Computing and Informatics, Vol. 36, 2017, 793–814, doi: 10.4149/cai_2017_4_793

REAL-TIME TRAFFIC LIGHT RECOGNITION BASED ON C-HOG FEATURES

Xuanru Zhou

Beijing Key Laboratory of Information Services, Beijing Union University
Beijing 100101, China
e-mail: zhouxuanru1989@163.com

Jiazheng Yuan*

Scientific Research Office, Beijing Open University
Beijing 100081, China
e-mail: xxtjiazheng@buu.edu.cn

Hongzhe Liu

Beijing Key Laboratory of Information Services, Beijing Union University
Beijing 100101, China
e-mail: xxthongzhe@buu.edu.cn

Abstract. This paper proposes a real-time traffic light detection and recognition algorithm that would allow for the recognition of traffic signals in intelligent vehicles. This algorithm is based on C-HOG features (Color and HOG features) and Support Vector Machine (SVM). The algorithm extracted red and green areas in the video accurately, and then screened the eligible area. Thereafter, the C-HOG features of all kinds of lights could be extracted. Finally, this work used SVM to build a classifier of corresponding category lights. This algorithm obtained accurate real-time information based on the judgment of the decision function. Furthermore, experimental results show that this algorithm demonstrated accuracy and good real-time performance.

Keywords: C-HOG features, SVM, traffic light recognition, intelligent vehicles

* corresponding author

1 INTRODUCTION

The increasing levels of automotive vehicle usage on roads have increased the traffic crash frequencies, which in turn, causes an increase in the number of fatalities and injuries on roads every year. Consequently, Intelligent Transportation Systems (ITS) have caught researchers' attention, particularly the visual systems based on image processing technologies. Most importantly, this includes a road sign recognition, a crossing pedestrian recognition, a license plate recognition, and a traffic light recognition [1, 2]. Traffic signal detection plays an important role that would allow ITS to ensure safety when driving through intersections.

Main research question of this study is to use machine learning method to achieve the recognition of traffic lights in intelligent cars. This paper presents a method for the traffic signal identification that is based on the C-HOG feature. C-HOG feature is a new feature proposed in this paper. C-HOG includes a HOG component and a block color histogram component. The HOG (Histograms of Oriented Gradients) is a method that was first proposed by Dalal and Triggs in 2005. The edge of the traffic light is where the gradient largely exists, and the nature of the HOG is a statistic of the gradient information, allowing the HOG to accurately describe the shape of the traffic light. At the same time, the histogram divided the entire target area into sub-blocks, allowing the target local color to be reflected in the whole eigenvector. This method has some interference immunity; C-HOG features combine two characteristics: shape and color, which describes the characteristics of traffic lights. This paper will first show how to convert the RGB space to the YCbCr space to identify the red and green areas, and then filter out the irrelevant region. The C-HOG feature of the target area will then be extracted, finally allowing the traffic lights to be identified by an SVM classifier. This method can be applied in intelligent vehicles and ensure safety. Intelligent cars can identify traffic lights in real time by the proposed method, and with a higher accuracy rate. This is the main goal of this paper.

2 BACKGROUND

This section reviews the most relevant methods for automatic traffic light identification. In recent years, many researchers have focused on intelligent identification of traffic lights with attention to the following aspects: signal characteristics, including the color and shape of the traffic lights; the machine's learning process; a priori map.

Several previous studies aimed to identify color characteristics of traffic signals, such as studies based on HIS space [3, 4, 5], HSV space [6, 7, 8], RGB space [9, 10], and the similar distance of RGB space [11], and Lab color space [12, 13]. Since the color component is closely related to luminance in the RGB color space, three components including R, G and B will subsequently change as long as brightness changes, so RGB color space has a poor threshold adaptation and it is not suitable for extraction of the color signal. Wavelet transformation was used to convert RGB space to HIS or HSV space, although other space is possible in order to eliminate the correlation between components. However, other problems exist, including the large amount of computation time, it is time-consuming, which results in poor real-time performance, plus the singularity problems.

Furthermore, other researchers studied the shape features of traffic signals [14, 15, 16, 17, 18, 19, 20, 21] including calculating circularity of the candidate area [14, 15, 16], calculating the standard deviation of the round candidate area [17], Hough transform circle detection [18, 19, 20], and calculating the rectangle of the candidate region [21]. These methods require high quality images obtained from the video, an obvious characteristic, but they are not conducive to shape recognition at large distances. Another method used by Cheng uses the light's rectangular plate for template matching [22], but it relies too much on the rectangle plate, which is not highly detectable in evenings or on cloudy days. Charette [23] and Iwasaki [24] also applied the template matching method to identify traffic lights, but template matching is time and computer memory consuming. In 2009, Charette and Nashashibi [25] proposed an approach to detect traffic lights using the geometry information of traffic light poles, what has showed promising results. However, this method requires exceptionally high quality images. At the same time, the practicability was poor because there is not a single shape of traffic light poles.

A few other researchers used machine learning to identify traffic lights, most of which were based on the classifier, such as the method used by Chiang et al. [26]. Their method extracted the local binary features of traffic lights and used classifiers to train and recognize their characteristics, but the study was limited and could only identify circular traffic lights. Another method by Cai et al. [27] used a wavelet transform and a classifier to identify the arrow-shaped lights, but the processing time per frame reached 152 millisecond, and therefore it could not fulfill real-time requirements. Kim [28] used the SVM method to identify lights; however, this method was only applicable during nights.

Other researchers have used a method based on an A Priori Map [29, 30]. The drawing of an a priori map in advance is necessary, and the vehicle must be equipped with GPS and with inertial sensors and cameras to draw a large number of images. The position of the traffic lights appearing in the camera is calculated based on a priori maps. However, this method requires a lot of manpower and resources to draw the a priori map.

3 DETECTION AND IDENTIFICATION OF TRAFFIC LIGHTS USING THE C-HOG FEATURE

The method used in this paper can be implemented at different times during the daylight only (morning, noon and afternoon). It also can be implemented during different weather conditions – in sunny, cloudy, and overcast days. Different types of lights can be identified using this method, including circular lights, and directional arrow-shaped lights. Moreover, this method can recognize colors including red and green. The algorithm is divided into two parts: detection and identification of traffic lights.

