Journal of Computers Vol. 28, No. 2, 2017, pp. 197-205
doi:10.3966/199115592017042802015

Traffic Light Recognition Based on Prior Knowledge and Optimized Threshold Segmentation

Xiao-Ping Du[1, 2], Hui Xiong[1,*], and Xiao-Fei Li[3,*]

[1] School of Software, Beihang University,
Beijing 100191, China
{xpdu, tomxiong}@buaa.edu.cn

[2] Key Laboratory of Advanced Manufacturing Technology for Automobile Parts, Ministry of Education,
Chongqing University of Technology,
Chongqing 400054, China

[3] State Key Laboratory of Automotive Safety and Energy, Tsinghua University,
Beijing 100084, China
lixf11@mails.tsinghua.edu.cn

Received 31 December 2015; Revised 03 June 2016; Accepted 03 July 2016

Abstract. Traffic light recognition (TLR) is an important component of vehicular vision system for an autonomous vehicle, which has been extensively studied. However, existing algorithms cannot guarantee real time and effectiveness at the same time. Thus we propose an automatic traffic light recognition method based on prior knowledge and optimized threshold segmentation. The method uses the differential GPS (DGPS) and designed labeling software to obtain the prior knowledge of traffic lights. Then an optimized threshold segmentation combining some custom constraints for traffic light detection is introduced. Finally, a tracking algorithm with the prior knowledge for logic validation is combined, to get more accurate light detection results. Experimental results show that our proposed method can achieve the detection rate of 99.4%, with average 31ms processing time per image.

Keywords: intelligent vehicle, prior knowledge, threshold segmentation, traffic light recognition

1 Introduction

Traffic light recognition (TLR), as an important part of intelligent vehicle navigation and advanced driver assistance systems (ADAS), has been researched for many years. Although extensive investigation and significant progress have been made [1-3], TLR is still a challenging visual task due to the large variability in appearance, cluttered background. What's worse, different countries may have different standards for traffic lights (TLs), such as size, shape and numbers. In this paper, we focus on vision-based traffic light detection, so infrastructure to vehicle (I2V) technology is not discussed. The recognition of traffic lights can be used to warn fatigue driving and illegal driving, and make it possible for the color blindness and color weakness to drive a car. And it will make driverless technology in the urban driving alive further forward. Therefore in this paper we discuss the novel proposed method of TLR, which can be extended and integrated easily.

* Corresponding Author

2 Related Work

The researches of traffic light recognition can be divided into four parts: the conventional image processing [4, 12-15], computer vision and machine learning [5-11], based on a priori map information [16-17] and the integration of transcendental map and machine learning [19].

The traditional image processing technology is mainly based on the color and shape features of traffic lights for traffic light recognition. The study of color features is mainly based on the existing color model space, such as HSV space [12], RGB color space [13] and CIE color space [14]. The original image captured by camera is RGB format, while RGB color components are affected easily by the presence of illumination variation. On the other hand, the conversion from RGB to HSV space or CIE space is nonlinear transformation, time-consuming, and hence the transformation is difficult to meet real-time performance of on-board vision. The study of shape features mainly focuses on the circular area [12], the rectangular board [15] and the straight shape support bar [4], among those features, the circular feature is not available for arrow lights, and the straight shape support bar is low and fine vulnerable to background interference. Machine learning methods including Adaboost [5], Bayesian classifiers [6], and support vector machines (SVMs) [7]. These methods, from today's point of view, are considered using hand-codes features such as Haar wavelet [8], and a histogram of oriented gradient (HOG) [9] or scale-invariant features transform (SIFT) in [10]. However, designing such features requires a great deal of time, and is suitable for a particular scenario. So in recent years, the researchers pursue a more powerful machine learning algorithms in the TLR system, among which the convolutional neural network (CNN) [11] comes into being. CNN owns the specialty of self-learning feature extraction, which can deal with much more data and deeper models.

