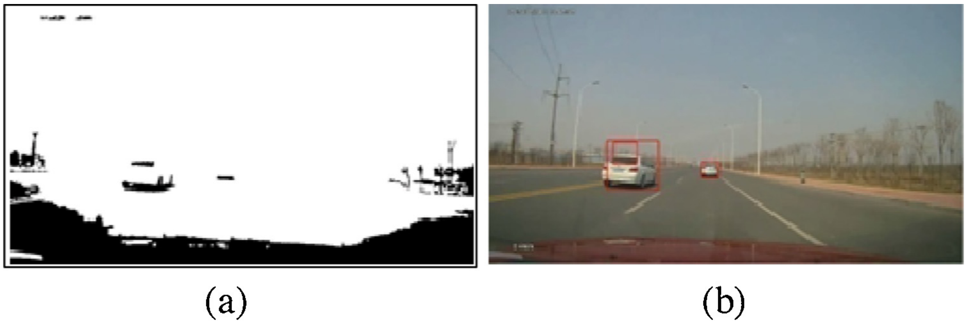

Fig. 4. Hypothesis generation; (a) shadows detected by estimating the gray level of the road, (b) hypotheses generated based on the positions of the shadows.

where μ is the average of gray values and σ is the standard deviation.

Fig. 3 shows the gray level distribution of the road which is the experimental results from 300 frames of the captured video in natural scenes. From Fig. 3 we can see that the distribution is symmetrical and the peak values corresponding to the gray level are between 100 and 110. The upper bound of the shadows under vehicles is set to $Th = \mu - 3\sigma$according to the literature [13] and [18]. The areas $B(x, y)$ are extracted using the following equation:

$$B(x, y) = \begin{cases} 1, & G(x, y) > Th \\ 0, & else \end{cases} \tag{2}$$

where $G(x, y)$ is the input image.

The morphological method is used to make the shadow areas more noticeable. Fig. 4(a) shows the result of the shadow detection. Experiments on the captured videos show that the vehicle takes 50–150 pixels of the 800 × 640 images. Areas are removed if they exceed this range and are non-vehicles. Finally, a bounding box is generated based on the left, right and

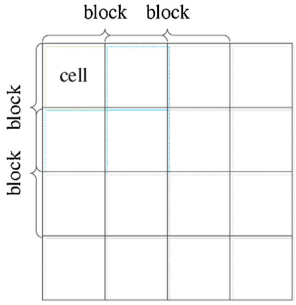

Fig. 5. Standard HOG configuration. Each cell consists of 2 × 2 pixels and each block consists of 2 × 2 cells; the step between two adjacent blocks is two pixels, and there are 9 orientations.

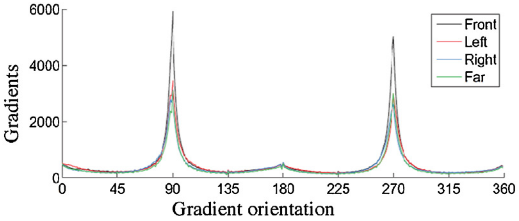

Fig. 6. Gradient orientation distribution of the GTI dataset.

bottom location of each shadow area as shown in Fig. 4(b). These bounding boxes are hypothetical vehicle areas which need to be verified further.

2.2. Hypothesis verification

Gradient information is very commonly used in vehicle detection. The primary gradients of vehicle images are expected to be in the vertical and horizontal directions [24,26]. According to this knowledge, HOG descriptors are used to extract vehicle features. The HOG descriptors extract features based on evaluating well-normalized local histograms of image gradient orientations in a dense grid [22]. The concept focuses on the fact that the character and shape of the local object can be described by local luminance gradients or edge directions. The HOG descriptor is carried out through the following steps:

(1) Divide the image into small regions (“cells”), several cells forming a block;
(2) Accumulate a histogram of gradient orientations from each cell;
(3) Combine and contrast/normalize the histograms in the same block;
(4) Obtain the histogram of the block by combining the histograms in Step (3);
(5) Move to next block and execute Steps (3) and (4);
(6) Combine these block histograms into one vector.

An example of the HOG structure is shown in Fig. 5.

