2015 IEEE 18th International Conference on Intelligent Transportation Systems

A Vision-Based Hierarchical Framework for Autonomous Front-Vehicle Taillights Detection and Signal Recognition

Zhiyong Cui
School of Software and Microelectronics
Peking University, PKU
Beijing, China
1201210546@pku.edu.cn

Shao-Wen Yang
Intel Labs, Intel Corporation
Taipei, Taiwan
shao-wen.yang@intel.com

Hsin-Mu Tsai
Department of Computer Science
and Information Engineering
National Taiwan University, NTU
Taipei, Taiwan
hsinmu@csie.ntu.edu.tw

***Abstract*—Automatically recognizing rear light signals of front vehicles can significantly improve driving safety by automatic alarm and taking actions proactively to prevent rear-end collisions and accidents. Much previous research only focuses on detecting brake signals at night. In this paper, we present the design and implementation of a robust hierarchical framework for detecting taillights of vehicles and estimating alert signals (turning and braking) in the daytime. The framework contains three-layers of processes, (i) detecting the vehicles in image; (ii) extracting taillight candidates using a clustering technique; and (iii) estimating rear-light signal states. The three-layer structure of the vision-based framework can effectively filter out noises in the image background and select correct pairs of taillights. Comparing to existing work addressing nighttime detection, the proposed method is capable of recognizing taillight signals under different illumination circumstances. The experiment results show our framework outperforms the state-of-the-art luminance-based method in different weather conditions during the daytime.**

I. INTRODUCTION

As reported by National Bureau of Statistics of China in 2014 [1], the total number of motor vehicles for civilian use reached 154.47 million, where as the road traffic death caused by collisions toll per 10 thousand vehicles was 2.22 persons. Thus, traffic safety is still a big issue for both countries and individuals. To avoid rear-end collisions, various detection systems has been introduced to autonomous systems, like driverless cars, advanced driver assistance systems (ADAS). Although radar-based [2] and laser-based [3] methods proved to be available for vehicle detention, vision-based methods for vehicle detection has received more attention recently.

In the context of intelligent transportation system (ITS), tracking the motion of vehicles is important to take actions proactively to prevent rear-end collisions and accidents. In terms of drivers, not only behavior but also intention can be reflected by appearance of vehicle, and, as a result, can be addressed by computer vision techniques. Other than the motions of vehicles, the taillight signals (turning and braking) of vehicles can reflect drivers' intentions more directly. Thus, it is reasonable and preferable to design a mobile tracking system providing the ability to detect vehicle taillights and estimate the signal states. Due to the improvement of computational capabilities of microprocessor and the development of computer vision techniques, tracking vehicles can even be realized on a single dashboard camera (dashcam) [4].

Fig. 1. Framework: hierarchical structure with 3 layers

Generally, related research on vision-based tracking or detecting algorithms can be devided into the following two categories:

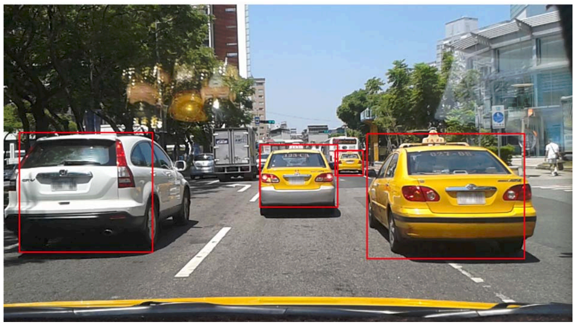
1. The one includes algorithms based on temporal information which can be used to track an object in a sequence of frames. Methods which can track multiple vehicles [5] even in the daytime [6] have been proposed. Meanwhile, some general tracking methods are utilized, such as particle filters [7] or Kalman filters [8], [9], [10].

2. The other includes algorithms based on frame features which analyze the relevant features, especially red color, within a single frame. Noisy pixels in the frame will be discarded by setting thresholds to a certain color [2], by using morphology methods [11], or by converting RGB to an alternative color spaces, like YCbCr [12], HSV [8], or YUV [4].

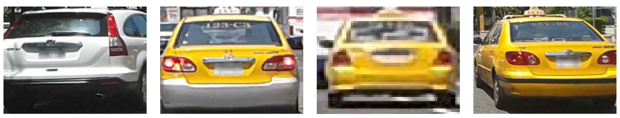
Some methods may combine algorithms from the above two categories to increase reliability and efficiency [4], [8]. But most research focusing on detecting taillight at night is based on methods of separating or tracking red pixels in the frames. When applying those methods under daylight circumstances, the noise in the background of the frame will make these methods unreliable. Although there are several daytime vehicle detection algorithms [4], [9], [10], they mainly utilize morphology methods to identify light pairs and track them with Kalman filter. As this method needs to compare the luminance of the taillight, it is hard to recognize the signal states in limited time before the autonomous system has to take proactive actions.