Using a prior map to detect TLs in [16-17], can obtain the region of interest (ROI) to improve the recognition accuracy. However, this method requires to draw the map information of TLs, which not only needs to collect video capture and GPS information, but also needs much manual processing. Hence it has limited applicability. A fusion of priori map and machine learning [19] can express richer information than a single method, improve the robustness of TLR system to illumination variation, while the real-time needs to be further improved.

In this paper, a new method combining prior knowledge technique and optimized threshold segmentation theory is proposed for detecting traffic lights in a clustered background. The method uses differential GPS technology to mark the location of TLs in advance, at the same time the number and shape of traffic lights can be got. After the TLR method is started by the GPS-based locating information, calibration information of the camera is used to estimate ROI for traffic lights. And then a segmentation combining optimized empirical threshold and some custom constraints is deployed. Finally, the color information is used for the light classification. At last combining the simple tracking algorithm with the prior knowledge for logic validation. Tests show that the method has a relatively high recall rate and averaged detection time of 31ms to meet real-time requirements.

3 The Overview of Our Method

Recognition of targets in this paper are traffic lights of domestic cities, the types of TLs contain circular lights, left arrow lights, straight arrow lights and right arrow lights, the colors of TLs contain red and green. TLR of this paper includes the acquisition of prior knowledge, the detection of TLs and the verification of TLs. And the detection of TLs needs to set ROI, segment image, filter TL candidate region; the verification of TLs consists of color recognition and tracking decision. The overall flow chart of TLR in this paper is shown in Fig. 1.

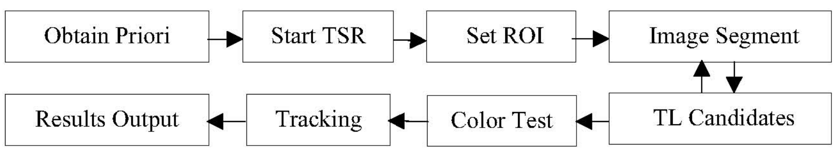


Fig. 1. The overall flow chart of TLR

4 Acquisition of Priori Knowledge

Traffic lights exist only at a particular location, starting TLR method in real time will take up system resources, which is not conducive to the real-time performance of autonomous vehicles. In this paper, we use base station and mobile station to achieve differential GPS (DGPS), with a centimeter level positioning accuracy. On the other hand, the use of inertial navigation instrument ensures that DGPS still have signal outputs by internal inertial senor, even when the GPS signal is lost. Just start TLR method when approaching the location of TLs to improve driverless real-time capability.

In the acquisition of autonomous navigation routes, simultaneously uses designed labeling software to collect the information of traffic lights, such as the location (longitude and latitude), number, shape (circular or arrow-shaped), direction (straight or left or right) and integrity (complete or broken), without the need of extra manual handling. In the following custom constraint verification, the full use of prior information can effectively avoid TL false detection, reduce TL loss detection, and sequentially improve the reliability and safety of an autonomous vehicle.

5 Detection of Traffic Lights

TLR method which detects TLs from a static picture or a dynamic video involves three stages: ROI extraction, image segmentation and TL candidate filtering.

5.1 ROI Extraction

Reasonable extraction of ROI can improve the processing speed of TLR method, effectively eliminate background interference. In this paper, the original input is RGB image obtained by AVT Mako monocular camera after opening TLR method based on positional information of TLs. The RGB image resolution is 1292*964, and the camera is calibrated by the checkerboard method. Since monocular camera is mounted on the right of the windshield, as shown in Fig. 2 (a), and the ROI of traffic lights usually appears in the top right corner of the picture, as shown in Fig. 2 (b).

