Traffic Light Recognition With High Dynamic Range Imaging and Deep Learning

Jian-Gang Wang, *Senior Member, IEEE*, and Lu-Bing Zhou

***Abstract*— Traffic light recognition (TLR) detects the traffic light from an image and then estimates the state of the light signal. TLR is important for autonomous vehicles because running against a red light could cause a deadly car accident. For a practical TLR system, computation time, varying illumination conditions, and false positives are three key challenges. In this paper, a novel real-time method is proposed to recognize a traffic light with high dynamic imaging and deep learning. In our approach, traffic light candidates are robustly detected from low exposure/dark frames and accurately classified using a deep neural network in consecutive high exposure/bright frames. This dual-channel mechanism can make full use of undistorted color and shape information in dark frames as well as the rich context in bright frames. In the dark channel, a non-parametric multi-color saliency model is proposed to simultaneously extract lights with different colors. A multiclass classifier with convolutional neural network (CNN) model is then adopted to reduce the number of false positives in the bright channel. The performance is further boosted by incorporating temporal trajectory tracking. In order to speed up the algorithm, a prior detection mask is generated to limit the potential search regions. Intensive experiments on a large dual-channel dataset show that the proposed approach outperforms the state-of-the-art real-time deep learning object detector, which could cause more false positives because it uses bright images only. The algorithm has been integrated into our autonomous vehicle and can work robustly on real roads.**

***Index Terms*— Traffic light recognition, autonomous vehicle, high dynamic range imaging, deep learning.**

I. Introduction

TRAFFIC Light Recognition (TLR) locates the traffic light from an image and then estimates the status of the light signal. Automatic detection and recognition of traffic light is an important perception function for Advanced Driver Assistance Systems (ADAS) or an autonomous vehicle. While much effort has been devoted to traffic light recognition, relatively limited attention has been paid to practical traffic light recognition problems. Among the most challenging issues in practical traffic light recognition are computation time, varying lighting conditions, rear lights of the vehicles ahead, confusion with other similar ambient light, low resolution, and occlusion. In this paper, in the premise of ensuring real-time conditions, we are interested in the problem caused by lighting conditions and rear lights or head lights of the vehicles ahead.

The traffic light recognition problem has been extensively investigated. Surveys on traffic light recognition can be seen in [1]–[3]. The traditional approaches can be broadly divided into three categories: template matching, circular extraction, and color distribution. The first category applies the template of red and green lights to the extracted region. In the second category, Hough Transform is applied to detect the circular shape from the image. The third category is mainly color segmentation. Sensitivity to the lighting conditions is a major disadvantage of these approaches.

Traffic light candidates are generated from an image based on color and shape [4]–[7]. The candidates are then pruned by some preprocessing before being fed to the classifier. Color threshold segmentation was performed in the HSV [8] or RGB [9] space followed by pruning based on shape, temporal information, edge and symmetry [8], [9]. An adaptive template matching scheme was proposed to recognize the traffic light states in [1].

Besides image, some prior information, e.g. annotated map and GPS, has been used to find the region of interest (ROI) for improving the detection robustness [10], [11]. Some non-passive methods [12]–[14], e.g. communication between the vehicle and traffic light or car-to-car, are also proposed. As special infrastructures are needed, massive adoption is limited.

Different from existing approaches, in this paper, we use a High Dynamic Range (HDR) camera which has more than one channel corresponding to different exposure values. Given that traffic light detection is naturally a computer vision problem, we aim to extract the light pattern from a dark background. It is clear that the lights can be detected from the lower exposure channel (dark background) much more robustly than from the bright channel. To the best of our knowledge, we are the first to use HDR camera in traffic light recognition, one only exception is [15] in which they also utilized two channels as in this work by fusing simple color thresholds and HOG feature based SVM classification [16]. They conducted experiments on several urban scenes but no accuracy was reported. It is clear that threshold based color segmentation

Manuscript received August 25, 2017; revised March 12, 2018 and May 31, 2018; accepted June 6, 2018. This work was supported by the A*STAR Grant for Autonomous Systems Project, Singapore. The Associate Editor for this paper was H. G. Jung. *(Jian-Gang Wang and Lu-Bing Zhou contributed equally to this work.) (Corresponding author: Jian-Gang Wang.)*

J.-G. Wang is with the Robotics Department, Institute for Infocomm Research, Singapore 138632 (e-mail: jgwang@i2r.a-star.edu.sg).

L.-B. Zhou was with the Autonomous Vehicle Department, Institute for Infocomm Research, Singapore 138632. He is now with nuTonomy, Singapore 139954 (e-mail: zlubing@gmail.com).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org. This demonstration video shows the traffic light recognition results obtained by the HDR deep learning approach presented in this paper. The algorithm has been integrated into A*STAR's autonomous vehicle (IIR AV, https://www/a-star.edu.sg/i2r/RESEARCH/AUTONOMOUS-SYSTEMS) via Data Distribution Service (DDS) and demonstrated successfully on real roads. For more demo videos, please refer to https://www.youtube.com/watch?v=HQqaAvuJI_I.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2018.2849505

will be affected by the outdoor illumination although the HSV color space is invariant to lighting conditions to some extent. Furthermore, the accuracy obtained by using the hand-crafted features, like HOG, is much lower than the one obtained by the state-of-the-art deep learned features [17]. In this paper, instead, the use of HDR camera in TLR is extensively studied. We fully take advantage of the HDR camera, i.e. the traffic light detection in the low exposure channel is fast and robust to the environment illumination and the corresponding location in the bright channel can be easily found. In addition, we adopt deep learning and tracking to improve the TLR accuracy.

Given the traffic light candidates, machine learning algorithms can be applied to identify the traffic light states. For example, Adaboost has been applied in [4]. The state-of-the-art machine learning is deep learning. The advantage of deep learning over traditional machine learning technologies is that the accuracy can be improved continuously with a large number of training samples. The progress in computer hardware makes deep learning feasible for real-time application. In this paper, we adopt deep learning to classify the traffic light from images.

The traffic light recognition together with signal lights recognition [4], [18], [19] is naturally a computer vision problem. Image processing and geometric estimation as either preprocessing or post processing could further enhance a machine learning based approach. Considering the road is flat, the geometric properties, e.g. projection between the camera and the road plane, can be used to reduce the number of the traffic light candidates to be verified by the classifier and consequentially save computational cost significantly.

The flowchart and diagram of the proposed approach are shown in Fig. 1(a) and (b), respectively. Both HDR channels are used. Firstly, traffic light candidates are detected from the low exposure channel. After pruning the traffic light candidates by the saliency map and region of interest, the detected traffic lights (counterparts in the bright image to the candidates in lower exposure channel) are recognized by a convolutional neural network. In order to speed up traffic light detection algorithm, a ROI is determined based on the camera calibration and prior knowledge about the heights of traffic lights. The light detection is executed within the ROI. In order to improve the robustness and accuracy of the traffic light recognition, a tracking technology is developed based on temporal trajectory analysis.