In this paper, we proposed a novel and robust hierarchical framework to detect taillights of vehicles and estimate alert

978-1-4673-6596-3/15 $31.00 © 2015 IEEE
DOI 10.1109/ITSC.2015.156



(a) Detected bounding boxes in original image



(b) Vehicles in the bounding box

Fig. 2. Detected bounding boxes and vehicles in a frame

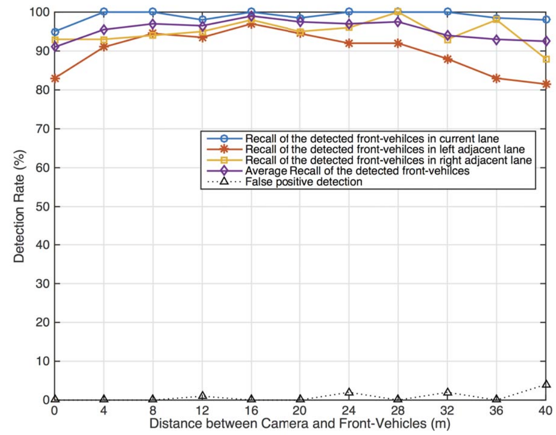


Fig. 3. Relationship between the detection rate of vehicles and the distance from camera to front vehicles

signals under different illumination circumstances based on our previous work [13]. The framework encompasses the following three stages, shown in Fig. 1: (i) detecting the vehicles in image; (ii) extracting taillight candidates by applying image processing method; and (iii) estimating rear-light signal states. The 3-layer structure of the framework is designed to increase the level of robustness. The key points of the framework, superior to existing work [4], [9], [10], [14], are listed as follows:

- Hierarchical structure: getting rid of noisy pixels and wrong taillight pairs with the help of hierarchical structure;
- One shot scheme: detecting and recognizing taillight signals with only one frame;
- Precise classification: classifying alert signal states into 3 classes, containing braking (**B**), turning (**T**), and no signals (**N**);
- Robustness detection: being able to detect taillight signals even the vehicle is swerving or changing lane;
- Multi-weather conditions: being capable for different illumination and different weather conditions;

The rest of this paper is structured as follows: Section II describes the proposed method. Section III illustrates the framework of our system, which can track rear lights of a vehicle and detect signals. Section IV presents the analysis of the experimental results followed by section V, which finally concludes the paper.

II. Proposed Method

In this paper, we present a novel and robust framework for detecting vehicle rear light signals, which can detect rear light signals in different illumination conditions. Our method is put forward as the following:

1) Detecting bounding boxes of vehicles with Deformable Part Model (DPM) [15], [16] to ensure finding correct pair of tail lights.
2) Preprocessing images of vehicle bounding boxes to extract candidates of the red color region and eliminate noises with a density-based clustering algorithm [17].
3) Extracting the feature of taillight by using sparse representation method and classifying the state of taillight signal [18], [19].

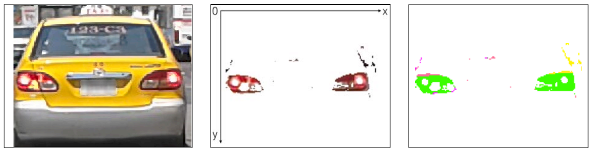
A. First Layer: Detecting Vehicle with DPM

To eliminate noisy pixels and select correct candidates of taillight pairs in the frames, we choose Deformable Part Model (DPM), which is a multi-scale object object detection method, to detect vehicles. DPM can be characterized by the combination of the following three parts:

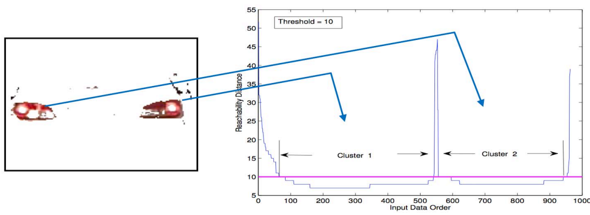
- Strong low-level features based on histograms of oriented gradients (HOG)
- Efficient matching algorithms for deformable part-based models in pictorial structures
- Discriminative learning with latent variables using latent SVM.