(a) (b)

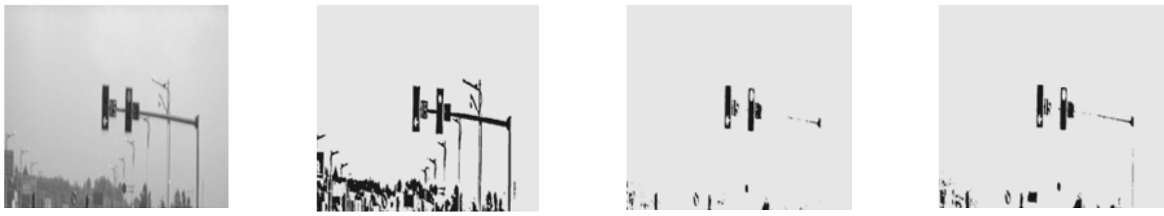
Fig.2. (a) The installation of the camera, (b) original image

There are two common processing means of setting ROI: 1) first compress, then set ROI; 2) set ROI directly. During the experiment, make a comparison of the two methods. Specifically, the first method compress the input image resolution 1292*964 into 720*480, and then truncate the upper right as ROI; the second method truncate directly the upper right of the 1292*964 input image as ROI. Experimental results show that the latter is more suitable for setting ROI, because TLR verification algorithm has a custom constraints, such as the minimum length and width constraints of candidate rectangle is 20 pixels, and candidate rectangle from the first method dose not satisfy this constraint. This paper uses the second method to set ROI.

5.2 Image Segmentation

This paper uses optimized single threshold segmentation to separate traffic light panel and foreground. The new segment method is based on the number of traffic lights acquired in priori knowledge, combining the method of empirical threshold method with the finite corrosion and expansion. The effect of threshold segmentation depends on the selection of threshold value, in order to determine the better threshold value, it is necessary to preprocess the ROI before threshold segmentation, such as gray-scale conversion, smoothing processing, histogram equalization and so on. OpenCV corresponding algorithm used in the experiment to make a comparison of the effects of exclusive use of pre-processing methods and combined use of them, concluded that gray-scale conversion is essential, while smoothing and histogram equalization has little effect on the result of segmentation, also time consuming. Due to the use of the camera with automatic exposure, automatic gain, taking into account the importance of driverless real-time, only uses gray-scale conversion of ROI in the preprocessing stage.

The common image segmentation algorithms have empirical threshold, average gray threshold, adaptive threshold and Ostu threshold [18], the latter two are more used. Tests consider cloudy and sunny conditions separately, contrasting proposed optimized threshold segmentation method with adaptive threshold and Ostu threshold method. In Fig. 3 and Fig. 4, (a) the ROI diagram, (b) results of adaptive threshold segmentation method, (c) results of Ostu threshold segmentation method, (d) results of optimized threshold segmentation method.



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

Fig. 3. The results in the cloudy day

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

Fig. 4. The results in the sunny day

Results show that in the cloudy day the effect of adaptive threshold method is modest, Ostu method can segment TL panel better and faster; in sunny, especially in strong light condition, both of adaptive threshold and Ostu threshold method are ineffective, however the empirical threshold method, combining with finite erosion and dilation (namely the proposed optimized empirical threshold method) has effective segmentation both in cloudy and sunny conditions.

5.3 Candidate Region Filtering

To avoid interference on TLR, firstly our proposed method uses optimized empirical threshold to segment ROI, extracts the contour of connected blocks, and then uses custom constraints one to filter enclosing rectangle, deleting regions that are not satisfied with the shape feature of panel obviously, so as to realize curve fitting of connected edges. Finally we use custom constraints two to filter and fit polygon further to obtain a very low false detection rate of TLs candidate regions.