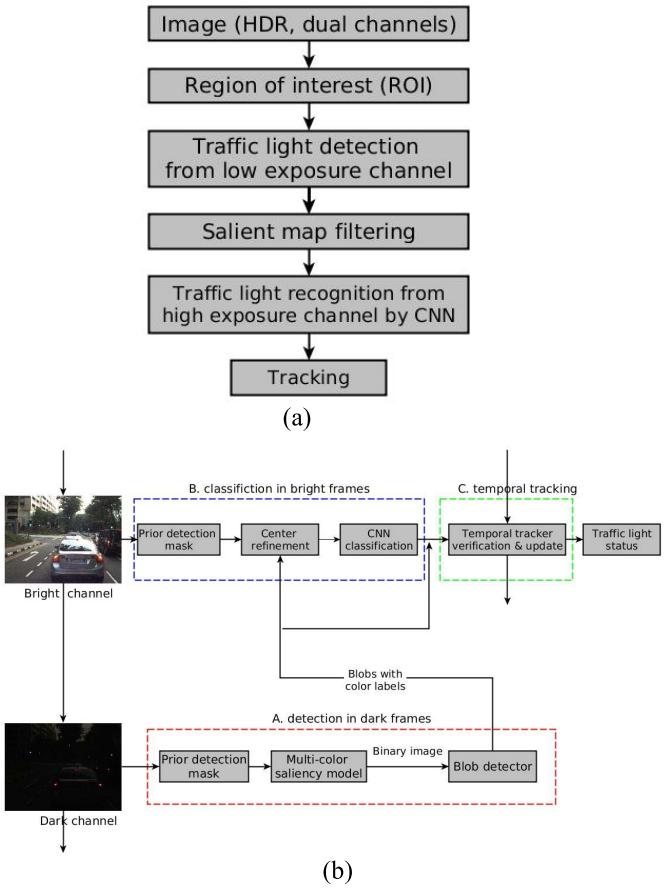


Fig. 1. (a) Flowchart of our traffic light recognition system; (b) Diagram of the proposed dual-channel traffic light recognition system.

II. HDR Imaging for Traffic Light Detection

A. Traffic Light Detection From Low Exposure Channel

Traffic light detection is an essential step of a successful traffic light classification and tracking system. The performance of bright image based traffic light detection approaches is seriously affected by the lighting conditions, rear lights of the vehicles ahead and confusion with other similar ambient light, e.g. traffic sign and pedestrian. In order to detect traffic light robustly, we propose a novel method to combine the two exposures of a HDR camera. Different from previous HDR approaches in which a synthesis image from bright and dark channels is used to detect traffic lights, in our approach the dark and bright channels are used to detect and recognize traffic lights, respectively. We can use a HDR camera in such a way as we can set the successive two channels as bright and dark by setting the dynamic range of the HDR camera. The region of the traffic lights (detected in the dark channel) can be found easily in the corresponding bright channel because the two channels are obtained within a very short time, 40ms for our camera frame rate 25 fps with high-definition serial digital interface (HD-SDI). This cannot be affected largely by high speed.

The use of the HDR camera makes the detection of the traffic light more robust than previous approaches as the lights are easily separated from the dark background in the low exposure channel. The HDR dual-channel mechanism can make full use of undistorted color and shape information in dark frames as well as rich context in bright frames.

An example of the traffic light detection and recognition based on dual-channel images is shown in Fig. 2. We can see that the traffic lights and rear lights of the vehicles ahead are prominent in the dark image and with rich context in the bright image.

One of the challenging issues in HDR camera traffic light detection is the low lighting conditions. A simple color threshold is applied [15] to detect traffic light from the dark channel. However, the performance is unreliable because it is

Fig. 2. Traffic light detection and recognition results. (a) Bright image; (b) Dark image; (c) Overlay saliency map of the ROI to the dark image; (d) Traffic lights detection and recognition results (the traffic light state is displayed in the upper right).

hard to adapt the changing illumination by only a threshold. In this paper, a salience map filtering is adopted to handle this problem.

B. Salience Map Filtering

The majority of existing methods utilize various color spaces and tuned color thresholds to detect color blobs of traffic lights. The color is primarily used for finding region of interest and classifying traffic light states. Instead of RGB color space, other color spaces (e.g. YCbCr [20] which is translated from RGB) are considered because color and intensity are mixed in three channels of the RGB color space.

Usually, specific color parameters are set for different colors (red, green and amber), and these parameters are sensitive to lighting conditions. While each pixel needs to be verified, the consumed time grows linearly with the number of colors. This paper proposes a non-parameter model to simultaneously extract blobs of various colors at an approximately constant time. To illustrate the robustness of our method, RGB color space is utilized although many papers have mentioned HSV or other spaces are better [15]. Firstly, the 3D RGB color space is partitioned to $M \times M \times M$ grids. In this paper, we use $M = 32$ without fine tuning. Secondly, the histograms for red, green and amber colors are separately calculated from the corresponding subsets of traffic light samples. Let H_r, H_g and H_a be the normalized histograms of red, green and amber colors, respectively. The histogram is first normalized to the range of [0, 1], and those values above 0.1 are truncated, then again the resulting histogram is normalized to [0, 1]. The truncation is meant to prevent extreme dominance of a single color bin.

Given an input image, red saliency score of a pixel (i, j) is computed by:

$$S_r(i, j) = \sum_{(i', j') \in N_d(i,j)} H_r(i', j') \tag{1}$$

where $N_{\mathrm{d}}(i, j)$ represents neighborhood of pixel (i, j) within a maximal distance of d; H_r represents the normalized

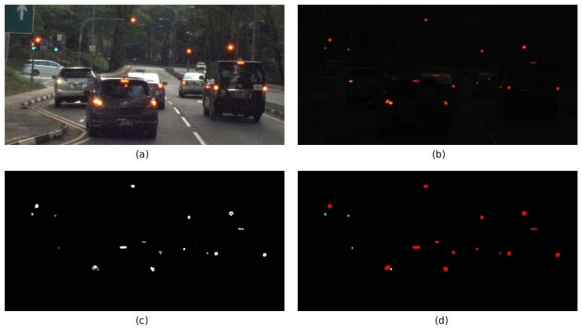


Fig. 3. Saliency map. (a) raw bright frame; (b) raw dark frame; (c) saliency map of (b); (d) the saliency map with color label.

histograms of red color. Then a saliency mask is obtained by simply applying a rough threshold, T, to the saliency image S_r. In our experiments, $d = 2$ and $T = 0.2$ are used. We would like to mention that these settings are selected without special tuning. With the learned histogram models for different light types, the saliency maps could be separated. However, it is computationally redundant to compute the saliency score of each pixel for each color. To fuse the color models of different colors, a *Max* operator is applied to merge normalized histograms of red, green and amber colors:

$$H = max(H_r, H_g, H_a) \tag{2}$$

Then overall saliency map, S, is generated by replacing H_r in Eq. (1) with H.

$$S(i, j) = \sum_{(i', j') \in N_d(i,j)} H(i', j') \tag{3}$$

Once the saliency score for a pixel is above the threshold, the three channel histogram models are further applied to compute the channel saliency scores. The color type of the pixel is set to the color with the maximize channel saliency score.

In this way, the majority of pixels could be filtered out by the overall saliency score, and the types of remaining small portion of pixels could be determined by individual saliency models. Fig. 3 shows an example of the proposed saliency model.

Contours are extracted from the resulting binary image. We use the function findContours() in the OpenCV [21]. Shape criteria are adopted to remove those obviously incorrect blobs. These criteria include the area of blobs in pixels and circularity. However this is an optional step.