The features of a HOG descriptor are generally used for human detection. In recent years, they have been applied to vehicle detection and achieved a certain degree of success [18,23,24]. However, their main drawback has been the high dimensions, as in the HOG descriptor in Fig. 5. A cell will generate a vector with nine dimensions. Each block has 2 × 2 cells, and the total number of blocks is 3 × 3 = 9. Therefore any 8 × 8 pixel image will create a feature vector with 9 × 2 × 2 × 3 × 3 = 324 dimensions. Consequently, it will take more time for feature extraction and classification, which makes it impossible to realize real-time detection. Vehicle images have many gradient features, as can be seen in Fig. 6, which shows the histogram of the gradient orientations for the GTI (Grupo de Tratamiento de Imágenes) vehicle dataset [27].

As we can see, gradients of vehicle images are mainly distributed in horizontal and vertical directions. This is obvious because the vehicle’s windows, wheels, and contours contribute significantly to the gradients.

To take advantage of this feature, we constructed two HOG descriptors as shown in Fig. 7. In Fig. 7(a), the size of a cell is set to 2 × 4, and a block is formed by 1 × 2 cells. More horizontal gradient features can be obtained using this structure. Similarly, features obtained using the HOG descriptor in Fig. 7(b) can detect more vertical gradient information. Using these two HOG descriptor 72 dimensions of the features are extracted, which is much lower than with the standard HOG descriptor. As a result, feature extraction takes less time. The combination of the two types of HOG feature vectors forms the final feature vector as follows:

$$Vector = \begin{pmatrix} Vector1 \\ Vector2 \end{pmatrix} \tag{3}$$

where $Vector1$ and $Vector2$ represent the horizontal and vertical HOG features individually, and $Vector$ is the final vector.

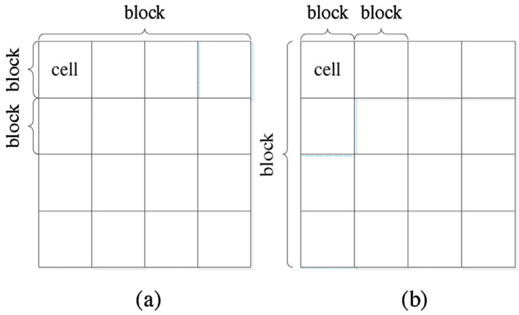

Fig. 7. Proposed HOG descriptors; (a) HOG descriptor for a horizontal gradient feature, (b) HOG descriptor for a vertical gradient feature.

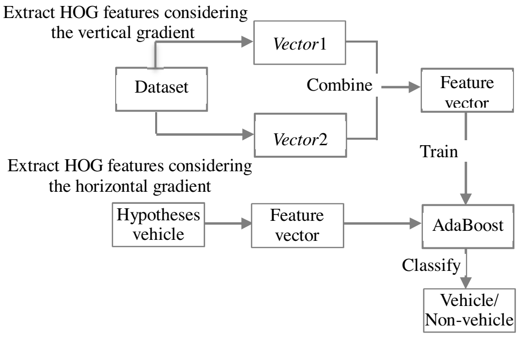

Fig. 8. Frame of the hypothesis verification.

Most vehicle features extracted by the HOG descriptor were classified using SVM [17,21–23]. The AdaBoost classifier was also applied in vehicle detection. In [28], Haar-like features with AdaBoost were used to verify the hypothesis. Khammari et al. [21] combined gradients with AdaBoost for vehicle detection. In this study, we trained both AdaBoost and SVM classifiers with combined feature vectors.

The procedure for HV is shown in Fig. 8. Horizontal and vertical feature vectors were extracted using the proposed HOG descriptor, and then AdaBoost and SVM were trained using the combined features individually. Second, we used the AdaBoost and SVM classifiers to verify the HG.

3. Experiment results

The database used in the experiments contains two parts. The first was the GTI vehicle database, which was used to train the classifier. The second was the videos collected in real traffic scenes used for hypothesis generation and testing the classifier. All of the following experiments are conducted on Window 7 on a 2.13GHZ Intel Core 2 CPU equipped with 2GB Memory and Matlab 2012b.

The GTI vehicle dataset included 3425 vehicle images and 3900 non-vehicle images classified into four models: distant (1950 simples), middle (1475 simples), close left (1950 simples) and close right (1950 simples). Vehicles in this dataset vary in shape, color and size, and all the images are normalized to the same size of 64 × 64, as shown in Fig. 9(a). Therefore, the classifier trained by these samples can have a higher recognition rate in different conditions.