4 DETECTION OF TRAFFIC LIGHTS

Figure 1 shows three steps to traffic light detection: color extraction, color suppression and regional filter.

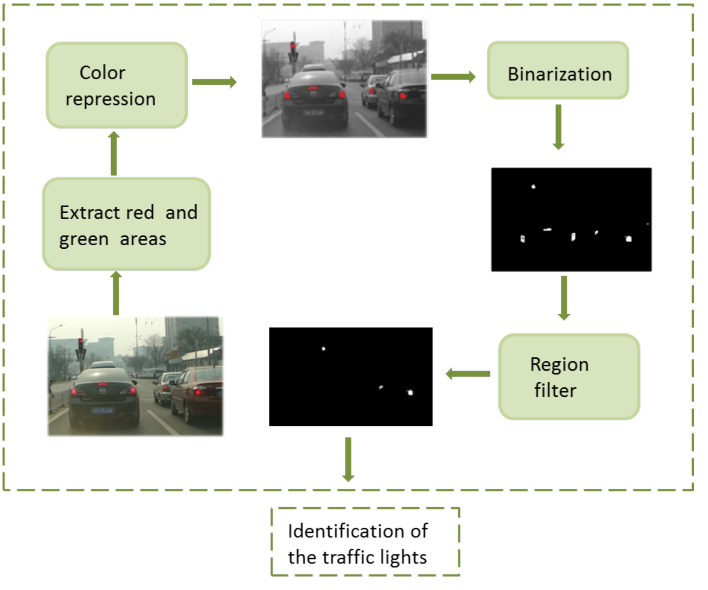

Figure 1. Three steps of detection

4.1 Color Extraction

Choosing an appropriate color space is the key for color extraction. The YCbCr space can describe the color characteristics in traffic lights, and was used by Cai et al. [31]. By observing the red light histogram at the Cb channel and the green light histogram at the Cr channel, the value of red color is found to have a range in Cb channel, the value of green color is found to have a range in Cr channel. By observing these ranges, the thresholds were determined in order to segment colors. Furthermore, the conversion from RGB to the YCbCr space requires less time compared to other spaces. After many experiments the results have shown that the YCbCr space is relatively stable to identify colors.

The color of traffic lights is observable at the Cb and Cr channels. The corresponding pixel values of red and green areas at the Cb and Cr channels were extracted to find the region of interest. This is followed by the Color Suppression step, which incorporates enhancing the color of the ROI (Region of Interest) and repressing the color of other non-related areas, as shown in Figure 1. This step aims to enhance the colors of red and green areas in an image.

4.2 Region Filter

The next step includes excluding other objects that have similar colors to traffic lights and were not excluded in the previous step. As this research developed a recognition algorithm for each color area, any increase in the color areas would increase the time required to perform the algorithm. Therefore the use of a regional filter could reduce the number of the non-light areas, thus improving the running time and accuracy of the algorithm. A threshold was used during the region filter process. Thresholds were obtained through a large number of experiments and thus the accuracy of the filtering process was ensured. In order to avoid the impact on the detection and identification of any interference, binarization was implemented on the ROI. All the contours of the binary images were then traversed, and the areas in line with the characteristics of traffic lights based on contour feature were filtered out.

4.2.1 Area Filter

Because of the light present in a small proportion of the images, other characteristics were not obvious at longer distances. For example, a red light appears as a red dot at a distance, and therefore making lights difficult to identify, regardless of the use of other features as circle feature, etc. Therefore, the algorithm is only valid at a maximum distance of 70 meters from the signalized intersection stop line, while the minimum distance is 0.5 meters or less. However, the algorithm can still identify the lights at distances over the maximum value, but at slightly lower accuracy. Assuming there are N profiles in the image, marked as R_i, then we strike the area of the contours. A represents the area of all the contours.

$$Bool(R_i(A)) = \begin{cases} 1, & A_{max} \le A \le A_{min}, \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

4.2.2 Circumscribed Rectangular Filter

The circumscribed rectangle of each contour was struck one by one to find the appropriate width and height, the condition of the filter is the aspect ratio [25] of the rectangle.

$$R_{wh} = \frac{R_i(\text{width})}{R_i(\text{height})}, \tag{2}$$

$$Bool(R_i(R_{wh})) = \begin{cases} 1, & 0.8 \le R_{wh} \le 1.5, \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

4.2.3 Density Filter

All filtration steps of Section 4.2 have been based on binary images. Each pixel value of the ROI is 1. Therefore, the density is defined as a ratio of all pixels in the ROI, divided by the area of ROI. The threshold was calculated using a large number of relevant experiments. ρ represents the density of the contour, $f(x, y)$ represents the pixel value of the point (x, y), A represents the area.

$$\rho = \frac{A}{\sum_{i=1}^{m}\sum_{j=1}^{n} f(x, y)}, \tag{4}$$

$$Bool(R_i(\rho)) = \begin{cases} 1, & \rho \geq 0.6, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

After three filtration steps defined above, the color blocks that met the conditions were retained. Then the coordinates of the center of the color block and the width and height of the circumscribed rectangle were calculated.

$$Filter(R_i) = \begin{cases} R_i, & Bool(R_i(A)) \cap Bool(R_i(R_{wh})) \cap Bool(R_i(\rho)), \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

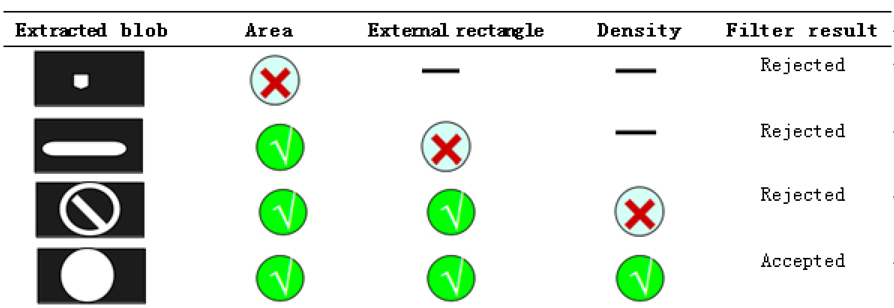

Figure 2. Filter schematic

5 IDENTIFICATION OF TRAFFIC LIGHTS

5.1 Region Selection

The areas to be detected were determined in the original image according to the coordinates of the center, width and height of the color area in the binary image.