C. Auto Exposure for Uncontrolled Illumination

How to make a vision system work robustly under dynamic light conditions is still a challenging problem. Adjusting camera exposure should be considered for successful traffic light detection. The HDR camera used in this paper, Point Grey Zebra2, has auto-exposure function. However, it will be disabled when we activate high dynamic range by setting dynamic ranges for each channel. As the scene dynamic range of the camera sometimes is wider than the dynamic range that we set for the camera due to the sunlight and skylight, the traffic light detection may be unstable. Although the use of the salience map makes the detection of large illumination variations more

reliable than a simple threshold, the severe illumination change in outdoor uncontrolled environment should be considered.

Auto exposure keeps image features by controlling exposure parameters including gain and shutter speed. Here we consider the auto exposure for both dark and bright channels.

Although some auto exposure methods have been developed [22]–[24], a real-time approach is needed in our application. In this paper, we propose an auto exposure method which adjusts the exposure based on the difference of the average intensity of an image mask from a reference value.

Assuming a target value of the average image intensity is represented as I_{t}, the average intensity of the current frame is I_{c}, we define a factor

$$f = \frac{I_t}{I_c} \tag{4}$$

We aim at to update gain or shutter value to let f in a small interval of 1. In other words, the gain and shutter are updated to keep the average intensity of the image close to a target value.

To achieve desired factor f, shutter and gain are jointly adjusted within their respective ranges $[s_{min}, s_{max}]$ and $[g_{min}, g_{max}]$. In the implementation, shutter is adjusted first and then gain, as large gain normally also introduces noise. Specifically, shutter time is firstly adjusted to reach the factor. In a result, the factor f_s by shutter is calculated as:

$$f_{\mathrm{s}} = \frac{s_{\mathrm{t}}}{s_{\mathrm{c}}} \tag{5}$$

where s_{c} is current shutter, and s_{t} is the updated shutter.

Basically, the shutter value is in direct proportion to intensity. If shutter only (within shutter range) is able to achieve the desired factor f, i.e. $f_s = f$, there will be no more gain adjustment. If f_s is not able to reach target image intensity, then shutter update will reach its extreme within the range, and the remaining portion of the factor will be covered by gain adjustment, i.e. $f = f_s f_g$, where f_g is the factor by gain. It is a common observation, that when the gain increases or decreases approximately 6 db, the intensity doubles or halves. Based on this observation, the remaining factor can be easily achieved by adjusting the gain.

III. Region of Interest

In order to speed up traffic light detection algorithm, prior knowledge about the location of traffic light on an image can be used.

The transformation matrix between the 3D world coordinate system and image plane is as follows.

$$\begin{bmatrix} ut \\ vt \\ t \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{6}$$

where (x, y, z) are the coordinates of the world and (u, v) are the coordinates on the image plane. The world coordinate system is defined with origin set as the middle point of the autonomous vehicle's head. The X-axis points forward toward the front of the vehicle but lies in the ground plane. The Y-axis points toward the left-hand side of the vehicle but lies in the ground plane, the Z-axis points up and perpendicular to the ground. The coordinate centre is defined as the projection of the camera centre on the ground. There are 11 unknown parameters in equation (6), so we need at least four groups of 3D and 2D coordinates to calibrate the camera, i.e., to calculate the parameters. However for robustness, we calibrate the camera using all the 3D and 2D coordinates as possible and the parameters can be obtained by a least-square fitting procedure. Note these 3D coordinates are obtained by placed a few fixed-height cuboids on a flat ground where their coordinates on the ground plane are measured. The 2D coordinates in image plane can be easily obtained by picking the pixel.

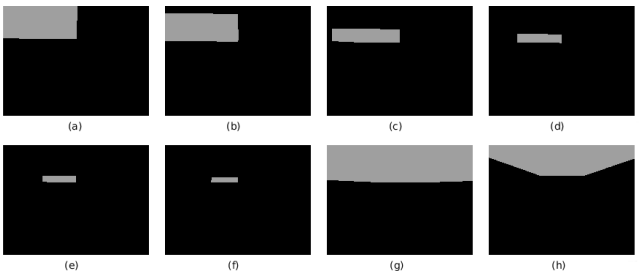


Fig. 4. Prior detection masks corresponding to different detection ranges (unit: meter) in x, y and z directions. In our real experiments, the mask in (g) is applied. (a) x[0, 10], y[0, 8] z[2.5, 4]. (b) x[10, 20], y[0, 8] z[2.5, 4]. (c) x[20, 30], y[0, 8] z[2.5, 4]. (d) x[30, 40], y[0, 8] z[2.5, 4]. (e) x[40, 50], y[0, 8] z[2.5, 4]. (f) x[50, 60], y[0, 8] z[2.5, 4]. (g) x[0, 60], y[−8, 8] z[2.5, 4]. (h) x[0, 60], y[−8, 8] z[4.5, 7].

Given accurate vehicle localization and pose, as well as the pre-stored 3D localization of traffic lights, pixel location in the image plane could be computed, and then traffic light status can be easily verified using colors, shapes, or weaker classifiers. However, this kind of method relies on the accuracy of map and localization, which is not very practical for general players. In this paper, a coarse range of the relative position of the traffic lights is utilized to estimate the potential areas (region of interest) in the image plane. Specifically, the detection ranges for vertically hanged traffic lights in three directions are: [0m, 60m] in longitudinal detection (x), [−8m, 8m] in lateral direction (y), and [2.5m, 4m] in an upward direction (z). To get the prior detection region, various (x, y, z) values by incrementing x, y, and z with small steps are substituted into Eq. (6). The resulting prior detection region is shown in Fig. 4(g). Fig. 4(h) shows the prior detection mask for horizontally hanged traffic lights, which differ in height direction with a range of [4.5m, 7m]. Suppose as either the vehicle pose or the traffic light location is accessible, such prior detection masks could be further shrunk to narrow the search region. Fig.4(a) - Fig.4(f) show more examples for prior masks at various ranges.

In our experiments, we use the image only and do not make any assumptions about the map, vehicle's location and pose, so we use Fig. 4(g) as the region of interest. Specifically, when no accurate localization and vehicle pose is provided, a very rough 2D GPS level position can be sufficient to guess a very loose range in x, y, z direction. Based on these 3D ranges, we can make an intensive 3D grid and correspondingly generate the 2D image ROI. In other words, we adopt

an ROI corresponding to the longest distance where the traffic lights are expected to be found.

IV. Deep Learning for Traffic Light Recognition

Deep learning has been a state-of-the-art machine learning technology. It aims to learn hierarchical representations of data by using deep architecture. Deep learning has recently achieved very promising results in a wide range of areas such as computer vision, speech recognition and natural language processing [25].

In this paper, we adopt deep learning to recognize traffic light status from images. The idea is that a convolutional neural network can be learned from a large traffic light database and the detected traffic light candidates can be classified by this CNN. By designing model (including parameter setting) carefully, we have shown that a high recognition accuracy can be achieved in real-time.

A. Correspondence Between the Dark and Bright Channels

The images captured via dark and bright channels are not synchronized although the difference between their timestamps is very small. In addition, the vehicle motion will make it challenging for aligning the detected light from dark image to bright image, especially when the vehicle suffers from vibration. Hence, it requires to re-locate the traffic light candidate in the bright channel based on the detection results obtained from the dark channel.