The real traffic scenes are captured by a driving record mounted on the front window of the car. Real traffic scenes contain many interference, such as traffic lines, trees and billboards (see in Fig. 9(b)). We captured totally 63,000 frames under strong and weak illumination. Besides, the captured data contain videos with side light which will stretch the shadow can gain the detection difficulty.

3.1. Experiments on hypotheses generation

The videos used were collected on a city road under different light intensities. It was challenging to detect a vehicle in these videos because of the complex background. Billboards, wires, street lights and trees are very common on city roads and increase the difficulty of vehicle detection.

Fig. 10 shows the hypothesis results using pre-knowledge shadows. The video in the first row of Fig. 10 is captured on a sunny day, and the rear window of a car was falsely treated as shadows because of its dark color. Thus the system generated a false hypothesis. In the other two rows, the trees beside the road also generated false hypotheses because the low illumination made the trees become dark areas after binarization. However, the goal in this step was to obtain potential

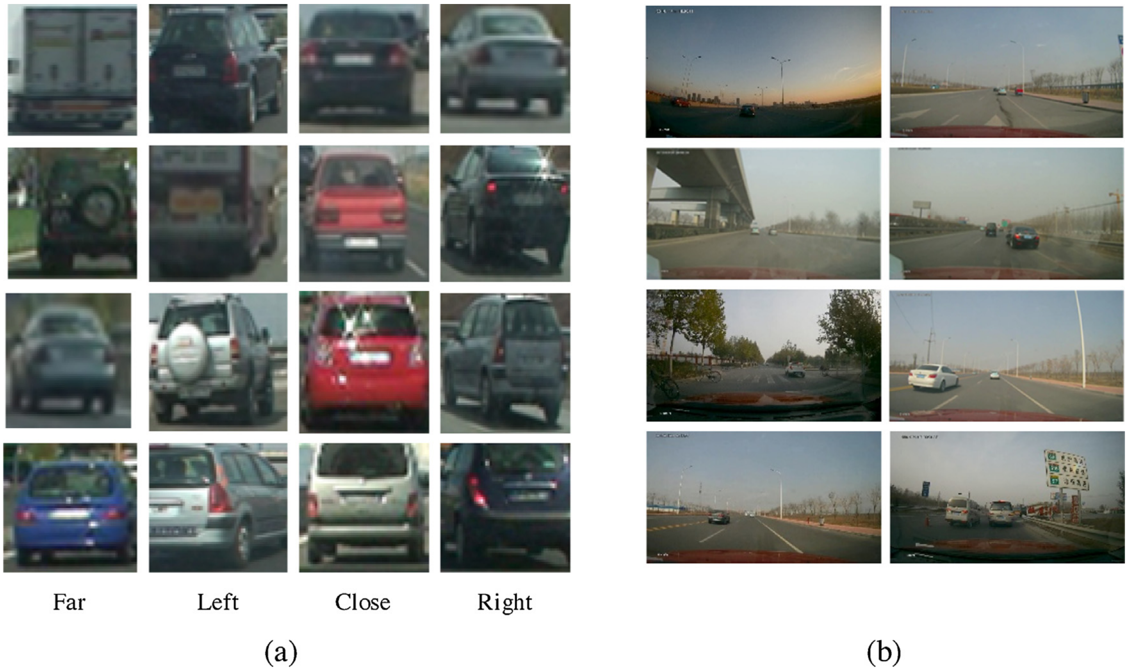

Fig. 9. The database used in the experiments; (a) the Vehicle Dataset GTI Vehicle Dataset, (b) the real traffic scenes dataset.