Assume that the image has N enclosing rectangles of connected blocks, denoted by Ri, $i = 1, \ldots, N$, custom constraints one is defined as

$$\mathrm{Bool}(R\ i)=\begin{cases}1,\ \mathrm{CV\_IS\_SEQ\_HOLE}(Ri)\cap(Hi\ \rangle\ 20\cap Wi\ \rangle\ 20)\cap\\ \quad(\max(Hi,Wi)/\min(Hi,Wi)\ \rangle\ 1.45\\ 0,\ \mathrm{else}\end{cases} \quad \textbf{(1)}$$

Where H_i is the height of R_i, W_i is the width of R_i, CV_IS_SEQ_HOLE(Ri) represents whether there are holes in *Ri*, if it is equal to true, the holes are on behalf of TLs.

Further polygons filtered out by custom constraints one must satisfy custom constraints two:

$$Ratio_{area} = Area(Quadi)/Area(R_i) \quad \textbf{(2)}$$

$$Bool(Contour_i)=\begin{cases}1,\ Bool(R_i)\ \cap(isContourTotal(\ 4)\cap\\ \quad \mathrm{isConvex}(Contour_i)\cap Ratio_{area}\rangle 0.5)\\ 0,\ else\end{cases} \quad \textbf{(3)}$$

Where $Quad_i$ is quadrangle, $Ratio_{area}$ is the area ratio of $Quad_i$ and R_i. Specifically custom constraints two requires that the sides of polygon is four, quadrangle is convex and $Ratio_{area}$ is bigger than a given threshold.

After the above filtering process, the TL candidate regions obtained have very low false positive rate, and then candidate regions are sorted from left to right according to the TLs are in the same horizontal line, further eliminating the false detection. Use statistical traffic lights color thresholds in advance (described in the next section) to verify color and exclude non-TLs panel, and finally contrast the number of remaining candidate regions with the number of TL in prior knowledge, denoted as *detect_num* vs *know_num*. There are three contrasting results:

(1) *detect_num* = *know_num*, directly the next step to verify;

(2) *detect_num* < *know_num*, indicate that exist loss detection, then at most go through two erosion and one dilation in sequence to extract the contour of connected blocks again, still filter TL candidate regions according to custom constraints. The two erosion in turn uses crisscross nuclear and rectangular nuclear corrode white areas, dilate black panels. If empirical threshold segmentation plus one erosion can guarantee *detect_num* = *know_num*, stop erosion and dilation, directly the next step to verify, otherwise a second erosion step. In TL candidate filtering step, up to two erosion and one dilation;

(3) *detect_num* > *know_num*, then sort the candidate regions by the vertical distance from the upper left corner, choose the first know_num for the next verification step.

6 Verification of Traffic Lights

In this section, traffic lights that have been detected are classified further according to the color information. And then accurate lights detection results can be acquired by combining the simple tracking algorithm with the prior knowledge for logic validation.

6.1 Color Recognition

Considering the real-time performance of intelligent vehicle, the paper directly verifies color in RGB color space. The selected areas by TL candidate region filtering saved as RGB images. Moreover regard the RGB images containing TL as positive samples, and regard the RGB images no TLs or containing false detection area and original RGB images as negative samples. Positive and negative samples as statistical samples of traffic light color thresholds by the following steps: firstly extract B, G, R three-channel components; then calculate the sum of pixels of single channel, separately denoted as *BPixels*, *GPixels* and *RPixels*; and then computer the threshold of green lights and red lights. The statistic show that TLs must meet the following constrains:

$$\mathrm{Bool}(TL)=\begin{cases}1,\ GThreshold\ \rangle 0.96069\cap RThreshold\ \rangle 0.98981\\ 0, else\end{cases} \quad \textbf{(4)}$$

Where *GThreshold* = (*GPixels* + *BPixels*) / (2**RPixels*) and *RThreshold* = (*RPixels* + *BPixels*) / (2**GPixels*). In actual testing, statistical traffic light color thresholds can be used to filter out non-TLs better, however have an ineffective effect on the distinction between red and green lights. In practice, use the following constraints to identify the color.

$$Color\ (TL) = \begin{cases} 1, & \text{Bool}\ (TL) \cap GPixels\ \ \rangle\ RPixels \\ 0, & \text{Bool}\ (TL) \cap GPixels\ \ <=\ RPixels \\ -1, & else \end{cases} \tag{5}$$

Where *Color*(TL) = 1 represents the light is green light, *Color*(TL) = 0 represents the light is red light, and *Color*(TL) = -1 represents non-TLs.