Based on the detected blobs in the dark frame, corresponding sub-images with richer texture are obtained from the next bright frame. However, due to the motion between consecutive frames, center refinement is needed to ensure the sub-images are cropped with canonical center. In our implementation, the center position p and radius r of light blobs could be evaluated in the dark frame. In the following bright frame, the new center is searched within a $12r \times 12r$ image window centered at p. Intuitively, the light blob center normally has highest brightness and color variance in this image window. Specifically, based on RGB space, brightness image

$$I = 0.2126 * R + 0.7152 * G + 0.07221 * B \tag{7}$$

and the variance image

$$V = |R - I| + |G - I| + |B - I| \tag{8}$$

are computed. Similar to the approach proposed in [26] and [27], the new center is then found at the highest response in a weighted image:

$$\alpha V + (1 - \alpha) I \tag{9}$$

where α controls the weights of the variance image and brightness image, since brightness often varies greatly with the lighting conditions, hence we roughly set it to 0.7 (biased to variance image) by experience.

Image patches in bright frames are cropped as $12r \times 12r$ window centered at the new center, corresponding to the candidate's center in the dark channel.

Fig. 5. Labeling training samples on an image, blue : Vertically Aligned Red Light (VARL); green: Horizontally Aligned Red Light (HARL); red: Other Fake Red Light (ORFL).

B. Customized Convolutional Neural Network

There are still fake false positives from braking lights and other shinning objects although most of them have been filtered out from the traffic light candidates. To achieve better robustness, a CNN classifier is trained to distinguish the true positive from false positives.

The CNN model is customized based on the classic CaffeNet [28]. In our model, the number of output in the last layer is 13: 12 positive classes plus background class. The 12 positive classes include:

1) Horizontally Aligned Red Light (HARL)
2) Vertically Aligned Red Light (VARL)
3) Horizontally Aligned Green Light (HAGL)
4) Vertically Aligned Green Light (VAGL)
5) Left Vehicle Light (LVL)
6) Right Vehicle Light (RVL)
7) Green Arrow Light (GAL)
8) Red Arrow Light (RAL)
9) Amber Light (AL)
10) Green Pedestrian Light (GPL)
11) Red Pedestrian Light (RPL)
12) Other Fake Red Light (OFRL)

An example of the labeled image is shown in Fig. 5 where true HARL and VARL are labeled in green and blue, and OFRL are labeled in red.

The reason behind the categorization is to reduce within-class variance (split horizontal and vertical lights) and learn to distinguish red traffic lights from common false positives (class LVL, RVL, and OFRL), while reducing effort on data collection and annotation.

We adopted CaffeNet [28] in this paper because the speed requirement is high in our application. As real-time performance is critical, the size of the input image is 111×111 in our model rather than 224×224 in the original CaffeNet model. In addition, whilst we kept the main body of original architecture, the first convolutional layer is modified: kernel size changed from 7 to 3, and stride changed from 4 to 2.

Fine tuning strategy is utilized to train the weights of the CNN classifier using Caffe [9]. The basic learning rate and decay weight are set to 0.001 and 0.0005. The multipliers of learning rate for modified layers, i.e. first convolutional layer and output layer, are set to 10 in first 2,000 iterations, and then set back to 1 as the other layers. In total, the training procedure takes 50,000 iterations.

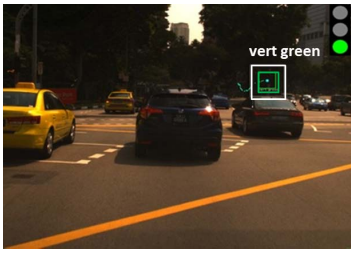


Fig. 6. Plot the trajectory of the traffic light (marked in green) onto current frame.

V. Temporal Trajectory Analysis

In order to improve the accuracy and robustness of traffic light recognition, tracking technology is needed. In this paper, we develop a traffic light tracking approach based on temporal trajectory analysis.

A high-speed in-vehicle camera ensures that the relative position and size of a target in the captured image constantly change from frame to frame. Temporal spatial analysis is a process to examine previous frames and determine whether a candidate detected in the current frame has been found in the same area earlier. Temporally, traffic light status stays consistent for certain period of time in practice. Meanwhile, locations of the light in the image plane in sequential frames are spatially continuous. Proper temporal spatial tracking can greatly benefit the task in two aspects: (1) improve the result's smoothness by filling light status of the middle frames with missing or low confident detections; (2) improve the detection confidence and reduce isolated false positives. Overall, this module is trying to track two entities: spatial location of detected instances and history of traffic light status.

Specifically, we term the whole tracking history of a light instance as a trajectory. A trajectory has a few components: type, a vector of points storing history locations, lifetime, and discontinuity. The trajectory is categorized in terms of three colors of light and one Boolean flag indicating the trajectory stability. This will result in six types of trajectories: stable red, stable green, stable amber, temporary red, temporary green, and temporary amber. For instance, a stable green trajectory means the trajectory has been confirmed as a tracking of a green traffic light (horizontal or vertical lights are not separated). The lifetime depicts the existing period of the trajectory since the first detection of a traffic light instance at the beginning of the trajectory, and the discontinuity of trajectory records the number of passed frames since the last detection of the instance. An example of the trajectory is shown in Fig. 6.

The trajectory pool updates after every frame. At the very beginning, every trajectory is initialized as a temporary trajectory. It requires a minimal lifetime of 1 second and minimal 5 detections of the instance for changing a temporary trajectory to stable trajectory (these parameters are empirically set). A trajectory is removed from the trajectory pool when its lifetime is above a threshold (e.g. 70 second in our experiments). Normally lifetime of red, green or amber lights is below the threshold. Even if sometimes the red light may last above the threshold, it is still feasible to split the whole cycle to two trajectories. Given an input bright image, traffic lights are first detected using the aforementioned dual channel fusion method. These newly detected points are then added into the trajectory pool. As an example, suppose a red light point is detected in the current frame, these trajectories with red color in the pool are traversed and verified by calculating the distance between the point and the latest point stored in a trajectory. If the minimal distance among all red trajectories is below a certain value (60-pixel distance is used in this paper), the new point will be added into that trajectory. If no valid trajectory is found, then a new temporary red trajectory will be created with the new point. When a stable trajectory is found, we would be confident to say the new point is a stable red light. If only a temporary trajectory is found, then the new point is considered a temporary red light, which occasionally is a false alarm.

VI. Experimental Results

Based on the salience map and region of interest discussed in section II and III respectively, the computational consumption is significantly reduced. This is very important for autonomous vehicles on which many modules run simultaneously. In addition, the accuracy has been improved because some possible false positives are prevented due to the reduction of the candidates by the region of interest.

A. Evaluation of Performance

In order to quantitatively evaluate the performance of the proposed method, the traffic light detector has been conducted on a large database collected with our autonomous vehicle.

We computed the number of True Positive (TP), False Positive (FP) and False Negative (FN) for all test images. We used precision and recall, defined as equations (10) and (11), to evaluate the performance.

$$Precision = \frac{TP}{TP + FP} \tag{10}$$

$$Recall = \frac{TP}{TP + FN} \tag{11}$$

The database for training and testing the classifier includes 4,142 images selected from 50 video clips. The images are selected to have nearly equal number of samples for each class. We labeled the images manually and 21,070 labeled boxes were annotated. The number of boxes for each class is about 1,750. In our experiments, 3,722 (about 90%) images were used as a training set and the remaining 420 images were used for the purpose of evaluation during the training process. A total of one million training samples are generated from these seed samples. This can be done by randomly scaling and translation of the raw seed samples. We found that the balance among the different classes is important to achieve good classification performance; hence the number of the generated training samples for each class is nearly uniform.