The frames recorded by the dashcam are the initial input of DPM. The model of DPM is trained from PASCAL 2007 dataset for cars and the output of DPM is a sequence of coordinates of the detected vehicle's bounding boxes in the input frame, as shown in Fig. 2(a). In general, DPM has an extremely high recall rate when detecting vehicles with different poses and colors. As described in our previous work [20], with the parameters of the dashboard camera and the positions of bounding boxes in the frame, we can approximately estimate the distance between the front-vehicle and the dashboard camera.

After 200 frames, containing multi-vehicles, are tested, the result shows that most of the distances from camera to front vehicles are distributed in the range of 5 - 40 meters. Meanwhile, even farther vehicles, which are very small in the frame, can be detected by DPM as shown by the 3rd vehicle in Fig. 2(b). Fig. 3 shows that the average recall rate of the



(a) Vehicle bounding box (b) Red pixels (c) Two largest clusters



(d) Two clusters generated by OPTICS, containing the points under the pink line

Fig. 4. Demonstration of extracting rear light regions by OPTICS

detected front-vehicles is above 90%, and the recall rate in the current lane, close to 100%, is obviously higher that of neighboring lanes. Also, the rate of false positive detection is almost 0 percentage. Thus, as the first layer of our framework, detecting vehicle's bounding boxes by DPM is good enough to be the precondition of the taillight recognition in the following steps.

B. Second Layer: Extracting Taillight Candidates

Although the first layer of the framework can detect the bounding box of a vehicle precisely, the bounding box may still contain noise (red pixels), like red ornaments or logos on the vehicle's surface shown in Fig. 4(a). Thus, in the second layer, noise in the bounding box must be eliminated when extracting correct taillight pair.

1) Color Space Conversion: As hue-saturation-value (HSV) color space is easier to adjust and manipulate the threshold parameters than red-green-blue (RGB) color space [9], we convert the pixels' color space from RGB into HSV. Since this phase is not threshold-dependent, we set loose thresholds, ranging from Pink-Red to Red-Orange, to extract red pixels as many as possible. After testing, the thresholds, presented in Table I, turn out to be able to select both dark and bright red colors.

TABLE I. LOOSE HSV THRESHOLD VALUES OF RED COLOR

	Minimum	Maximum
Hue (H)	331°	20°
Saturation (S)	0.2	1
Value (V)	0.05	1

2) Red Pixel Clustering: Most of red pixels in bounding box of vehicle gather into several clusters, in which the two largest ones usually belong to the regions of the two rear lights, as shown in Fig. 4(b). Thus, we use a density-based clustering algorithm, ordering points to identify the clustering structure (OPTICS) [17], to extract the two biggest clusters. The input of the OPTICS is the x and y coordinates of the red pixels (points) and the OPTICS algorithm generates an ordering of the points $\boldsymbol{o}$:{1..n} and corresponding reachability-distance $\boldsymbol{r}$:{1..n}. The 2D plot of reachability distance of the ordered pixels, illustrated in Fig. 4(d), can help us distinguish the clusters. As the denser the point gather, the lower reachability distance the point gets, the "valley" shapes in the plot of reachability distance represent the clusters with high density.

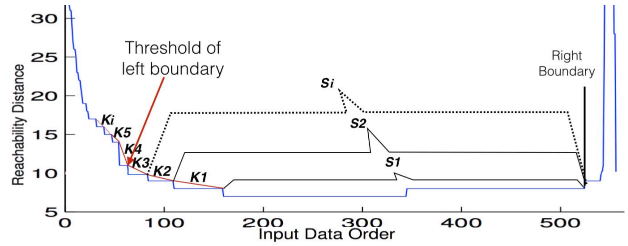


Fig. 5. Defining the left boundary of a taillight cluster in the reachability distance plot.

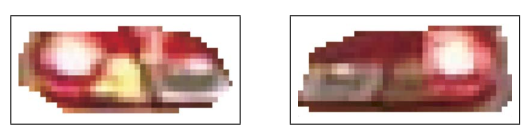
3) Taillight Bounding Box Extraction: Although we can identify the clusters in the reachability-plot easily, it is hard to define a global density threshold (which is graphically a horizontal line in the reachability plot, Fig. 4(d)). As the shapes and sizes are different for various vehicles, determining the thresholds (***T***) of clusters need to consider several aspects, like the cluster size (***S***), the gradient of reachability distance at the cluster boundary (***K***) shown in Fig. 5. According to results of our tests on 100 vehicles images, we get the parameters and use Bayes' theorem to determine the threshold:

$$P(T_j|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|T_j)P(T_j)}{p(\boldsymbol{x})} \tag{1}$$