For example, the red light in the binary image of the red light is a white circle, with known location information. That location information was used to locate the red circle in the original image. The width (W) and height (H) of the external rectangle of the red circle is known, it was calculated in the Section 4.2.2. The width of the area to be detected is 1.4 W; the height of the area to be detected is 4.2 H. This area can include the edge of the light and extract the features that are richer, as shown below: The thresholds in this section are the means after a great deal of testing, and have strong generalization ability.

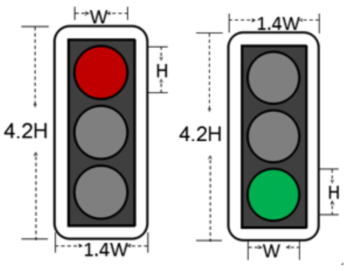

Figure 3. Region schematic

5.2 C-HOG Feature Extraction

The HOG features can describe the shape of the lights, the block color histogram feature can describe local colors, and the combination of the two features called C-HOG in this paper is a perfect description of the whole feature of traffic lights.

5.2.1 HOG Feature Extraction

The HOG (Histograms of Oriented Gradients) is a method that was first proposed by Dalal and Triggs [32] in 2005. HOG describes the distribution of gradient intensity and gradient orientation in an image, so the image can be better shown in shape and appearance. The HOG feature generation process is as follows: the image is divided into a plurality of small units called "cells", which consist of a block, called "block"; finally, the histograms of the oriented gradients "cell" and "block" were counted.

The rectangular areas to $(20, 40)$ were scaled before the extraction of HOG feature; the process is further described as follows:

1. The gradient value of the horizontal direction and the vertical direction of the image is calculated in Equation (7):

$$\begin{cases} G_x(x,y) = H(x+1,y) - H(x-1,y), \\ G_y(x,y) = H(x,y+1) - H(x,y-1). \end{cases} \tag{7}$$

2. The gradient magnitude and direction of the pixel are calculated in Equations (8) and (9):

$$G(x, y) = G_x(x, y) - G_y(x, y), \tag{8}$$

$$\alpha(x, y) = \tan^{-1}\left(\frac{G_x(x, y)}{G_y(x, y)}\right). \tag{9}$$

3. The image is uniformly divided into several cells, divide the gradient direction into 9 bins, then get the HOG characteristics of the cells.
4. The adjacent cells $(5, 5)$ constitute one block, finally the blocks are normalized to obtain the HOG feature of the block.

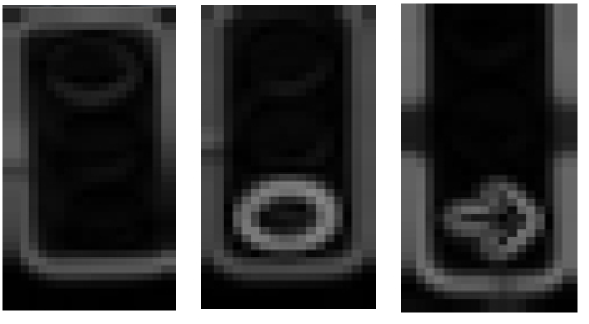

Figure 4. Gradient map

5.2.2 The Block Color Histogram Feature Extraction

Color histogram is a color feature with scaling and rotation invariance that is widely used to describe the proportion of different color areas in a whole image. An ordinary color histogram is known as a type of target characteristics that can extract color information from the target, but cannot effectively express the color distribution in different positions when the statistical range includes an entire area but a part of the local color information is missing. The block color histogram feature was chosen in which the image area to be detected was divided into a small region histogram, so the local color information of the traffic lights could be reflected, and even when the target was partly obscured, the impact of the target color on the histogram was limited, and did not affect other areas. This process is shown below:

1. The Red, Green, and Blue (R, G, and B) components of the pixels of the detected rectangular area are extracted.
2. The current image area to be detected into n units (cell) are then divided; the cell size was 5×5, similar to the HOG feature cell size; at the same time the histogram features of each cell are calculated into three components.

3. The histogram feature of each cell is comprehensively normalized, and the color histogram feature of the image area is obtained for detection.

The HOG features and the block color histogram features were connected into C-HOG features sequentially, as the input feature vector of the SVM.

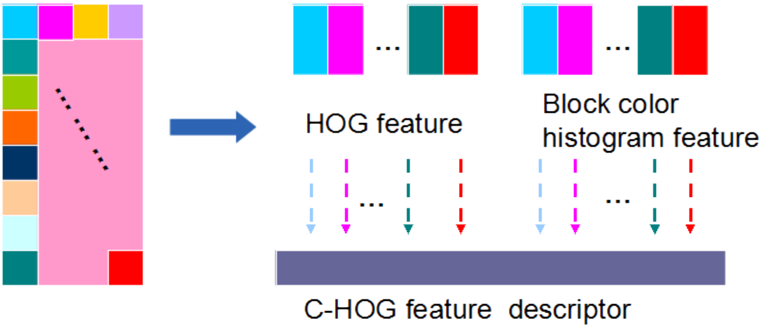

Figure 5. C-HOG features schematic

5.3 SVM Classifier

SVM is a pattern recognition method based on statistical learning, and is widely used in areas of pattern recognition, such as pedestrian detection and face recognition. It is a common method of machine learning when the training samples are limited.

The C-HOG features were extracted using a linear SVM classification which was coded in the C computer language. The penalty coefficient C of the model parameters was set as 0.8 because it has shown the best prediction accuracy through many experiments.