In our approach, the resolution of the original image is 1600×1200, new samples are generated by shifting the center of the detected traffic light candidates following a uniform

TABLE I

OUR HDR APPROACH: CONFUSION MATRIX OF THE TEST RESULTS WITHOUT/WITH REGION-OF-INTEREST, THE TEST RESULTS WITH THE ROI ARE RECORDED IN BRACKETS

	HARL	VARL	HAGL	VAGL	LVL	RVL	GAL	RAL	AL	GPL	RPL	OFRL
HARL	**441(441)**											
VARL		**783(783)**										
HAGL			**568(468)**									
VAGL			9(9)	**549(549)**						9 (9)		
LVL		18 (0)			**1152(66)**	72(0)						
RVL					18(0)	**864(33)**						
GAL				6(6)			**180 (180)**					
RAL								**90 (90)**				
AL									**72 (72)**			
GPL				6 (6)						**117 (117)**		
RPL				3 (3)							**162 (162)**	
OFRL		3(0)										**33 (39)**

TABLE II

OUR HDR APPROACH: THE PRECISION AND RECALL WITHOUT/WITH ROI, THE RESULTS WITH THE ROI ARE RECORDED IN BRACKETS

	HARL	VARL	HAGL	VAGL	LVL	RVL	GAL	RAL	AL	GPL	RPL	OFRL	Average
Recall (%) without ROI/(with ROI)	100 (100)	97.4 (100)	98.1 (98.1)	97.3 (97.3)	98.5 (100)	92.3 (100)	100 (100)	100 (100)	100 (100)	92.9 (92.9)	100 (100)	100 (100)	98 (99)
Precision (%) without ROI/(with ROI)	100 (100)	100 (100)	100 (100)	96.8 (96.8)	92.8 (100)	98 (100)	96.8 (96.8)	100 (100)	100 (100)	95.1 (95.1)	98.2 (98.2)	91.7 (100)	97.5 (98.9)

TABLE III

OUR HDR APPROACH: DETECTION RATE WITHOUT/WITH REGION-OF-INTEREST, THE TEST RESULTS WITH THE ROI ARE RECORDED IN BRACKETS

	HARL	VARL	HAGL	VAGL	LVL	RVL	GAL	RAL	AL	GPL	RPL	OFRL	average
Ground truth without ROI/(with ROI)	447 (447)	804 (804)	477 (477)	567 (567)	1311 (66)	921 (36)	192 (192)	93 (93)	72 (72)	132 (132)	174 (174)	39 (39)	
Detection rate (%) without ROI/(with ROI)	98.7 (98.7)	97.4 (97.4)	98.1 (98.1)	100 (100)	94.7 (100)	95.8 (100)	96.9 (96.9)	96.8 (96.8)	100 (100)	93.2 (93.2)	94.8 (94.8)	92.3 (100)	96 (97.9)

random distribution from -0.2 to 0.2 times of the rectangle's width or height, and then resizing the detected traffic light candidates following a uniform random distribution from 1 to 1.2 times. The generated samples are finally resized to 111×111.

In order to evaluate the performance of the system, we ran our code through another set of video sequences (63 videos, each video about 4 minutes long) which include the different time of a day, weather conditions, expressway, and urban road. 1,800 images are selected (sampling interval of every 80 frames) from these video sequences. The selected images are manually labeled and 5,229 (3,099 if ROI is considered) labeled boxes were annotated as ground truth. Some experimental results are shown in Fig.7(a).

The test results on the full data set are recorded in Table I where the test results with the region-of-interest are recorded in brackets.

We can see that the accuracies of vehicle light are worse than the one of other classes. One reason is that the goal of the application is to detect and recognize the traffic light status, and lots of vehicle lights in the dataset are not completely annotated. Another reason for this could be that the types of the vehicle light are much more than the one of other types. In order to improve the accuracy of the vehicle light, we have to collect more training samples to cover more kinds of vehicle lights.

In this paper, we are interested in traffic light recognition although we consider vehicle light as a potential application in the future. By applying ROI discussed in Section III, we can see from Table I that most of the vehicle lights will be removed from the results because they are at the lower part of the images.

Based on Table I, the precision and recall can be computed with equation (10) and (11), respectively. The results are shown in Table II where the test results with the ROI are recorded in brackets.

From Table II, we can see that by applying the ROI, the average recall of the system has been improved from 98% to 99% and the average precision has been improved from 97.5% to 98.9%.

In our paper, the detection is based on saliency map generated from the dark frame. The detection rate is computed as follows.

$$D = (TP + FN)/G \tag{12}$$

where G is the number of the ground truth.

TABLE IV

YOLOv2 APPROACH: CONFUSION MATRIX OF THE TEST RESULTS WITHOUT/WITH REGION-OF-INTEREST, THE TEST RESULTS WITH THE ROI ARE RECORDED IN BRACKETS

	HARL	VARL	HAGL	VAGL	LVL	RVL	GAL	RAL	AL	GPL	RPL	OFRL
HARL	**423 (423)**	9(9)										
VARL	9 (9)	**771 (771)**										
HAGL			**456 (456)**	6 (6)								
VAGL			6 (6)	**543(543)**	3 (3)					12(12)		
LVL		3 (3)			**1161(63)**	63 (0)						
RVL					51 (0)	**864 (36)**						
GAL				6 (6)			**177 (177)**					
RAL		6 (6)						**78 (78)**				
AL									**69 (69)**			
GPL			3 (3)	6 (6)						**121 (121)**		
RPL				3 (3)							**162 (162)**	
OFRL		3 (3)	3 (3)									**30 (33)**

TABLE V

YOLOv2 APPROACH: THE PRECISION AND RECALL WITHOUT/WITH REGION-OF-INTEREST, THE RESULTS WITH THE ROI ARE RECORDED IN BRACKETS

	HARL	VARL	HAGL	VAGL	LVL	RVL	GAL	RAL	AL	GPL	RPL	OFRL	Average
Recall (%) without ROI/(with ROI)	97.9 (97.9)	97.3 (97.3)	97.4 (97.4)	96.3 (96.3)	95.6 (95.5)	93.2 (100)	100 (100)	100 (100)	100 (100)	90.2 (90.2)	100 (100)	100 (100)	97.3 (97.9)
Precision (%) without ROI/(with ROI)	97.9 (97.9)	98.8 (98.8)	98.7 (98.7)	96.3 (96.3)	94.6 (95.5)	94.4 (100)	96.7 (96.7)	92.9 (92.9)	100 (100)	92.5 (92.5)	98.2 (98.2)	83.3 (84.6)	95.4 (95.5)

TABLE VI

YOLOv2 APPROACH: DETECTION RATES WITHOUT/WITH REGION-OF-INTEREST, THE RESULTS WITH THE ROI ARE RECORDED IN BRACKETS

	HARL	VARL	HAGL	VAGL	LVL	RVL	GAL	RAL	AL	GPL	RPL	OFRL	average
Ground truth without ROI/(with ROI)	447 (447)	804 (804)	477 (477)	567 (567)	1311 (66)	921 (36)	192 (192)	93 (93)	72 (72)	132 (132)	174 (174)	39 (39)	
Detection rate (%) without ROI/(with ROI)	96.6 (96.6)	97.0 (97.0)	96.9 (96.9)	99.5 (99.5)	93.6 (95.6)	99.3 (100)	95.3 (95.3)	90.3 (90.3)	95.8 (95.8)	90.9 (90.9)	94.8 (94.8)	92.3 (100)	95.2 (96.1)

In our experiments, the number of the ground truth for each class is shown as the first row in Table III. Based on Table I, the detection rates are given as the second row in Table III where the test results with the region-of-interest are recorded in brackets.