6.2 Candidate Region Filtering

Beyond recognizing and identifying TLs from a given frame, tracking aims to avoid loss detection in sequential frame images. In this paper, taking into account the importance of autonomous real-time, a simple tracking algorithm is adopted. If there has loss detection, tracking the detection results of the previous frame, most tracking 5 frames, if still cannot detect the traffic light, the default is passable.

And then makes a comparison of testing results and marked results according to the prior knowledge to perform the logic verification, divided into the following three cases:

(1) TLs are circular, as long as a light can be detected, whether accessible is up to its color;

(2) TLs are circular and arrow-shaped, firstly uses *detect_num* TLs horizontal relative position and *know_num* TLs positional information to make sure that *detect_num* TLs and *know_num* TLs corresponding one by one, then the direction information of *know_num* TLs is extracted from prior knowledge, and finally judges whether the direction that TL indicates can passable according to the color recognition of each TL;

(3) TLs are irregular, for example traffic light is off, not used. The default is passable.

7 Evaluation and Results

The proposed method was validated with several video images collected from monocular camera and some static pictures in different traffic scenes of China, not LaRA [20] or LISA [21] Traffic Light Dataset with the reason that different countries may have different standards for traffic lights, also China has a more complex traffic, and we need consecutive video frames to verify the real-time performance of proposed method. Further among them, we automatically collected and labelled the prior knowledge of TLs online for video images, while we manually labelled the number and the shape of TLs offline for static pictures. To be specific, video images were used to verify the real-time performance and reliability of the method, static pictures in different scenarios were used to verify the robustness of the method. Our method was implemented with Visual Studio on a 2.60 GHz Intel i5 processor, used C++ programming with OpenCV library and needs an average processing time of 31ms per frame to detect TLs.

7.1 Performance Analysis

For evaluating the performance of our proposed method, the well-established methodology used in the PASCAL object challenges [21] is utilized. Based on the analysis above, here we focus on f_n (false negative or TLs loss) and recall rate for performance evaluation of different threshold segmentation algorithms and performance evaluation of own tracking decision, then we focus on false negative, f_p (false positive) and average time for performance evaluation of different methods. Also here we define:

$$\text{Recall}\ Rate = \frac{\text{Frame}Nums * \text{Light}Nums - \sum_{index=1}^{FrameNums} \text{Bool}\,(Loss_{index})}{\text{Frame}Nums * \text{Light}Nums} \tag{6}$$

$$Average\ Time = \frac{\sum_{index=1}^{FrameNums} \text{recognizeTime}\,(Loss_{index})}{\text{Frame}Nums} \tag{7}$$

Where *FrameNums* means the total frames for testing, *LightNums* means the number of TLs per frame, $\mathrm{Bool}(Loss_{index}) = 1$ means that there has no false reorganization on the current frame, or $\mathrm{Bool}(Loss_{index}) = 0$ means that there has false reorganization on the current frame. Besides *recognizeTime* is a function that stands for the TLR processing time per frame.

As described in Table. 1, we can see that our proposed method performs significantly better than the adaptive threshold (Baseline1) and Ostu threshold method (Baseline2) on video 1 and video 2. In video 1, there are 325 frames (*FrameNums* = 325) that TLs change from red to green, and from green to red, involving the shape of arrows and circles (*LightNums* = 2), and there is the interference of digital signal lights. In video 2, there are 369 frames that TLs are both circles, just change from red to green. According to case 2 in the last section, video 1 is more difficult than video 2 to detect TLs which the paper pays attention to.