5.3.1 Sample Training

Pictures of traffic lights were collected from the video under actual road conditions; the color of the lights might be red or green, and the shape might be round or arrow. The pictures were collected in different sizes, under different weather conditions and at different times of day when they were taken, so there was enough generalization ability.

3 000 positive samples and 5 000 negative samples were collected for training. We arranged and combined the positive and negative samples to extract the HOG feature and generate the feature vector, which was obtained and trained by the linear SVM. Two kinds of classification were obtained when the training was complete, each of which had four groups. So far, we had obtained the classification support vector and the separating hyper-plane. The weights w, bias b and the HOG eigenvector x extracted from the detection window were set as variable inputs of the SVM classification hyper-plane.

5.3.2 Classification and Identification

For the new detection window, the next step was to extract the C-HOG features. After the feature extraction, classifiers were used to find the corresponding recognition results.

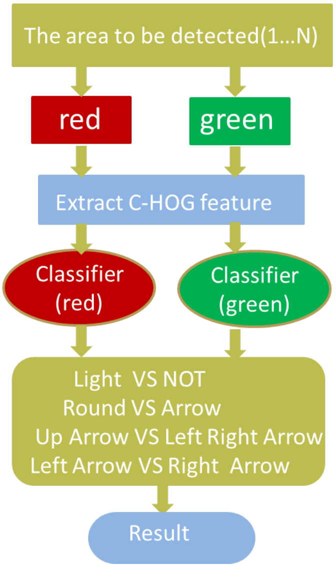

Figure 6. Schematic of the classifier identification

In this research red lights have four classifiers and green lights also have four classifiers. If there is a red right arrow light to be identified, then only the four classifiers of red light can be selected based on its red color. The positive samples from the first classifier were all the red lights; negative samples included any pictures in addition to the traffic lights. After identifying using the first classifier, the current area could be identified as a traffic light. Positive samples in the second classifier included a collection of all circular red lights; negative samples included the collection of all arrow lights. This region was not a red round light determined by the second classifier. Positive samples in the third classifier included a collection of all the red straight arrow lights; negative samples included the collection of left arrow lights and right arrow lights. Positive samples in the fourth classifier consisted of a collection of all the red left arrow lights; negative samples consisted of the collection of right arrow lights. The third classifier determines this region as not including the straight arrow lights. Because the third classifier did not identify the type of the current light, the fourth classifier was used. The fourth classifier provided the final results. However, if the current region was a red round light, the algorithm was set to end after the second classifier recognition. This includes the identification process for green lights as well.

5.4 Discrimination of Location Information

For an intelligent vehicle, information relating to color and shape is not enough; the vehicle also needs to identify the relative position of the traffic light. When there is a traffic light in the area, its color and shape can be determined through the judgment of the SVM; the center coordinate of the color region is also known. The location information will be certified once the size of the horizontal centers has been sorted according to size. The location information, type, and color of the traffic lights all provide a reliable basis for the vehicle's decision making process.

The diagram in Figure 7 is an example to illustrate the process. Classifiers can identify the color and shape of the traffic light. The result of red left arrow light and green circle light can be obtained from classifiers. At the same time, coordinates of two lights can be known. After analyzing the information provided by the classifiers and the coordinate information from the traffic light, it was concluded that red left-arrow light is on the left; the green circle light is on the right. The results would then be sent to the intelligent vehicle's decision-making system. An intelligent vehicle also can get lane information through other devices. If we assume that the intelligent vehicle is located in the left lane, we do not consider right lane information; because the left light is red, the intelligent vehicle should stop before the stop line.

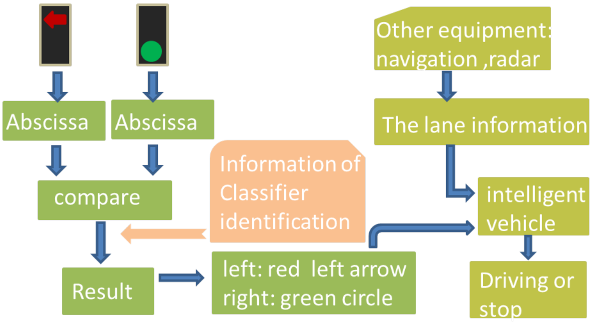

Figure 7. Auxiliary function of location information

6 EXPERIMENTAL SETUP

The equipment used in the experiments is shown in Figure 8. For this paper the PIKE-F100 industrial digital camera was chosen, with a resolution of $1\,000 \times 1\,000$ and a frame rate up to 60 fps, using the data interface 1394. This camera was installed on the vehicle's front window glass 1.2 m above the ground level. During the test, the camera's frame rate was set to 30 fps to obtain a real-time image in front of the vehicle. The IPC we used is from the GEMOTECH series with an i7 processor, 2.67 GHz CPU, 3.17 GB of available memory; it also has excellent resistance to high temperatures and a capability for anti-electro-magnetic interference. Two types of traffic lights were customized and chosen, which were located at the southwest and the northeast side of the testing ground, with exactly the same configuration as the actual road.

Figure 8. Schematic of equipment in test field

7 EXPERIMENTAL RESULTS AND DISCUSSION

The classification accuracy rate is typically the indicator to evaluate the performance of the SVM classifier. There are 4 conditions of the classifier predictions; a total of four types of situations are referred to as: TP, FN, FP, TN. 3 000 samples of the red left arrow lights have been chosen as examples to perform the test. In this paper, the ten-fold cross-validation method was adopted, and made 10 times to calculate average to obtain the final accuracy, which was 99.36 %.

7.1 Test of the Basic Conditions

After the classifier's accuracy test, the classifier can be seen as having the condition with further experiments. The program was loaded on an intelligent vehicle before the real road test. Many repetitive experiments were carried out on the testing ground with the basic conditions, and good experimental results were achieved. The diagram of the testing ground is in Figure 9.

Traffic lights were located at the northeast and the southwest corners of the playground. The playground is the intelligent vehicle test site seen in Figure 9.