From Table III, we can see that the average detection rate has been improved from 96% to 97.9% by applying the ROI.

B. Comparison With the State-of-the-Art

To the best of our knowledge, there is no publically available HDR traffic light benchmark database. Most published traffic light recognition systems are evaluated on their databases which are collected using a normal color camera. Multiple exposure images were used in [15], where the authors conducted experiments on several urban scenes, but there was no accuracy reported. This makes the comparison between previous methods with the proposed approach in this paper difficult.

Nevertheless, in order to show the performance improvement by the proposed HDR approach, we compare our approach with the state-of-the-art deep learning object detection approach. For this purpose, the results obtained by running state-of-the-art approach on only high exposure images of our test data will be used. In order to fairly compare the performance, the state-of-the-art approach is re-trained with the same database used for training CaffeNet in the last section.

For our applications, two criteria for selecting an object detector are: (1) real-time; (2) high accuracy. Although there are several other deep learning object detection approaches, like Faster-RCNN [29], only few of them can be run in real-time. In this paper, one of the state-of-the-art object detection approaches, You Look at Only Once (YOLO) [30], is adopted. We select YOLO because it can run in real-time and the accuracy of the version 2 (YOLOv2) [30] outperforms other state-of-the-art methods like Faster R-CNN [29] and SSD [31] on publicly available benchmark databases. In doing so, YOLOv2 is run on single channel images, bright channel images, of our database.

Some experimental results obtained by YOLOv2 are shown in Fig. 7(b).

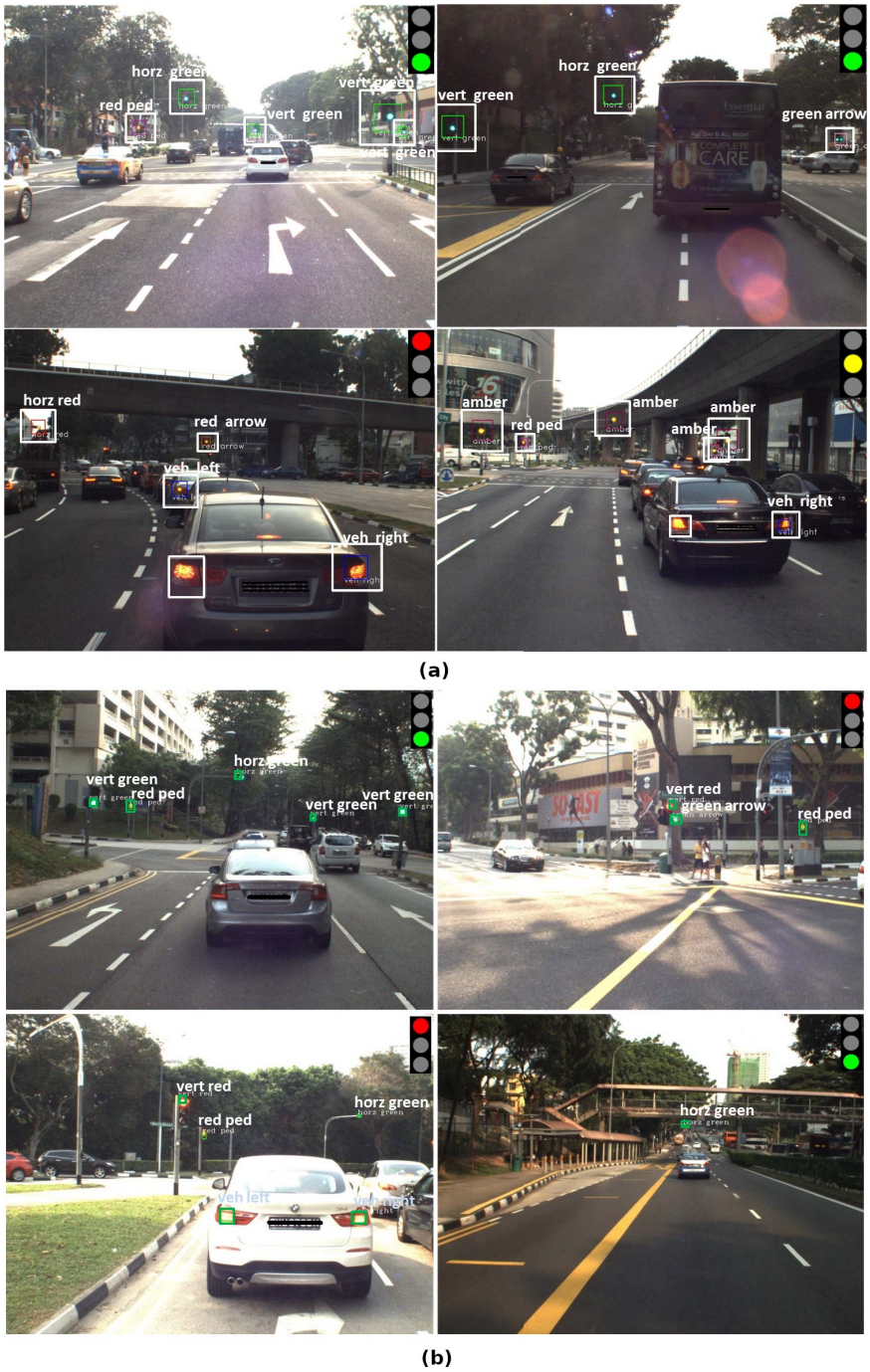


Fig. 7. Some experimental results of traffic light detection by our HDR and YOLOv2 methods, respectively. (a) Our HDR approach results including: green ("vert green" or "horz green", marked in green), red ("vert red" or "horz red", marked in red), amber ("amber", marked in amber), green arrow ("green arrow", marked in pink), red pedestrian ("red ped", marked in purple), vehicle lights ("veh left" or "veh right", marked in blue) (b) YOLOv2 approach including: green ("vert green" or "horz green"), red ("vert red" or "horz red"), green arrow("green arrow"), red pedestrian ("red ped"), vehicle lights ("veh left" or "veh right").

The same tests with the HDR traffic light recognition presented in the last section have been done with the YOLOVv2 and the results are shown in Table IV to VI, respectively.

By considering detection rate, the true precision and recall can be computed. This can be done by multiplying precision or recall rate with detection rate, respectively. The comparison of the true precision and recall rate for YOLOv2 and our approach are given in Table VII where the rates with region-of-interest are recorded in brackets. We can see that our approach achieves better performance than YOLOv2 either with or without ROI. This shows that the proposed dual-channel approach is better than the existing approach which uses only one channel. With ROI, the precision rate is improved from 92.5% to 96.8% and the recall rate is improved from 94.3% to 96.9%.