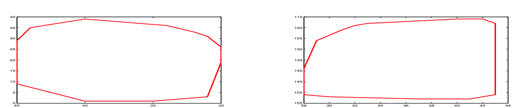
$\boldsymbol{x}$ represents the discrete variables, $\boldsymbol{x}_i = \{S_i, K_i\}$ and $p(\boldsymbol{x}) = \sum_{i=1}^{n} p(\boldsymbol{x}|w_j)P(w_j)$. We choose the T_i as the threshold subject to

$$T_i = \{T_i | i = \arg\max_i P(T_i|\boldsymbol{x})\} \tag{2}$$

After getting the pixels in the two clusters, we compute the convex hulls of the taillights, which is the precise bounding boxes of the taillights, shown in Fig. 6. With the bounding box of taillight, we can take some additional applications into practice: i)calculating the luminance of the taillight and ii) estimating the vehicle orientation.

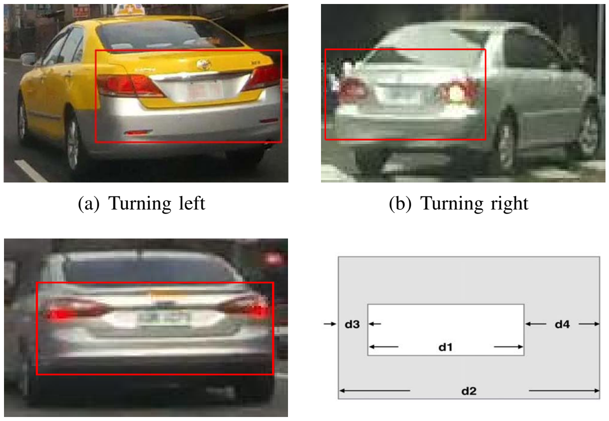


(a) Vehicle rear light polygon bounding box



(b) Convex hull polygon

Fig. 6. The left and right rear light convex hull. Lamp luminance is calculated as average luminance of pixels in the polygon.



(a) Turning left (b) Turning right

(c) Going ahead (d) Parameters, d1, d2, d3 and d4, determine the orientation

Fig. 7. Three vehicle orientation detecting results shown in (a), (b) and (c). (d) demonstrates the relative positions between inner and outer bounding box

4) Computing Luminance of Taillight Region: Computing luminance of taillight region need to calculate the luminance of each pixel, and then calculate the average value of all pixels' luminance in the convex hull. We compute luminance using the following formula according to the Rec. 709 primaries [21], where R, G, and B represent the red, green, and blue respectively:

$$Luminance = R \times 0.2126 + G \times 0.7252 + B \times 0.0722 \quad (3)$$

5) Vehicle Orientation Estimation: The position of the vehicle's taillight relative to the vehicle bounding box can help in estimating vehicle orientation. Comparing to other vehicle orientation detection methods, like using a tree-based classifer [22], our method needs nearly no additional computation after OPTICS is executed. And three values are critical to infer the orientation of vehicle (Fig. 7). The first one is the ratio *k*, calculated as :

$$k = \frac{d_1}{d_2} \quad (4)$$

The remaining two values are the horizontal distance between the leftmost boundaries of the two bounding boxes (d3) and the one between the rightmost boundaries (d4), shown in Fig. 7(d). Then, a simple decision tree with these parameters is used to estimate the vehicle pose. Despite the high-efficiency of this method, the method has the best performance only if the front-vehicle and the vehicle with dashboard camera mounted are in the same lane since the basis of this method is measuring the relative position and the relative angle to the camera.

C. Third Layer: Estimating the Taillight Signal State

Considering various circumstances, including vehicle color, natural light, rear light shape, and luminance, it is important to extract the features of taillight and surrounding pixels to correctly judge the state of the taillight, shown in Fig. 8. Sparse representation is a good choose for the framework to estimate the signal state as it has been proved that sparse representation is good at object detection and noise reduction by Hierarchical Matching Pursuit (HMP) [18], [19]. We design the method of 3rd layer based on HMP and add some original ideas. The image patches of the extracted taillights will be the input of the 3rd layer. The steps of the 3rd layer are shown below:

(a) No light (b) Braking (c) Turning (d) T & B

Fig. 8. Four states of vehicle rear light. T & B means Turning with braking

1) Learning Dictionary with K-SVD and Sparse Coding with OMP: To extract feature from the taillight image, learning a dictionary is the first step. K-SVD is a simple and efficient dictionary learning algorithm, which generalizes the idea of K-Means and updates the dictionary sequentially. Given a set of h-dimensional observations $Y = [y_1, \cdots, y_m] \in R^{h\times n}$ (image patches in our case), K-SVD learns a over-complete dictionary $D = [d_1, \cdots, d_m] \in R^{h\times m}$, in which d_i is called an atom, and an associated sparse code matrix $X = [x_1, \cdots, x_n] \in R^{m\times n}$. We find the best dictionary represent data samples Y as sparse compositions by solving:

$$\arg\min_{D,X} \|Y - DX\|_F^2 \qquad s.t. \|x_i\|_0 \le T_0 \quad (5)$$

where the notation $\|A\|_F$ denotes the Frobenius norm and T_0 is the sparsity level, which bounds the number of non-zero entries in the ith column of X, x_i. In the first stage, the over-complete dictionary D is fixed and only the sparse code matrix is optimized. Orthogonal matching pursuit (OMP) is introduced to compute the representation vectors x_i for each example y_i, by approximating the solution of

$$\arg\min_{x_i} \|y_i - Dx_i\|^2 \qquad s.t. \|x_i\|_0 \le T_0 \quad (6)$$

In the second stage, we update each column $k = 1, 2, ..., K$ in D by computing the overall representation error matrix

$$E_k = Y - \sum_{j\neq k} d_j x_T^j \quad (7)$$

where E_k is restricted by choosing only the columns corresponding to $\omega_k = \{i|1 \le i \le n, x_T^k(i) \ne 0\}$. We obtain E_k^R by applying SVD decomposition $E_k^R = U\Delta V^T$ and update the d_k and x_r^k with the first columns of U and $V\Delta(1,1)$, respectively. Then, we can get the sparse coding X of an input image Y.

2) Spatial Pyramid Max Pooling and Normalization: Spatial pyramid max pooling generates different level representations from sparse codes to reflect the real nature of the extracted features. For an image patch, I, The feature at level, U, can be represented as

$$F(I) = [max_{j\in I}|x_{j1}|, \cdots, max_{j\in I}|x_{jm}|] \quad (8)$$

The feature after spatial pyramid max pooling can be described as following:

$$F(I) = [F(I_1^1, \cdots, F(I_1^{V_1}), \cdots, F(I_U^{V_U}))] \quad (9)$$

where, V_u represent the subunit of the image patches in level u. Then, we normalize the feature to provide a limited range

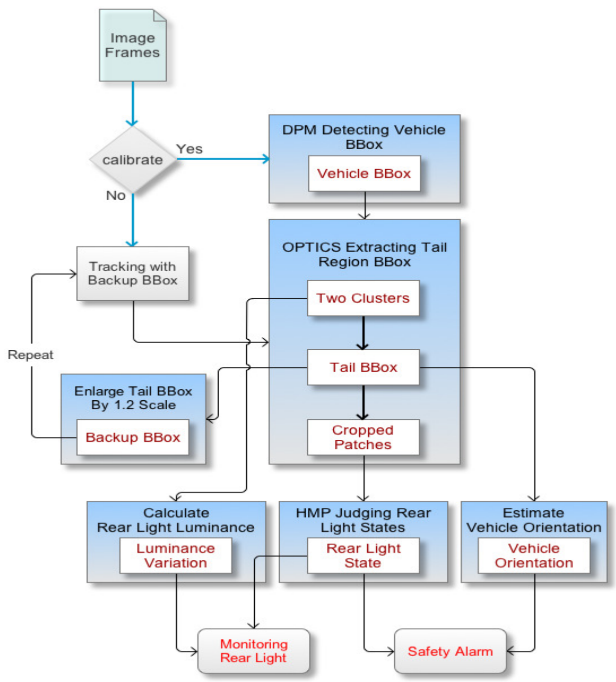
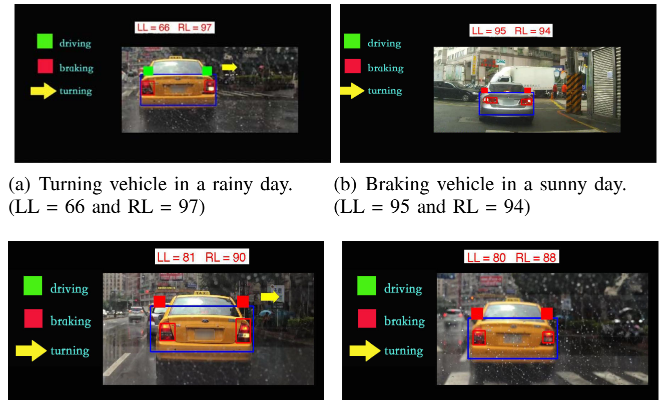