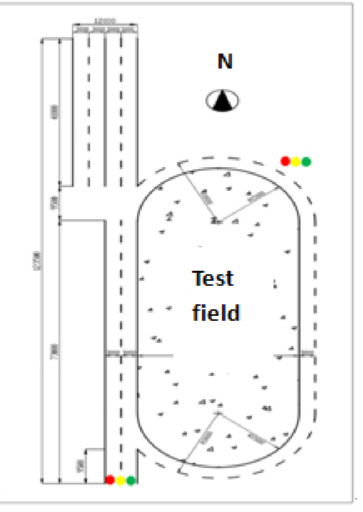

Figure 9. Schematic of the basic conditions

The intelligent vehicle completes 2 cycles of traffic light identification when driving a lap around the playground. The vehicle used in these experiments is illustrated in Figure 8 and the experimental results are shown in Figures 10, 11 and 12.

The basic testing ground is found outdoors because it meets the requirements for different weather conditions. At the same time, the test lane one side reached 123.5 meters, allowing it to meet diverse requirements for distance. Color features are very sensitive to illumination changes, especially in different weather conditions. Methods proposed by Chiang et al. [26] were found under different weather conditions, so the experiments were also carried out under different conditions, including sunny, overcast days and in the fog. Most methods require the test to be performed under a normal light environment during sunny days. Cai et al. [31] performed the test under different light conditions and light directions. The test environment for this paper includes the following: the sun at its highest in front of the camera and behind the camera. The two states in this paper are called non-normal light; in the non-normal state, the image will be overexposed and colors will appear distorted, even the color on edge of the light will be white, similar to a blooming effect. This paper performs the test under conditions of non-normal light to verify the accuracy of the algorithm. For an intelligent vehicle, if the light is red it is necessary to stop before the stop line. The stop line is an indicator in an intersection, and the state of the lights at intersections can determine whether an intelligent vehicle will go or not. The test range is from 0 to 70 meters from the stop line; a test was performed every 10 meters within this range, equaling a total of eight tests. This paper includes statistics on traffic light recognition results at intersections; the changes in the accuracy ratio with distance can be found below.

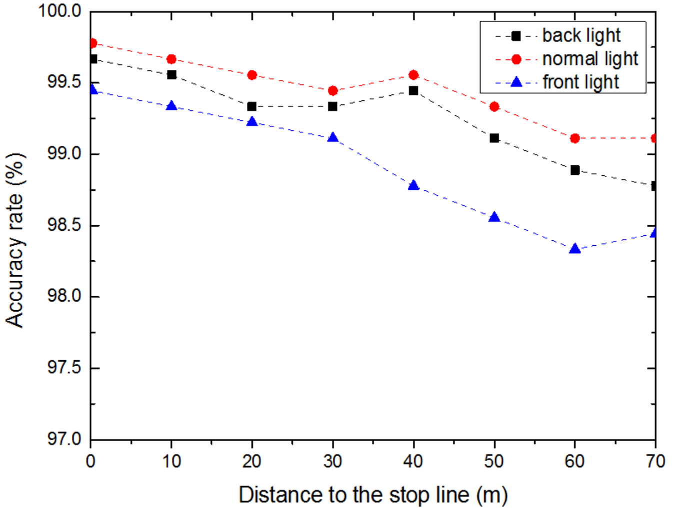

Figure 10. Schematic of accuracy rate on sunny day

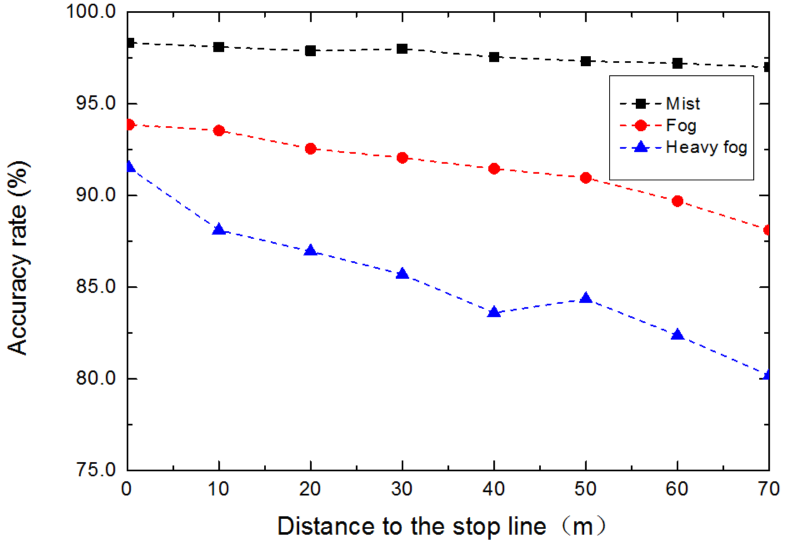

Figure 11. Schematic of accuracy rate on foggy day

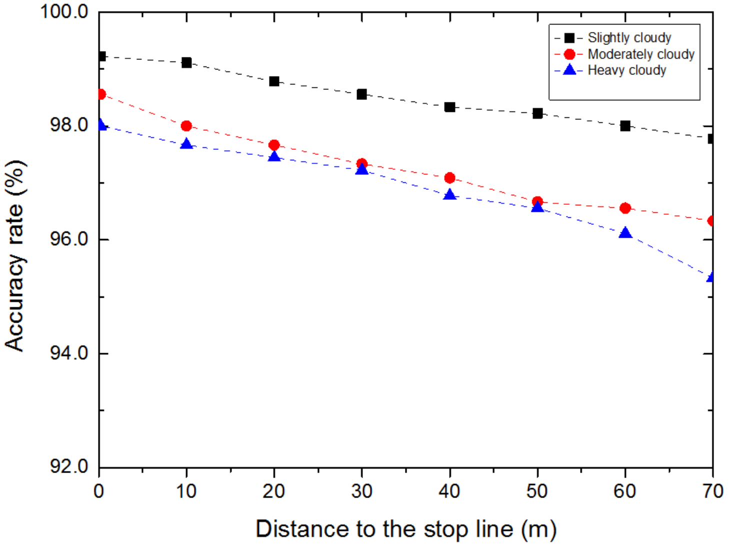

Figure 12. Schematic of accuracy rate on cloudy day

It was difficult to distinguish in the initial training because the environment of the test ground was relatively simple and there were no-signal samples similar to the traffic lights. So it had to be retrained to find a negative sample set. The retraining process consisted of making some pictures for negative sample training. The pictures identified in the pictures that as lights were not in fact lights. The classifiers obtained had a higher accuracy than before the training.