Table 1. Performance evaluation of different threshold segmentation algorithms on video 1

Algorithm	TLs Loss	Recall Rate
Baseline1	298	54.2%
Baseline2	24	96.3%
Proposed	4	99.4%

We evaluate the benefits of tracking decision by comparing the performance of our proposed method without tracking decision and the one with tracking decision. As shown in Table 2, we can see an increase in accuracy of our proposed method with tracking decision.

Table 2. Performance evaluation of tracking decision

Method	TLs Loss	Recall Rate
Without Tracking	22	96.6%
With Tracking	4	99.4%

In order to further evaluate the performance of proposed TLR, comparing baseline method convolutional neural network [11] in Table 3, we focus on false detection and real-time performance.

Table 3. Performance evaluation of different methods

Method	f_n	f_p
Baseline	6	8
Proposed	4	1

For the same video that TLs changes from red to green, and from green to red, involving different shapes, compared with baseline method, the proposed method has a higher recognition rate and the recognition time is short.

The cause maybe that the detection performance of a convolutional neural network is dependent on the size of the training databases, the quality of samples, and the choice of the parameters of each layer. What's more, the dataset in the paper is limited and has the interference of the digital signal besides the circular and the arrow traffic lights. However, for detecting speed, the traffic light recognition method proposed in this paper is obviously superior to the convolutional neural network.

7.2 Robustness Verification

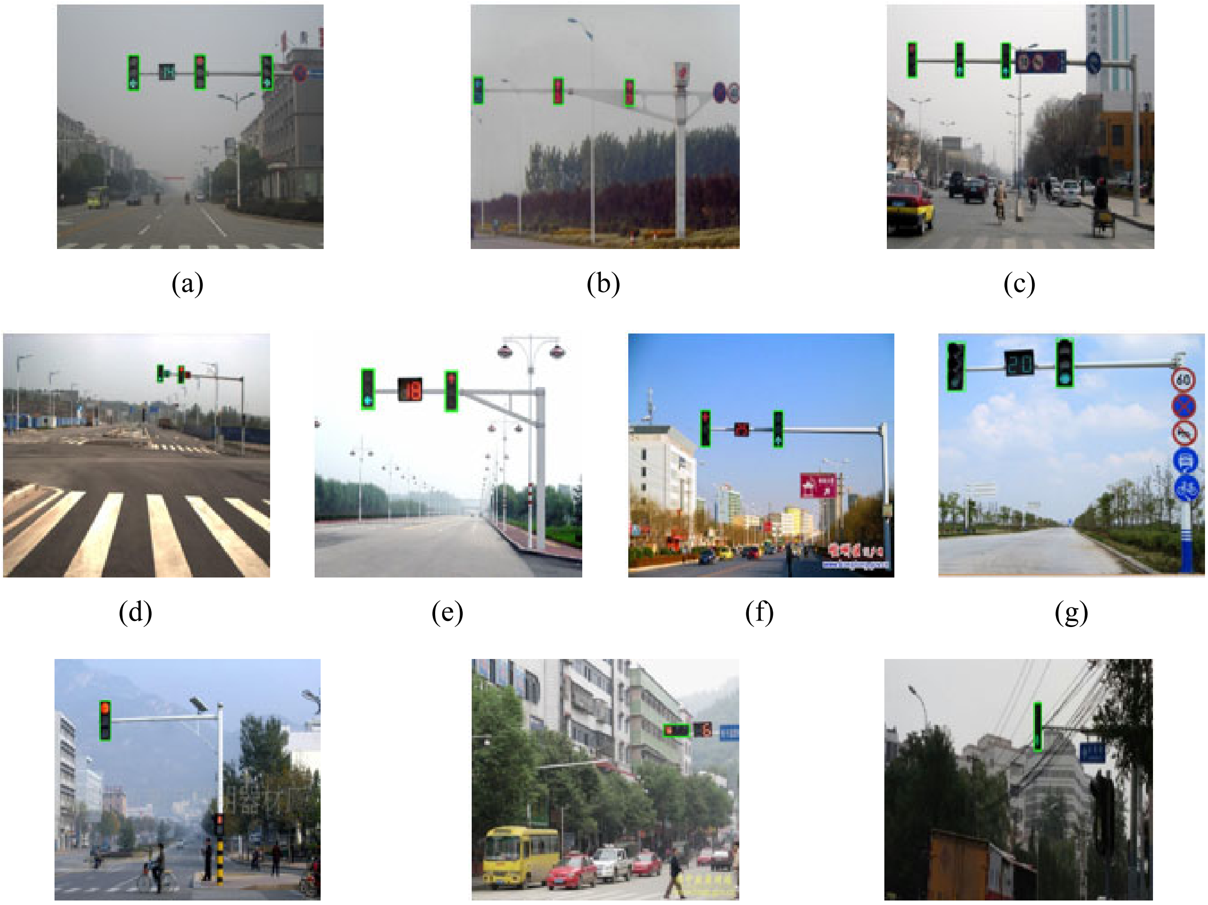
To further verify the robustness of the proposed method, in this section, we select a number of typical static pictures from the cellphone, the Internet and screenshots, involving five cloudy pictures and five sunny pictures, not include a rainy day, because there has no public traffic light database with the prior knowledge of traffic lights. Hence we manually labelled the number and the shape of TLs offline for static pictures. Except pictures whose width is not less than 1292, TLR method normalizes uniformly other pictures into 720*480 and extracts the upper part of them as ROI.