Fig. 9. Structure of the proposed framework



(a) Turning vehicle in a rainy day. (LL = 66 and RL = 97)

(b) Braking vehicle in a sunny day. (LL = 95 and RL = 94)

(c) Turning.(LL = 81 and RL = 90)

(d) Braking.(LL = 80 and RL = 88)

Fig. 10. Frames detected with luminance of rear light. LL and RL represent left luminance and right luminance respectively. (a) and (b) reveal normal outputs in sunny and rainy day. Taillights in (c) and (d) have similar level of luminance but with totally different signal states as our experiment tested.

of value. Normalized features will be the input for training or testing when classifying the taillight states by SVM.

3) Classifying Taillights before Training: The taillight signal's states can be basically divided into braking (**B**), turning (**T**), and no signals (**N**), whereas there are situations when the vehicle turning with braking. Thus, we consider this signal state of turning with braking as **TB**. As the state of left and right taillight signal may not be the same, we define a table of signal judging criterion, Table II, to determine the signal state after classification. For example, the **TB** is marked by the symbol of ⊗ in the table. By combining the signal states of two taillights, the detection accuracy can be improved.

D. Structure of the Framework

Fig. 9 shows the framework of the detecting system, which can track and detect the signal state of vehicle's taillight. Since the running of DPM and OMP algorithms consuming much time, we design a balance strategy of not running the DPM

TABLE II. CRITERION OF VEHICLE REAR LIGHT SIGNALS

		Rear Light State		
		No light	Braking	Turning
Turning Off	left	□		
	right	□		
Braking	left	△	□ ◇	
	right	◇	□ △	
Turning Left	left			◇ ⊗
	right	◇	⊗	
Turning Right	left	◇	⊗	
	right			◇ ⊗

on every frame. Due to the relative position of taillight in the frame will not change too much in half second or less time, we design a tracking method to mitigate the high computation time as depicted in the tracking with backup bounding box module in Fig. 9.

III. EXPERIMENT

In this section, we present experimental results obtained from the proposed method. Dataset for our experiment containing over 10000 frames is collected from vehicle dashboard camera.

A. Comparison with Existing State-of-the-art Methods

As we investigated, an existing state-of-the-art method (luminance-based method) has been adopted in [4], [9], [10] by tracking the red color regions in video frames and monitoring the variation of these regions' luminance in the daytime. Comparing to this method, we implement a similar but more precise method to compute the luminance of the taillight region by utilizing OPTICS, shown in Fig. 10. We find that the luminance-based method may not work in several scenarios, including cases when the image frames are over exposure, front-vehiles are swinging, detecting under rainy weather, or the luminance of taillights in both sides are asymmetric. As we can see in Fig.10, the ratio of left and right luminance in Fig. 10(d) does not reflect the actual state of the vehicle. And the ratios of LL/LR in Fig. 10(c) and Fig. 10(d) are nearly the same but the signal status are actually different.

To verify our statement, a short video is tested, in which the front-vehicle is changing lane with its right turning light shining. In this experiment, we record the luminance variations of both taillights, illustrated by blue and green lines in Fig. 11. The figure also presents the actual signals of the two lights, which are represented by yellow and orange lines, and the detected signals by our hierarchical methods indicated by red and purple lines. As we can see in Fig. 11, the right turn light is continually shining during the whole testing time and the brake light is turned on from 2.7sec to 7sec. The