The accuracy rate was expected to drop in the towards-light environment settings because of the halo phenomenon; the center color distortion made the image of the traffic lights white. At the same time as the camera was installed inside the vehicle, there were colorful lights in the images as a result of reflections in the front glass. The accuracy rate was a little higher in the backlight environment. The best test results were under normal light, when compared to the "backlight" and the "towards-light" settings.

On the foggy day, the information regarding color and shape was relatively clear at light-fog conditions; however, heavy fog caused the recognition rate to decline significantly because the color information was weak and the outline was not clear. The overall recognition rate was low on the cloudy day compared to the sunny day because the poor light led to a low quality of the picture obtained by the camera and the details of the images information were missing, but the overall recognition rate could meet the requirements.

7.2 Actual Road Testing

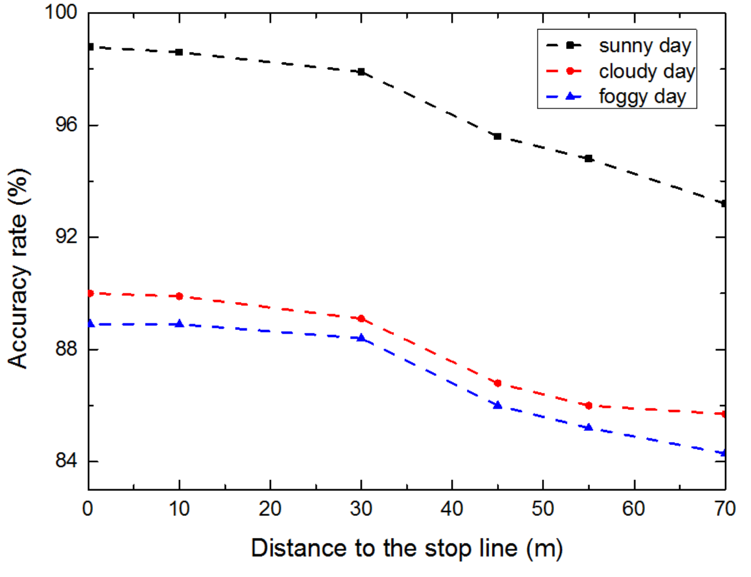

Figure 13. Schematic of accuracy rate on actual road

From the results of the basic test environment above, the accuracy was higher when the light was normal on sunny day, compared to cloudy and the foggy days. The tests on actual urban roads under the 3 conditions above were carried out. The test began when distance from the stop line was 70 m. The results are found below: Figure 13 shows that the accuracy rate of the simplest environment is lower than the rate of the actual scene, because the background was more complicated in the actual scene. There were more types of interference, the instances of which were too numerous to train in negative samples. Meanwhile, a series of experiments were performed on all kinds of traffic lights when the light was normal on a sunny day. The accuracy rate is below.

The experiment data illustrated in Figure 14 was analyzed and it showed weak color information which explained why some pictures were missed. The color area could not be extracted because there was no identifiable area. This was caused by the fact that the round and arrow lights were confusing when the shape of the traffic light was not clear. A small number of non-signal objects were also misjudged. Overall, we found good results on the actual road in clear weather and when the distance from the lights was adequate.

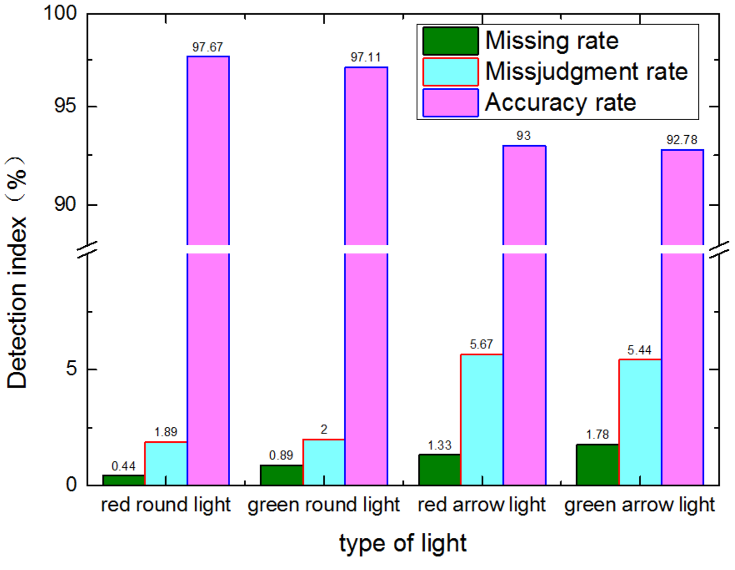

Figure 14. Schematic of the total accuracy rate

7.3 Processing Time

As can be seen from the data of the simplest test environment above, the accuracy was high when the light was normal on a sunny day, a bit cloudy day and in a foggy day. We carried out the test on actual urban roads in the 3 conditions above. The test began when distance from the stop line was 70 m. The results are below.

7.4 Comparison with Other Methods

The data in Table 1 is from the referenced literature; referenced literature has been marked out.