Fig. 8. Comparison of the results obtained by YOLOv2 and our HDR approach. Left: YOLOv2 TLR results (two false positives are detected which are caused by the reflection of the traffic light on the bus body and the sunlight on the building, respectively, marked in red circle); Middle: Our HDR TLR results; Right: corresponding dark image. The two false positives in the YOLOv2 TLR results (left) can be prevented in the HDR TLR results (middle) because there is no response for these two false positives in the dark channel and no traffic light candidates will be detected.

Fig. 9. Comparison of the results obtained by YOLOv2 and our HDR approach. Left: YOLOv2 TLR results (one false alarm is detected, marked in red circle); Middle: our HDR TLR results; Right: corresponding dark image. The false alarm in the YOLOv2 TLR results (left) can be prevented in the HDR TLR results (middle) because there is no response for the false alarm in the dark channel and no traffic light candidates will be detected.

Fig. 10. Comparison of the results obtained by YOLOv2 (top, false positives caused by traffic sign and pedestrian etc.) and our HDR approach (bottom). The false positives in the YOLOv2 TLR results (top) can be prevented in the HDR TLR results (bottom) because there are no responses for the false positives in the dark channel and no traffic light candidates will be detected.

In our experiments, the use of dark channel has been verified to prevent many false positives caused by using only bright images. The false positives caused by the traffic signs, sunlight, clothes of the pedestrian etc. can be prevented because the corresponding region in the dark channel images is not very visible. In Fig. 8, an example is given to compare the traffic light recognition results obtained by YOLOv2 and our methods, respectively. We can see that there are two false positives in the YOLOv2 results, one caused by the reflection of the traffic light on the bus body and another is caused by the sunlight on the building. On the contrary, HDR can prevent such false positives because there is no response for these two false positives in the dark channel and no traffic light candidates will be detected. Similarly, HDR can prevent false positives by YOLOv2 with the help of dark image in Fig. 9. In Fig 10, more examples show that false positives reported in YOLOv2 results can be prevented by our HDR approach, the dark images are not shown because of the space limitation.

TABLE VII

COMPARISON OF THE PRECISION AND RECALL WITHOUT/WITH REGION-OF-INTEREST, THE RESULTS WITH THE ROI ARE RECORDED IN BRACKETS

	YOLOv2	HDR
Recall (%)	92.6 (94.3)	94.1 (**96.9**)
Precision (%)	90.8 (92.5)	93.6 (**96.8**)

TABLE VIII
THE COMPARISON OF AVERAGE PROCESSING TIME FOR EACH FRAME WITH/WITHOUT REGION-OF-INTEREST

	With ROI	Without ROI	Time saving
HDR	35 ms	130 ms	**77%**
YOLO	40 ms	40 ms	0

The difference in performance on the same dataset has shown that the proposed approaches are better than the state-of-the-art technique in both accuracy and speed.

The algorithms presented in this paper have been implemented in C++. The traffic light detection and recognition on a GIGABYTE Mini-PC (2.5Ghz CPU + GTX 760) can run in about 30-40 fps depending on the number of the traffic lights in an image. Computation saving with and without the region of interest is given in Table VIII. We can see that in our approach the time saved by using the region of interest is significant. YOLOv2 does not save time by using the ROI because a fixed size image is required.

The algorithm has been integrated into our autonomous vehicle (IIR AV) [32] via Data Distribution Service (DDS) [33] and demonstrated successfully on real roads. The traffic light recognition results of a sequence, from left to right, top to down, at ten frame intervals, are shown in Fig. 11. We have submitted a demonstration video together with this paper. For the interested readers, please refer to link: https://www.youtube.com/watch?v=HQqaAvuJI_I

Fig. 11. The traffic light recognition results of a sequence of frames, from left to right, top to down, at ten frame intervals. The algorithm can recognize horizontally aligned green light (HAGL), vertically aligned green light (VAGL), left vehicle light (LVL), right vehicle light (RVL).

VII. CONCLUSION AND FUTURE WORK

In order to overcome the challenging problems in traffic light recognition, we have proposed a real-time traffic light recognition system based on HDR imaging and deep learning in this paper. Different from the state-of-the-art approach which use bright image only, the low and high exposures are used in our approach for traffic light detection and recognition, respectively. The clear background of the low exposure (dark) channel makes the traffic light detection much more reliable than the existing color image based approaches. On the other hand, the candidate regions in the high exposure corresponding to the dark channel can be recognized well because of the rich texture. By using salience map and region of interest, which can prune the traffic light candidates efficiently, the number of the traffic light candidates to be verified by CNN can be reduced significantly. This makes the system fast and robust to noise (such as vehicle's rear lights). Furthermore, a tracking technology is developed to further improve the reliability and accuracy. The experimental results, which are based on a large database collected from real roads, have shown that the performance of the proposed traffic light recognition is better than the state-of-the-art deep learning object detector which uses only bright images. The candidate selection from the dark images allows our approach to prevent false positives caused by the traffic signs, pedestrian, temporary road signs which have similar color with traffic light. The method presented in this paper has been integrated into our autonomous vehicle, the tests on the real road have shown that it can satisfy the requirements of an autonomous vehicle in terms of speed and accuracy.

Future investigation will look into using both dark and bright images as input to the CNN network. They carry both salient color and shape information as well as rich context and could further improve the accuracy of traffic light recognition. The performance of the proposed method at night could be done in the future. As traffic light detection from the dark channel should not be affected greatly at night, we believe that the proposed approach should be feasible at night. This can be done by properly adjusting camera parameters and re-training the CNN with the data collected at night. The generalization of the proposed approach for vehicle signaling lights, e.g. brake lights, left turn and right turn, could be investigated. The vehicle detection results, obtained based on vision, lidar or radar, will serve as region-of-interest of such signal light detector [18]. Furthermore, the fusion of the proposed approach with other traffic light localization technologies such as Route Network Definition File(RNDF) could be done in the future. By fusing with RNDF information, most of the false positives can be eliminated from the traffic light candidates.

REFERENCES

[1] M. B. Jensen, M. P. Philipsen, M. Trivedi, T. Møgelmose, and T. Moeslund, "Vision for looking at traffic lights: Issues, survey, and perspectives," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 7, pp. 1800–1815, Jul. 2016.

[2] M. Diaz, G. Pirlo, M. A. Ferrer, and D. Impedvov, "A survey on traffic light detection," in *Proc. Workshops New Trends Image Anal. Process.*, vol. 9281, 2015, pp. 201–208.

[3] M. P. Philipsen, M. B. Jensen, T. Møgelmose, T. B. Moeslund, and M. M. Trivedi, "Ongoing work on traffic lights: Detection and evaluation," in *Proc. 12th IEEE Int. Conf. Adv. Video Signal-Based Surveill. (AVSS)*, Aug. 2015, pp. 1–5.

[4] J. Gong, Y. Jiang, G. Xiong, C. Guan, G. Tao, and H. Chen, "The recognition and tracking of traffic lights based on color segmentation and CAMSHIFT for intelligent vehicles," in *Proc. IEEE Intell. Vehicle Symp.*, Jun. 2010, pp. 431–435.

[5] G. Siogkas, E. Skodras, and E. Dermatas, "Traffic lights detection in adverse conditions using color, symmetry and spatiotemporal information," in *Proc. Int. Conf. Comput. Vis. Theory Appl.*, 2012, pp. 620–627.