The proposed method can effectively filter out the panel aero of all TLs in Fig. 5. Only the right TL of (g) has wrong color recognition that recognized the green as non-TLs, the possible cause might be the light is too bright that doesn't satisfy the threshold of traffic lights, anyway the output is unaffected with the reason that green and non-TLs both means passable.

8 Conclusion and Future Work

This paper presents a novel method for recognizing traffic lights in a clustered background based on prior knowledge technique and optimized threshold segmentation theory. The method combines prior knowledge with traditional image processing technology to recognize various types of traffic lights. In addition, our proposed method is also convenient to integrate with other methods or sensors.

Experimental results show that the proposed method has good accuracy and real-time capability, which can be used in the vehicular vision system for an autonomous vehicle. The next study consists of the development of a more robust system which can deal with multiple scenes, and the extensive evaluation on a much larger dataset.



(a) (b) (c)

(d) (e) (f) (g)

(h) (i) (j)

Fig. 5. The results of robustness verification

Acknowledgement

This research was supported in part by the Opening Foundation of Key Laboratory of Advanced Manufacturing Technology for Automobile Parts, Ministry of Education, and Tsinghua Changan cooperation projects No.20142001333.

References

[1] M. Diaz, P. Cerri, G. Pirlo, M.A. Ferrer, D. Impedovo, A survey on traffic light detection, in: V. Murino, E. Puppo, D. Sona, M. Cristani, C. Sansone (Eds.), New Trends in Image Analysis and Processing—ICIAP 2015 Workshops. ICIAP 2015. Lecture Notes in Computer Science, vol. 9281, Springer, Cham, 2015, pp. 201-208.

[2] M.B. Jensen, M.P. Philipsen, A. Møgelmose, T.B. Moeslund, M.M. Trivedi, Vision for looking at traffic lights: issues, survey, and perspectives, IEEE Transactions on Intelligent Transportation Systems 17(7)(2016) 1-16.

[3] M.B. Jensen, M.P. Philipsen, C. Bahnsen, A. Møgelmose, T.B. Moeslund, M.M. Trivedi, Traffic light detection at night: comparison of a learning-based detector and three model-based detectors, in: G. Bebis et al. (Eds), Advances in Visual Computing. Lecture Notes in Computer Science, vol. 9474. Springer, Cham, 2015, pp. 774-783.