Method	Type of Light	Red Light	Green Light
Color feature and structural information [20]	Circle	89 %	89 %
LBP and SVM [26]	Circle	91.67 %	96.84 %
C-HOG and SVM	Circle	97.67 %	97.11 %
Wavelet transform and nearest classifier [27]	Arrow	92.72 %	92.72 %
C-HOG and SVM	Arrow	93 %	92.78 %

Table 1. Comparison of the accuracy rate with different methods

Table 1 compares the accuracy of the different methods. The vast majority of previous research did not provide accuracy of recognition, so there are fewer methods in Table 1. The LBP and SVM method identifies circular types of lights, its recognition accuracy for red lights is 91.67 %, and its recognition accuracy for green lights is 96.84 %. Due to the fact that fewer papers used classifiers to identify arrow lights, the recognition accuracy of the method given at present has not been found. Compared with other algorithms, the recognition rate for the round red light improved, the recognition rate for other types of lights increased slightly. The algorithm in this paper also has the following advantages:

The tests were completed in an intelligent vehicle that was modified by the China Beijing Automotive Group from two new models: C70 and C30. The vehicle meets the conditions for actual environmental testing. The test time for the basic test environment accumulated more than 300 hours; the actual test environment accumulated more than 500 hours. It was able to obtain real-time processing of each frame and had a good robustness.

The processing time per frame of the algorithm was about 90 ms; the fastest reached 72 ms. The algorithm met the real-time requirements of the intelligent vehicle in an actual environment.

The actual testing environment for the algorithm was an urban road in Beijing. The traffic was heavy and complicated, with diverse varieties of intersection lights. During the testing process, although there were occasional instances of missed detection and misjudgment, the vehicle passed through the intersections safely with the help of mistake exclusion mechanisms. The accuracy rate testing on the complicated road is shown in the Figure 13.

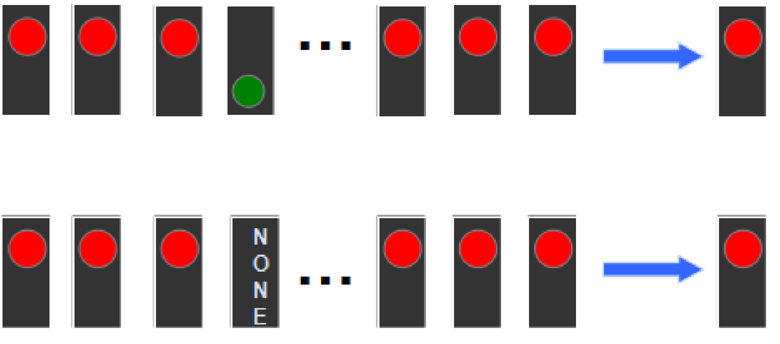

Figure 15. Schematic of the error exclusion mechanism

8 CONCLUSIONS

This paper presents an algorithm for detection and identification of traffic lights based on C-HOG features and SVM. A new feature called C-HOG was proposed for identification tasks. This feature is a combination of HOG features and color features. Machine learning methods were brought into the field of traffic light detection and identification to identify the different types of lights. In this paper, the steps were to extract the C-HOG features of a traffic light, and then use SVM to recognize those features. Experimental results show that the proposed method can detect a traffic light in a reasonable time with an accuracy level that is superior to other methods. In order to further improve the method, there are two improvements to make in the future. The first is how to accurately identify yellow lights. The second is to expand the application scope of the algorithm, to ensure that the algorithm can accurately identify the lights at night. Another research area of this paper in the future is to use tracking algorithm to identify a traffic light.

Acknowledgment

This project was supported by The National Natural Science Foundation of China (Nos. 61571045, 61372148), Beijing Natural Science Foundation (4152016) and The National Key Technology R & D Program (2014BAK08B02, 2015BAH55F03).

REFERENCES

[1] Lu, K. H.—Wang, C. M.—Chen, S. Y.: Traffic Light Recognition. Journal of the Chinese Institute of Engineers, Vol. 31, 2008, No. 6, pp. 1069–1075, doi: 10.1080/02533839.2008.9671460.

[2] Fan, B.—Lin, W. Y.—Yang, X.: An Efficient Framework for Recognizing Traffic Lights in Night Traffic Images. IEEE International Congress on Image and Signal Processing and IEEE International Conference on BioMedical Engineering and Informatics (CISP-BMEI), Chongqing, 2012, doi: 10.1109/CISP.2012.6469638.

[3] Wu, Y.—Zhang, X.—He, B.: Traffic Signal Recognition Method Based on Image Processing. Transport Information and Safety, Vol. 29, 2011, No. 3, pp. 51–54.

[4] Li, J.: An Efficient Night Traffic Light Recognition Method. Journal of Information and Computational Science, Vol. 10, 2013, No. 9, pp. 2773–2781.

[5] Chung, Y. C.—Wang, J. M.—Chen, S. W.: A Vision-Based Traffic Light Detection System at Intersections. Journal of Taiwan Normal University: Mathematics, Science and Technology, Vol. 47, 2002, No. 1, pp. 67–86.

[6] Zhishuai, Z.: Railway Signal Identification and Ranging Technology Based on Color Image Processing. Wuhan University of Technology, Wuhan, 2010.

[7] Choi, J.—Ahn, B. T.—Kweon, I. S.: Crosswalk and Traffic Light Detection via Integral Framework. 2013 19th Korea-Japan Joint Workshop on Frontiers of Computer Vision, IEEE, 2013, pp. 309–312, doi: 10.1109/FCV.2013.6485511.

[8] Gong, J.—Jiang, Y.—Xiong, G.: The Recognition and Tracking of Traffic Lights Based on Color Segmentation and CAMSHIFT for Intelligent Vehicles. 2010 IEEE Intelligent Vehicles Symposium, USA, 2010, pp. 431–435, doi: 10.1109/IVS.2010.5548083.

[9] Yu, C.—Huang, C.—Lang, Y.: Traffic Light Detection During Day and Night Conditions by a Camera. 2010 IEEE 10th International Conference on Signal Processing, 2010, pp. 821–824, doi: 10.1109/ICOSP.2010.5655934.

[10] Diaz-Cabrera, M.—Cerri, P.—Sanchez-Medina, J.: Suspended Traffic Lights Detection and Distance Estimation Using Color Features. 2012 15th International IEEE Conference on Intelligent Transportation Systems, 2012, pp. 1315–1320, doi: 10.1109/ITSC.2012.6338765.