[6] R. Charette and F. Nashashibi, "Traffic light recognition using image processing compared to learning processes," in *Proc. IEEE/RSJ Int. Conf. Robots Syst.*, Oct. 2009, pp. 333–338.

[7] M. Diaz-Cabrera, P. Cerri, and J. Sanchez-Medina, "Suspended traffic lights detection and distance estimation using color features," in *Proc. Int. IEEE Conf. Intell. Transp. Syst.*, Sep. 2012, pp. 1315–1320.

[8] J. Levinson, J. Askeland, J. Dolson, and S. Thrun, "Traffic light mapping, localization, and state detection for autonomous vehicles," in *Proc. Int. IEEE Conf. Robot. Automat. (ICRA)*, May 2011, pp. 5784–5791.

[9] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. ACM Int. Conf. Multimedia*, 2014, pp. 675–678.

[10] N. Fairfield and C. Urmson, "Traffic light mapping and detection," in *Proc. Int. IEEE Conf. Robot. Automat. (ICRA)*, May 2011, pp. 5421–5426.

[11] V. John, K. Yoneda, B. Qi, Z. Liu, and S. Mita, "Traffic light recognition in varying illumination using deep learning and saliency map," in *Proc. Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2014, pp. 2286–2291.

[12] V. Gradinescu, C. Gorgorin, R. Diaconescu, V. Cristea, and L. Lftode, "Adaptive traffic lights using car-to-car communication," in *Proc. 65th IEEE Veh. Technol. Conf.*, Apr. 2007, pp. 21–25.

[13] N. Kumar, N. Lourenco, D. Terra, L. N. Alves, and R. L. Aguiar, "Visible light communications in intelligent transportation systems," in *Proc. IEEE Intell. Vehicle Symp.*, Jun. 2012, pp. 748–753.

[14] K. Dresner and P. Stone, "A multiagent approach to autonomous intersection management," *Artif. Intell. Res.*, vol. 31, pp. 591–656, Mar. 2008.

[15] C. Jang, C. Kim, D. Kim, M. Lee, and M. Sunwoo, "Multiple exposure images based traffic light recognition," in *Proc. IEEE Intell. Vehicle Symp.*, Jun. 2014, pp. 1313–1318.

[16] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. Int. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, pp. 886–893.

[17] J. Wang *et al.*, "Learning fine-grained image similarity with deep ranking," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1386–1393.

[18] J.-G. Wang, L.-B. Zhou, Y. Pan, S. Lee, B.-S. Han, and V. B. Saputra, "Appearance-based brake-lights recognition using deep learning and vehicle detection," in *Proc. IEEE Intell. Vehicle Symp.*, Jun. 2016, pp. 815–820.

[19] M. Casares, A. Almagambetovand, and S. Velipasalar, "A robust algorithm for the detection of vehicle turn signals and brake lights," in *Proc. Int. IEEE Conf. Adv. Video Signal-Based Surveill.*, Sep. 2012, pp. 386–391.

[20] H.-K. Kim, J. H. Park, and H.-Y. June, "Effective traffic lights recognition method for real time driving assistance system in the daytime," in *Proc. 59th World Acad. Sci., Eng. Technol.*, 2011, pp. 1–4. [Online]. Available: https://waset.org/publications/725/effective-traffic-lights-recognition-method-for-real-time-driving-assistance-systemin-the-daytime

[21] *Finding Contours in Your Image, OpenCV Tutorials*. Accessed: Jul. 6, 2018. [Online]. Available: https://docs.opencv.org/2.4/doc/tutorials/imgproc/shapedescriptors/find_contours/find_contours.html

[22] H. Lu, H. Zhang, S. Yang, and Z. Zheng, "Camera parameters auto-adjusting technique for robust robot vision," in *Proc. Int. IEEE Conf. Robot. Automat.*, May 2010, pp. 1518–1523.

[23] V. Agarwal, B. R. Abidi, A. Koschan, and M. A. Abidi, "An overview of color constancy algorithms," *J. Pattern Recognit. Res.*, vol. 1, no. 1, pp. 42–54, 2006.

[24] I. Shim, J.-Y. Lee, and I. S. Kweon, "Auto-adjusting camera exposure for outdoor robotics using gradient information," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, Sep. 2014, pp. 1011–1017.

[25] *Deep Learning*. Accessed: Jul. 6, 2018. [Online]. Available: https://en.wikipedia.org/wiki/Deep_learning

[26] Y. Hu, X. Xie, W.-Y. Ma, L.-T. Chia, and D. Rajan, "Salient region detection using weighted feature maps based on the human visual attention model," in *Proc. Pacific Rim Conf. Multimedia*, 2004, pp. 993–1000.

[27] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1597–1604.

[28] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[29] S. Ren, K. He, R. Girshick, and J. Sun. (2015). *Faster R-CNN: Towards Real-Time Object Detection With Region Proposal Networks*. [Online]. Available: https://arxiv.org/abs/1506.01497

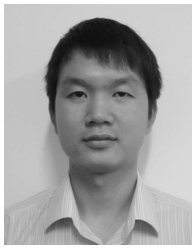
[30] J. Redmon and A. Farhadi. (2016). *YOLO9000: Better, Faster, Stronger*. [Online]. Available: https://arxiv.org/abs/1612.08242

[31] W. Liu *et al.* (2015). "SSD: Single shot multibox detector." [Online]. Available: https://arxiv.org/abs/1512.02325

[32] *Institute for Inforcomm Research Autonomous Vehicle (IIRAV)*. Accessed: Jul. 6, 2018. [Online]. Available: https://www.a-star.edu.sg/i2r/RESEARCH/AUTONOMOUS-SYSTEMS

[33] *Data Distribution Service*. Accessed: Jul. 17, 2018. [Online]. Available: https://en.wikipedia.org/wiki/Data_Distribution_Service

Jian-Gang Wang received the bachelor's degree from Inner Mongolia University in 1985, the M.Eng. degree from Shenyang Institute of Automation, Chinese Academy of Sciences, in 1988, and the Ph.D. degree from Nanyang Technological University in 2001. From 1988 to 1997, he was with the Robotics Laboratory, Shenyang Institute of Automation, Chinese Academy of Sciences, where he was an Associate Professor in 1995. From 1997 to 1998, he was a Research Assistant with the Department of Manufacturing Engineering and Engineering Management, City University of Hong Kong. He joined the Centre for Signal Processing, Nanyang Technological University, as a Research Fellow, in 2001. He is currently a Senior Scientist with Institute for Infocomm Research. He serves as an Editor of *Scholarpedia*. A paper he published in 2008 received the *Pattern Recognition* Journal Honorable Mention 2010. He was a recipient of the Chinese Academy of Sciences Award 1995, China, the A*STAR Borderless Award 2016, and the MTI Borderless Award 2016, Singapore. He has published widely, with four patents granted and over 70 publications in autonomous vehicle, computer vision, machine learning, and biometrics.



Lu-Bing Zhou received the B.Eng. degree from Beihang University, Beijing, in 2008 and the Ph.D. degree from the School of Electric and Electronic Engineering, Nanyang Technological University, in 2014. He joined the Institute for Infocomm Research, as a Research Scientist, in 2013. His expertise includes image processing, compute vision, machine learning, deep learning, robotics, and autonomous vehicles.