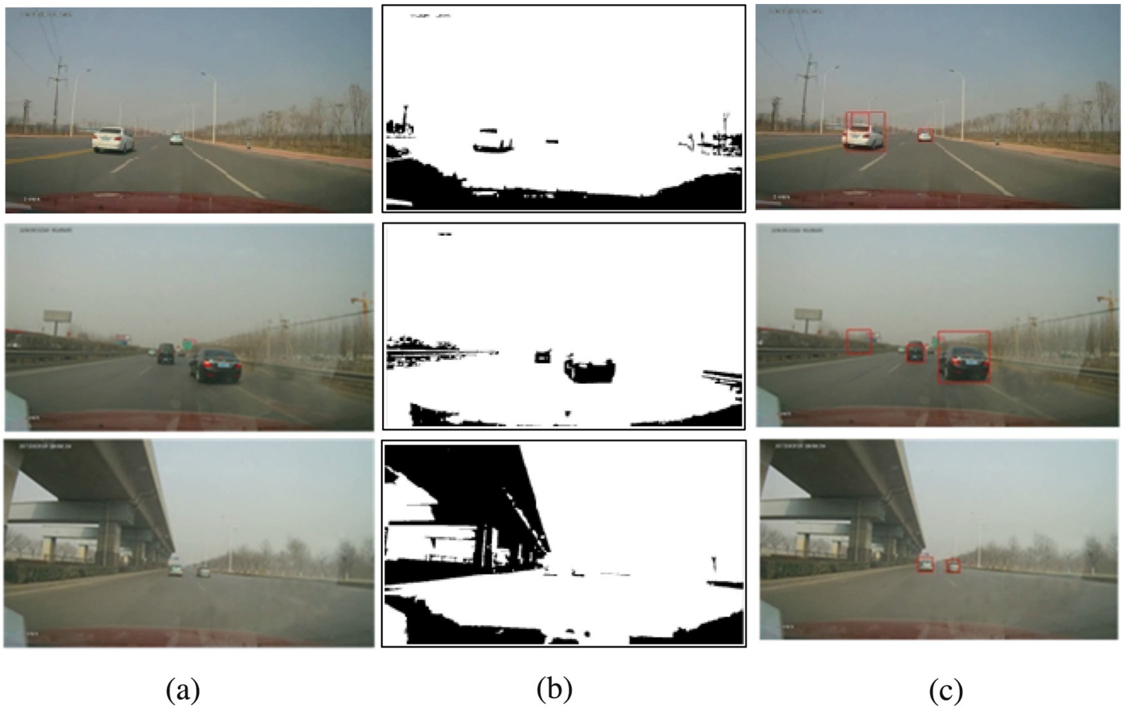

Fig. 10. Hypothesis generation results; Column (a) original images, (b) binarization results, (c) vehicle hypotheses.

vehicles, and the false hypothesis would be removed in the HV step. The following experiments showed the efficiency of our method in the HV step.

The sizes of the videos are 800 × 640. As we use the pre-knowledge shadow under the vehicle as the only clue for the hypothesis generation, it cost 0.01s per frame.

Table 1
The HOG feature vectors.

Vector	Cell	Block	Dimensions	Time (s)
1	4 × 4	2 × 2	8100	0.184087
2	8 × 8	2 × 2	1764	0.166573
3	16 × 16	2 × 2	324	0.122706
4	32 × 4	2 × 2	540	0.131331
5	4 × 32	2 × 2	540	0.129522
6	32 × 4	2 × 4	504	0.084968
7	4 × 32	4 × 2	504	0.085902

Table 2
Accuracy of SVM.

Vector	SVM accuracy (%)			
	Front	Left	Right	Far
1	97.20	93.85	89.55	95.29
2	**98.00**	96.11	90.79	96.34
3	96.80	94.26	86.48	96.31
4	97.20	96.51	**91.60**	96.93
5	84.80	81.15	74.80	81.56
6	92.40	92.42	79.92	90.98
7	96.00	**96.93**	91.39	**97.54**

Table 3
Accuracy of AdaBoost.

Vector	AdaBoost accuracy (%)			
	Front	Left	Right	Far
1	97.40	93.65	88.32	96.31
2	**98.60**	**97.67**	92.55	97.57
3	96.80	95.90	93.65	97.54
4	98.00	96.52	93.43	96.71
5	88.40	83.33	78.74	84.19
6	**98.60**	96.01	**94.84**	97.55
7	98.40	96.72	94.47	**98.52**

3.2. Experiments on the hypothesis verification

SVM and AdaBoost with HOG features, which are two successful methods of pattern classification, were used to verify hypotheses [29,30]. AdaBoost is an ensemble method which obtains strong classifiers by combining multiple weak classifiers linearly. In this paper, the GML AdaBoost Matlab Toolbox is used for AdaBoost classification and the iteration number is set to be 50 which can make a judgment by less time. For SVM, in our experiments we used the svmlight library and a linear kernel in all of our experiments. To test the efficiency of AdaBoost with HOG features, we made a comparison of SVM and AdaBoost using the HOG features extracted from the GTI vehicle dataset that contained 3425 positive samples and 3900 negative samples. The GTI vehicle dataset was separated equally into training and testing samples for all the experiments. Support Vector Machine (SVM) classifier.