[4] D.R. Charette, F. Nashashibi, Real time visual traffic lights recognition based on spot light detection and adaptive traffic lights templates, in: Proc. Intelligent Vehicles Symposium, 2009.

[5] X.B. Jin, X.W. Hou, C.L. Liu, Multiclass AdaBoost with hypothesis margin, in: Proc. the 20th International Conference on Pattern Recognition, 2010.

[6] M. Meute, C. Nunn, S.M. Gormer, S. Müller-Schneiders, A. Kummert, A decision fusion and reasoning module for a traffic sign recognition system, IEEE Transactions on Intelligent Transportation System 12(4)(2011) 1126-1134.

[7] J. Greenhalgh, M. Mirmehdi, Real-time detection and recognition of road traffic signs, IEEE Transactions on Intelligent Transportation System 13(4)(2012) 1498-1506.

[8] P. Viola, M.J. Jones, Robust real-time face detection, International Journal of Computer Vision 57(2)(2004) 137-154.

[9] N. Dalal, B. Triggs, Histograms of oriented gradient for human detection, in: Proc. IEEE Conference on Computer Vision and Pattern Recognition, 2005.

[10] D.G. Lowe, Distinctive image features from scale-in-variant key points, International Journal of Computer Vision 60(2)(2004) 91-110.

[11] J.Q. Jin, K. Fu, C.S. Zhang, Traffic sign recognition with hinge loss trained convolutional neural network, IEEE Transactions on Intelligent Transportation System 15(5)(2014) 1991-2000.

[12] M.Q. Gu, Z.X. Cai, Y. Li, Traffic light recognition with circularity and color histogram, Computer Engineering and Design (in Chinese) 1(33)(2012) 243-247.

[13] M. Hu, L.J. Yang, S.D. Zhu, Traffic sign segment based on chromatic aberration of three color-components, Mechanical & Electrical Engineering Magazine (in Chinese) 26(10)(2009) 23-26.

[14] Y. Zhang, G.J. Qin, Detection method of robust traffic sign, Modern Electronic Technique (in Chinese) 32(23)(2009) 177-181.

[15] C. Xu, N.Q. Tan, Y. Liu, Traffic lights recognition algorithm based on Lab color space and template match, Journal of Computer Application (in Chinese) 30(5)(2010) 1251-1254.

[16] J. Levinson, J. Askelan, J. Dolson, S. Thrun, Traffic light mapping, localization, and state detection for autonomous vehicles, in: Proc. IEEE International Conference on Robotics and Automation, 2011.

[17] N. Fairfield, C. Urmson, Traffic light mapping and detection, in: Proc. IEEE International Conference on Robotics and Automation, 2011.

[18] M. Cheriet, J.N. Said, C.Y. Suen, A recursive thresholding technique for image segmentation, Image Processing, IEEE Transactions on 7(6)(1998) 918-921.

[19] V. Joh, K. Yoneda, B. Qi, Z. Liu, S. Mita, Traffic light recognition in varying illumination using deep learning and saliency map, in: Proc. IEEE International Conference on Intelligent Transportation System, 2014.

[20] G. Siogkas, E. Skodras, E. Dermatas, Traffic lights detection in adverse conditions using color, symmetry and spatiotemporal information, in: Proc. International Conference on Computer Vision Theory and Applications, 2012.

[21] M. Everingham, L.V. Gool, C.K.I. Williams, J. Winn, A. Zisserman, The Pascal Visual Object Classes (VOC) challenge, International Journal of Computer Vision 88(2)(2010) 303-338.

[22] M.P. Philipsen, M.B. Jensen, A. Møgelmose, T.B. Moeslund, M.M. Trivedi, Traffic light detection: a learning algorithm and evaluations on challenging dataset, in: Proceeding of 18th IEEE Intelligent Transportation Systems Conference, 2015.