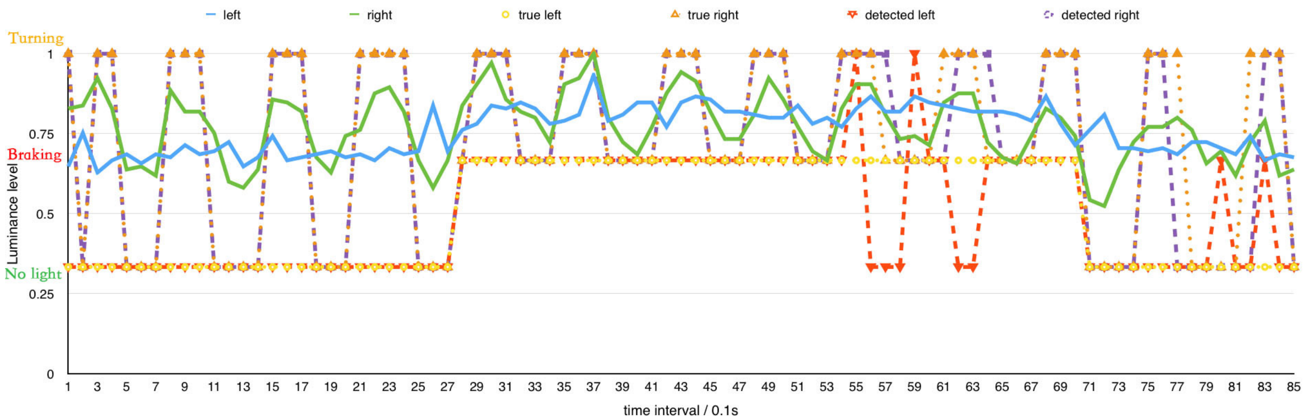


Fig. 11. Plot of signals when a car changing lane with its right turning light on. Blue and green lines represent luminance level, which are relative values compared to the largest luminance obtained in the experiment. The states of *Tuning, Braking and No light* are set at the trisector of Y-axis in the plot which has no relationship with the luminance values.

detected (red and purple) lines in the figure generated by our hierarchical metlhod is more similar to the ground truth (yellow and orange) lines.

However, if we determine the taillight states by using the green and blue lines, there is no mature and global model or threshold for detecting taillight signals. It is even harder to determine when the vehicle stops braking, especially in the case that both turning and braking lights are on. However, this problem can be solved by our method, as we can take the detected states of both lights into account to determine signal state. Although it is possible to adaptively learn the thresholds or models for the luminance-based method, it still delays the detection process. And since intelligent camera can change its settings frequently in response to luminance or environment changes (e.g., around 5s interval from 27 to 71 in X-axis in the Fig. 10), the luminance-based method will be even more difficult to be applied in real systems.

A more critical flaw of the luminance-based method revealed in this plot is the long time interval from a new turn signal generated to the signal detected. As the turn signal shine for 0.5 to 1 second a time, it take at least 1 second for detecting a new turn signal to rule out the possibility of braking or false alarm under ideal conditions. And in some more challenging cases, the time interval could be longer. However, the hierarchical method will not encounter this problem by detecting signal using only one frame with no need to other temporal information.

TABLE III. DETECTION RATE OF OUR METHOD

Real State(Column)	N	B	T	TB
N	94.8%	5.2%	0%	0%
B	13.5%	75%	0%	11.5%
T	8.8%	0%	91.2%	0%
TB	2.7%	2.7%	0%	88.6%

[1] N: No light; B: Braking; T: Turning; TB: Turning & Braking

TABLE IV. DETECTION RATE OF LUMINANCE-BASED METHOD

Real State(Column)	N	B	T	TB
N	79.2%	4.2%	16.6%	0%
B	0%	66.7%	0%	33.3%
T	15.8%	0%	84.2%	0%
TB	0%	19.0%	0%	81.0%

B. Taillight State Estimation Result

Table III summarizes the results of the proposed method. Each row in the matrix represents the detection rate of the four states, in which red figures is the true positives. *TB* in the table represents the state of turning with braking. Although, we only have 3 classes in the training set of HMP and do not have this class, this state could be detected by considering two light states, like one is turning and the other is braking.

As the experiment indicated, 3 states (*N*, *T* and *TB*) can be detected with accurate higher than 90%. After analyzing the testing dataset, we find it is the rainy weather at dusk that causes the *B* (braking) state with detection rate not as high as the others. At that time, most vehicles turn on their headlights and taillights of front-vehicle reflect the light in red color, which is very similar to braking signals. However, it is also possible to increase the detection rate of *B* state if more data can be collected and added into the training set.

Table IV presents the results of the luminance-based method. Comparing to our method, the detection rate of the each state generated by luminance-based method is obviously less than that of our method.

IV. CONCLUSION

In this paper, we propose a novel and robust framework for detecting turning and braking signals of vehicles, which can reliably operate in different illumination conditions. An object detection algorithm, DPM, is used to identify vehicles in images to improve robustness of tracking and accuracy of detection. We apply a density-based clustering method, OPTICS, to determine rear light regions precisely. As a sparse representation method, OMP plays a key role to classify rear light states. We present a computer vision based tracking system combining these methods, which operates in parallel with the prediction process to improve computational efficiency. Our experimental results reveal an average detection rate of above 90% which is superior to previous works.

Comparing to existing methods using luminance, our framework is superior as it can detect signals under different illumination circumstances. Our hierarchical framework is also able to recognize the taillight signal in one frame with no need to contrast the luminance level with previous frames. In the future research, we may focus on enhancing the recall and precision of vehicle recognition to obtain higher detection rate and reducing the time of detection to make the real-time system more efficient.