The HOG descriptors extracted local gradient features allowing for the geometry invariance and illumination anti-deformation. However, the structures of the HOG descriptor significantly impacted on the features, including the cell size, block size, number of orientation bins, and step of each block. Experiments were carried out using different HOG structures with SVM and AdaBoost individually. First, HOG descriptors with different structures were used to extract HOG feature vectors as shown in Table 1. The numbers of feature dimensions are listed in the table. The number of orientation bins was set as 9 and the step to half the block size, which was most commonly used [22,31]. Feature vector dimensions changed significantly with the sizes of cells and blocks from 8100 to 504. The run time taken for feature extraction decreased with the dimensions.

Each HOG feature vector was applied to train the SVM classifier and the AdaBoost classifier individually. The recognition rates of these two classifiers for the four types of samples (front, left, right and distant) are shown in Tables 2 and 3. As can be observed, the recognition rates for the front images were higher than for other images. Table 4 shows the average accuracy and classification time for all samples. The AdaBoost classifier performed better than the SVM classifier in accuracy and running time. In Tables 2–4, the highest accuracy in each column is marked in bold.

Table 4
Average accuracy and running time of SVM and AdaBoost.

Vector	Average accuracy (%)		Running time (s)	
	SVM	AdaBoost	SVM	AdaBoost
1	**96.51**	95.99	0.182543	0.008217
2	96.61	**97.24**	0.023558	0.008326
3	96.10	**96.79**	0.007856	0.008109
4	**96.89**	96.45	0.015661	0.008527
5	88.64	**90.03**	0. 015188	0.008372
6	96.83	**96.91**	0. 006912	0.008124
7	93.12	**96.02**	0.007868	0.008313

Table 5
Accuracy of the combined HOG features and standard HOG features.

Vector	Dimensions	Accuracy (%)					Running time (ms)
		Front	Left	Right	Far	Average	
2	1764	97.60	94.67	91.55	97.57	97.24	0.166573
6 + 7	1008	99.02	98.22	97.31	98.88	98.82	0.170870

From the above tables, we see that the AdaBoost classifier achieved higher accuracy on average in feature vectors 2, 3, 5, 6 and 7, while the SVM classifier performed better for feature vectors 1 and 4. However, the AdaBoost classifier showed more advantages than the SVM classifier in terms of running time which is more important for a real-time system.

3.3. Combination of two proposed HOG feature vector

In Table 4, the highest accuracy reached 97.24%. The dimensions of the feature vector that led to the highest accuracy were 1764 (see Table 1). The accuracies of feature vectors 6 and 7 reached 96.91% and 96.02% which were lower than the best, but the dimensions of these two vectors were much lower than for feature vector 2. Feature vector 6, as described in Section 2, contained more horizontal gradient information, and feature vector 7 contained more vertical gradient information. Therefore we combined these two vectors into one feature vector and trained the AdaBoost classifier. Table 5 shows the comparison of feature vector 2 with the new combined feature vector.

As Table 5 shows, the combined feature had higher accuracy than feature vector 2, and the average accuracy of the new feature vector was 1.58% better than for feature vector 2 which was the best result in Table 4. The dimensions of the combined feature vector were lower than for feature vector 2. The extraction time of the new feature is calculated by adding the feature extraction times of feature vectors 6 and 7, and this took 0.004 ms more than feature vector 2 which has no effect on real-time.

3.4. Experiments on the real traffic scenes

To test the efficiency of the proposed system, we collected real traffic videos using a monocular camera, with a video frame rate of 30fps, mounted on a car. The experimental results showed that our system met the real-time requirement. The system detected the left, right, distant, and front vehicles (see Fig. 11). It overcame the strong and low light conditions because the shadow under a vehicle was darker than for the area of the road that was not affected by light intensity. Potential vehicles were detected in the HG step shown on the second row of Fig. 11. Non-vehicles hypotheses were removed in the HV step (the third row in Fig. 1).

3.5. Some failure examples

In the experiments, when vehicles occlude with each other, shadows under vehicles may join together which can affect the detection accuracy. As a result, the occluded vehicles may be considered as one vehicle in the HG step, and then the classifier will make the wrong decision and classify this hypothesis into non-vehicles as shown in Fig. 12. A bus and a car join together in Fig. 12(a). These two vehicles are treated as one because their shadows connected with each other (Fig. 12(a)). Then, in the second step, this hypothesis is classified as non-vehicles as shown in Fig. 12(a). Vehicles occlusion is the main work that we are going to do in the future.