[11] Peng, W.—Guangzi, Z.—Kailiang, S.: A New Image Recognition Technology Based Recognition Algorithms Lights. Ordnance Industry Automation, 2009, pp. 73–75.

[12] Yelal, M. R.—Sasi, S.—Shaffer, G. R.: Color-Based Signal Light Tracking in Real-Time Video. Proceedings of the IEEE International Conference on Video and Signal Based Surveillance, 2006, p. 67, doi: 10.1109/AVSS.2006.34.

[13] Siogkas, G.—Skogras, E.—Dermatas, E.: Traffic Lights Detection in Adverse Conditions Using Color, Symmetry and Spatiotemporal Information. International Conference on Computer Vision Theory and Applications (VISAPP 2012), 2012, pp. 620–627.

[14] Gu, M.—Cai, Z.—Yi, L.: Recognition of Traffic Lights Using Round Rate and Color Histogram. Computer Engineering and Design, Vol. 33, 2012, No. 1, pp. 243–247.

[15] Shen, Y.—Ozguner, U.—Redmill, K.: A Robust Video Based Traffic Light Detection Algorithm for Intelligent Vehicles. Intelligent Vehicles Symposium, IEEE, 2009, pp. 521–526.

[16] Jie, Y.—Xiaomin, C.—Pengfei, G.: A New Traffic Light Detection and Recognition Algorithm for Electronic Travel Aid. Fourth International Conference on Intelligent Control and Information Processing, Beijing, 2013, pp. 644–648, doi: 10.1109/ICICIP.2013.6568153.

[17] Park, J. H.—Jeong, C.: Real-Time Signal Light Detection. 2008 Second International Conference on Future Generation Communication and Networking Symposia, IEEE, 2008, pp. 139–142, doi: 10.1109/FGCNS.2008.35.

[18] Lindner, F.—Kressel, U.—Kaelberer, S.: Robust Recognition of Traffic Signals. 2004 IEEE International Vehicle Symposium, Parma, Italy, 2004, pp. 49–53, doi: 10.1109/IVS.2004.1336354.

[19] Omachi, M.—Omachi, S.: Traffic Light Detection with Color and Edge Information. 2009 IEEE 10th International Conference on Signal Processing, 2009, pp. 284–287.

[20] Omachi, M.—Omachi, S.: Detection of Traffic Light Using Structural Information. 2010 IEEE 10th International Conference on Signal Processing, 2010, pp. 809–812.

[21] Ninglie, W.—Hongxia, X.—Long, Zh.: A Machine-Vision-Based Approach to Identify Traffic Lights. Fujian Computer, 2010, pp. 12–13.

[22] Cheng, X.—Naiqiang, T.—Yan, L.: Recognition Algorithm of Real-Time Traffic Lights Based on Lab Color Space and Template Matching. Computer Applications, 2010, pp. 1251–1254.

[23] de Charette, R.—Nashashibi, F.: Real Time Visual Traffc Lights Recognition Based on Spot Light Detection and Adaptive Traffic Lights Templates. 2009 IEEE Intelligent Vehicles Symposium (IV), Xi'an, China, IEEE, 2009, pp. 358–363, doi: 10.1109/IVS.2009.5164304.

[24] Iwasaki, S.—Premachandra, C.—Endo, T.—Fujii, T.—Tnimoto, M.—Kimura, Y.: Visible Light Raod-to-Vehicle Communication Using High-Speed Camera. Proceedings IEEE Intelligent Vehicle Symposium, 2008, pp. 13–18.

[25] de Charette, R.—Nashashibi, F.: Real Time Visual Traffic Lights Recognition Based on Spot Light Detection and Adaptive Traffic Lights Templates. Intelligent Vehicles Symposium, IEEE, 2009, pp. 358–363, doi: 10.1109/IVS.2009.5164304.

[26] Chiang, C. C.—Ho, M. C.—Liao, H. S.: Detecting and Recognizing Traffic Lights by Genetic Approximate Ellipse Detection and Spatial Texture Layouts. International Journal of Innovative Computing, Information and Control, 2011, pp. 6919–6934.

[27] Cai, Z.—Gu, M.—Li, Y.: Real-Time Arrow Traffic Light Recognition System for Intelligent Vehicle. The 16th International Conference on Image Processing, Computer Vision, Pattern Recognition. New York, IEEE Society, 2012, pp. 848–854.

[28] Kim, H.-K.—Shin, Y.-N.—Kuk, S.-G.—Park, J. H.—Jung, H.-Y.: Night-Time Traffic Light Detection Based on SVM with Geometric Moment Features. World Academy of Science, Engineering and Technology, 2013, pp. 454–457.

[29] Fairfield, N.—Urmson, C.: Traffic Light Mapping and Detection. 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 2011, pp. 5421–5426, doi: 10.1109/ICRA.2011.5980164.

[30] Levinson, J.—Askeland, J.—Dolson, J.: Traffic Light Mapping, Localization, and State Detection for Autonomous Vehicles. 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 2011, pp. 5784–5791, doi: 10.1109/ICRA.2011.5979714.

[31] Cai, Z.—Li, Y.—Gu, M.: Real-Time Recognition System of Traffic Light in Urban Environment. 2012 IEEE Symposium on CISDA, 2012, pp. 1–6.

[32] Dalal, N.—Triggs, B.: Histograms of Oriented Gradients for Human Detection. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2005, pp. 886–893, doi: 10.1109/CVPR.2005.177.

Xuanru Zhou is now a master at Beijing Union University and her main research direction is graphic image processing.

Jiazheng Yuan received his Ph.D. in computer science from Beijing Jiaotong University in 2007. He is now Professor in computer application technology and engineering, Beijing Union University, Beijing, China. His current research interests include graph and image processing, digital processing of cultural relics, digital museum.

Hongzhe Liu received her Ph.D. in computer science from Beijing Jiaotong University in 2012. She is now Associate Professor in computer application technology and engineering at Beijing Union University, Beijing, China. Her current research interests include digital museums, semantic calculating and image processing.