4. Conclusion

This study proposes a real-time vehicle detection system using a monocular camera. The experimental system included two steps: the HG and HV steps. In the first step, the shadow was taken as the only information for generating hypotheses

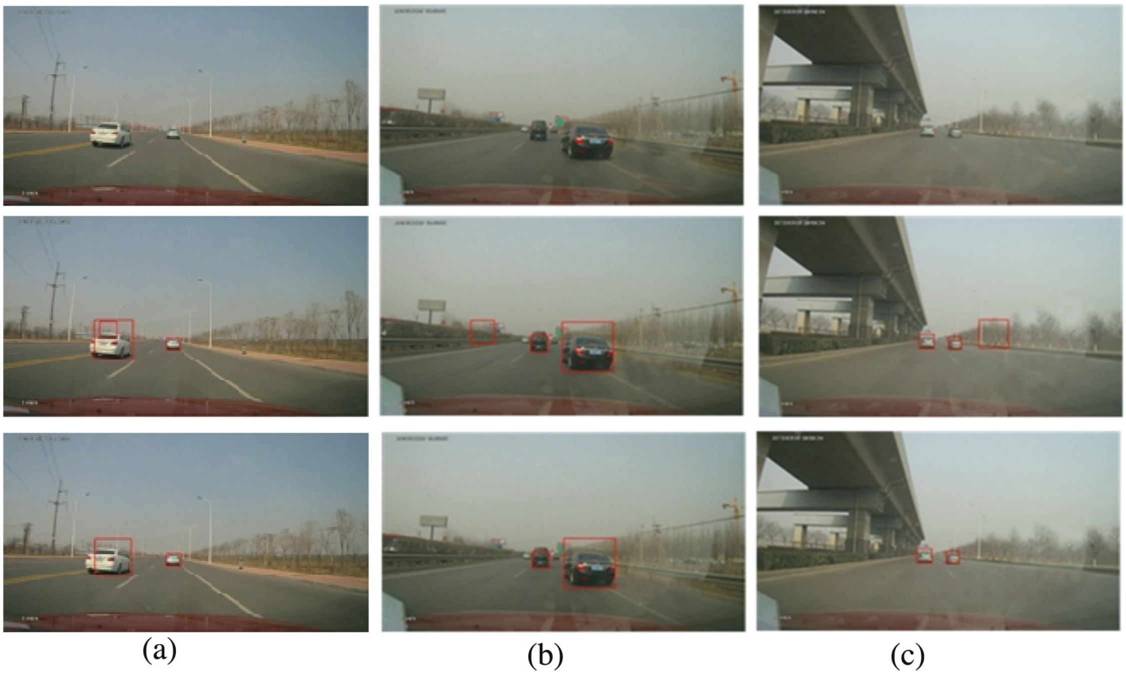

Fig. 11. Experimental results of the vehicle detection system in the real traffic scene. (a) Video with the left vehicle in strong light; (b) video with front and right vehicles in low light; (c) Video with distant vehicles.

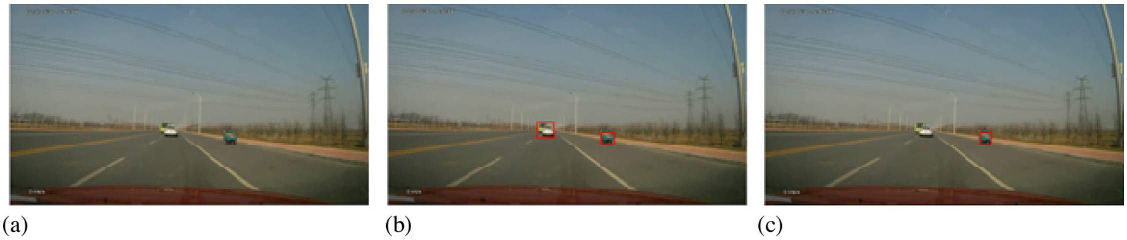

Fig. 12. Detection with vehicles occlusion (a) a bus joined with a car; (b) two vehicles are treated as one; (c) two vehicle are classified as non-vehicle because of occlusion.

which only cost 0.01s run time per frame and also guaranteed the precision ratio because each vehicle on the road cast a shadow. In the HV step, a series of HOG feature vectors were extracted. Both SVM and AdaBoost classifiers were trained using these features. Based on a comparison of the results of these two classifiers, it was concluded that the AdaBoost classifier performed better in terms of accuracy and run time. The accuracy reached 97.24% (see Table 4) using the HOG features, but the dimensions were very high. Such a system cannot run in real traffic scenes. Considering the drawbacks of the HOG features, we constructed two HOG descriptors that can generate features with lower dimensions and contain more vertical and horizontal gradient features, to fit the vehicle structure. The recognition rate increased to 98.82% with the combined features, and the feature dimension was reduced by half. The system met the real-time and accuracy requirements.

Acknowledgements

This work was granted by Tianjin Sci-tech Planning Projects (Grant No. 14RCGFGX00846), the Natural Science Foundation of Hebei Province, China (Grant No. F2015202239), Tianjin Sci-tech Planning Projects (Grant No. 15ZCZDNC00130), and National Natural Science Foundation of China (Grant No. 61305107).

References

[1] J.S. Kang, J. Kim, M. Lee, Advanced driver assistant system based on monocular camera, IEEE International Conference on Consumer Electronics (ICCE) (2014) 55–56.
[2] B. Jia, R. Liu, M. Zhu, Real-time obstacle detection with motion features using monocular vision, Vis. Comput. 31 (3) (2015) 281–293.

[3] X. Liu, Z. Sun, H. He, On-road vehicle detection fusing radar and vision, IEEE International Conference on Vehicular Electronics and Safety (ICVES) (2011) 150–154.

[4] E. Sifuentes, O. Casas, R. Pallas-Areny, Wireless magnetic sensor node for vehicle detection with optical wake-up, IEEE Sens. J. 11 (8) (2011) 1669–1676.

[5] Y. Iwasaki, S. Kawata, T. Nakamiya, Robust vehicle detection even in poor visibility conditions using infrared thermal images and its application to road traffic flow monitoring, Meas. Sci. Technol. 22 (8) (2011) 085501.

[6] C.H. Lee, Y.C. Lim, S. Kwon, J.H. Lee, Stereo vision–based vehicle detection using a road feature and disparity histogram, Opt. Eng. 50 (2) (2011) 027004 (027004).

[7] A. Bensrhair, M. Bertozzi, A. Broggi, A. Fascioli, S. Mousset, G. Toulminet, Stereo vision-based feature extraction for vehicle detection, IEEE Intelligent Vehicle Symposium 2 (2002) 465–470.

[8] H. Wang, Y. Cai, Monocular based road vehicle detection with feature fusion and cascaded Adaboost algorithm, Optik-Int. J. Light Electron Opt. 126 (22) (2015) 3329–3334.

[9] H. Wang, C. Yuan, Y. Cai, Smart road vehicle sensing system based on monocular vision, Optik-Int. J. Light Electron Opt. 126 (4) (2015) 386–390.

[10] Z. Sun, G. Bebis, R. Miller, On-road vehicle detection: a review, IEEE Trans. Pattern Anal. Mach. Intell. 28 (5) (2006) 694–711.

[11] X. Wen, L. Shao, W. Fang, Y. Xue, Efficient feature selection and classification for vehicle detection, IEEE Trans. Circuits Syst. Video Technol. 25 (3) (2015) 508–517.

[12] H. Wang, C. Yuan, Y. Cai, Smart road vehicle sensing system based on monocular vision, Optik-Int. J. Light Electron Opt. 126 (4) (2015) 386–390.

[13] C. Tzomakas, V.W. Seelen, Vehicle detection in traffic scenes using shadows, in: IR-INI, Institut Fur Nueroinformatik, Ruhr-Universitat, 1998.

[14] Z. Sun, G. Bebis, R. Miller, Monocular precrash vehicle detection: features and classifiers, IEEE Trans. Image Process. 15 (7) (2006) 2019–2034.

[15] S.S. Teoh, T. Bräunl, Symmetry-based monocular vehicle detection system, Mach. Vis. Appl. 23 (5) (2012) 831–842.

[16] M. Betke, E. Haritaoglu, L.S. Davis, Real-time multiple vehicle detection and tracking from a moving vehicle, Mach. Vis. Appl. 12 (2) (2000) 69–83.

[17] L.-W. Tsai, J.-W. Hsieh, K.-C. Fan, Vehicle detection using normalized color and edge map, IEEE Trans. Image Process. 16 (3) (2007) 850–864.

[18] N. Khairdoost, S.A. Monadjemi, K. Jamshidi, Front and rear vehicle detection using hypothesis generation and verification, Signal Image Process.: Int. J. (SIPIJ) 4 (4) (2013) 31–50.

[19] C. Demonceaux, A. Potelle, D. Kachi-Akkouche, Obstacle detection in a road scene based on motion analysis, IEEE Trans. Veh. Technol. 53 (6) (2004) 1649–1656.

[20] Z. Sun, G. Bebis, R. Miller, On-road vehicle detection using Gabor filters and support vector machines, 14th International Conference on Digital Signal Processing 2 (2002) 1019–1022.

[21] A. Khammari, F. Nashashibi, Y. Abramson, C. Laurgeau, Vehicle detection combining gradient analysis and AdaBoost classification, Proc. of IEEE Intell. Transp. Syst. (2005) 66–71.

[22] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 1 (2005) 886–893 (12).

[23] M. Cheon, W. Lee, C. Yoon, M. Park, Vision-based vehicle detection system with consideration of the detecting location, IEEE Trans. Intell. Transp. Syst. 13 (3) (2012) 1243–1252.

[24] J. Arróspide, L. Salgado, M. Camplani, Image-based on-road vehicle detection using cost-effective histograms of oriented gradients, J. Vis. Commun. Image Represent. 24 (7) (2013) 1182–1190.

[25] S. Yang, J. Xu, Y. Chen, M. Wang, On-road vehicle tracking using keypoint-based representation and online co-training, Multimed. Tools Appl. 72 (2) (2014) 1561–1583.

[26] J. Arróspide, L. Salgado, M. Nieto, F. Jaureguizar, On-board robust vehicle detection and tracking using adaptive quality evaluation, 15th IEEE International Conference on Image Processing (ICIP) (2008).

[27] J. Arróspide, L. Salgado, M. Nieto, Video analysis-based vehicle detection and tracking using an MCMC sampling framework, EURASIP J. Adv. Signal Process. 2012 (1) (2012) 1–20.

[28] S. Sivaraman, M.M. Trivedi, A general active-learning framework for on-road vehicle recognition and tracking, IEEE Trans. Intell. Transp. Syst. 11 (2) (2010) 267–276.

[29] H. Zhang, C. Gu, Support Vector Machines Versus Boosting, Electrical Engineering UC, Berkeley, 2008, pp. 1–19.

[30] X. Wen, L. Shao, W. Fang, Y. Xue, Efficient feature selection and classification for vehicle detection, IEEE Trans. Circuits Syst. Video Technol. 25 (3) (2015) 508–517.

[31] Y. Song, H. Wang, J. Zhang, A vision-based broken strand detection method for a power-line maintenance robot, IEEE Trans. Power Deliv. 29 (5) (2014) 2154–2161.

Gang Yan was born in Tangshan on the 3rd July, 1977 and received his B.S. degree and M.S. degree in Computer Science, from the Hebei University of Technology, China, in 2000 and 2003, respectively. He currently is a PhD. candidate in the School of Information, Hebei University of Technology since 2011. His research interests include Intelligent Transport System and Mobile Internet Application development.

Ming Yu received his BS degree from Beijing University of Post and Telecommunications, China in 1986, the MS degree from Hebei University of Technology, China in 1989, and his PhD degree in communication and information systems from Beijing Institute of Technology, China in 1999. He is currently a professor in Hebei University of Technology. His current research interests include image video understanding, intelligent media processing and pattern recognition.

Yang Yu received his B.S. degree in Electronics Science and Technology, M.S. degree in Pattern Recognition and Intelligent Systems, and PhD. degree in Microelectronics and Solid State Electronics from Hebei University of Technology, Tianjin, China, in 2004, 2008 and 2012 respectively. He is currently a lecturer at School of Computer Science and Engineering of Hebei University of Technology, Tianjin, China. His research interests include image processing and analysis, pattern recognition, computer vision and intelligent transportation system.

Longfei Fan is currently a lab technician at Civil Aviation University of China. He received his Master degree in Computer Technology from Hebei University of Technology, Tianjin, China, in 2015. His interest of research focuses on image process and pattern recognition.