ACKNOWLEDGMENT

This work was supported in part by Ministry of Science and Technology of Taiwan, National Taiwan University, and Intel Corporation under Grants MOST 103-2911-I-002-001 and NTU-ICRP-104R7501, NTU-ICRP-104R7501-1.

REFERENCES

[1] N. B. of Statistics of China, "Statistical communiqué of the people's republic of china on the 2014 national economic and social development," February 26, 2015.

[2] P. Thammakaroon and P. Tangamchit, "Predictive brake warning at night using taillight characteristic," in *Industrial Electronics, 2009. ISIE 2009. IEEE International Symposium on*. IEEE, 2009, pp. 217–221.

[3] C.-C. Wang, C. Thorpe, and A. Suppe, "Ladar-based detection and tracking of moving objects from a ground vehicle at high speeds," in *Intelligent Vehicles Symposium, 2003. Proceedings. IEEE*. IEEE, 2003, pp. 416–421.

[4] A. Almagambetov, S. Velipasalar, and M. Casares, "Robust and computationally lightweight autonomous tracking of vehicle taillights and signal detection by embedded smart cameras," *Industrial Electronics, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2015.

[5] D. Koller, J. Weber, and J. Malik, *Robust multiple car tracking with occlusion reasoning*. Springer, 1994.

[6] K. She, G. Bebis, H. Gu, and R. Miller, "Vehicle tracking using online fusion of color and shape features," in *Intelligent Transportation Systems, 2004. Proceedings. The 7th International IEEE Conference on*. IEEE, 2004, pp. 731–736.

[7] Y.-M. Chan, S.-S. Huang, L.-C. Fu, and P.-Y. Hsiao, "Vehicle detection under various lighting conditions by incorporating particle filter," in *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*. IEEE, 2007, pp. 534–539.

[8] R. O'Malley, E. Jones, and M. Glavin, "Rear-lamp vehicle detection and tracking in low-exposure color video for night conditions," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 11, no. 2, pp. 453–462, 2010.

[9] M. Casares, A. Almagambetov, and S. Velipasalar, "A robust algorithm for the detection of vehicle turn signals and brake lights," in *Advanced Video and Signal-Based Surveillance (AVSS), 2012 IEEE Ninth International Conference on*. IEEE, 2012, pp. 386–391.

[10] A. Almagambetov, M. Casares, and S. Velipasalar, "Autonomous tracking of vehicle rear lights and detection of brakes and turn signals," in *Computational Intelligence for Security and Defence Applications (CISDA), 2012 IEEE Symposium on*. IEEE, 2012, pp. 1–7.

[11] Q. Ming and K.-H. Jo, "Vehicle detection using tail light segmentation," in *Strategic Technology (IFOST), 2011 6th International Forum on*, vol. 2. IEEE, 2011, pp. 729–732.

[12] S. Nagumo, H. Hasegawa, and N. Okamoto, "Extraction of forward vehicles by front-mounted camera using brightness information," in *Electrical and Computer Engineering, 2003. IEEE CCECE 2003. Canadian Conference on*, vol. 2. IEEE, 2003, pp. 1243–1246.

[13] Z. Cui, S.-W. Yang, C. Wang, and H.-M. Tsai, "On addressing driving inattentiveness: Robust rear light status classification using hierarchical matching pursuit," in *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*. IEEE, 2014, pp. 2243–2244.

[14] H. Emami, M. Fathi, and K. Raahemifar, "Real time vehicle make and model recognition based on hierarchical classification." *International Journal of Machine Learning & Computing*, vol. 4, no. 2, 2014.

[15] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester, "Cascade object detection with deformable part models," in *Computer vision and pattern recognition (CVPR), 2010 IEEE conference on*. IEEE, 2010, pp. 2241–2248.

[16] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[17] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, "Optics: Ordering points to identify the clustering structure," in *ACM SIGMOD Record*, vol. 28, no. 2. ACM, 1999, pp. 49–60.

[18] L. Bo, X. Ren, and D. Fox, "Hierarchical matching pursuit for image classification: Architecture and fast algorithms." in *NIPS*, vol. 1, no. 2, 2011, p. 6.

[19] L. Bo, X. Ren, and D. Fox, "Multipath sparse coding using hierarchical matching pursuit," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013, pp. 660–667.

[20] Z. Cui, C. Wang, and H.-M. Tsai, "Characterizing channel fading in vehicular visible light communications with video data," in *Vehicular Networking Conference (VNC), 2014 IEEE*, Dec 2014, pp. 226–229.

[21] I. T. Union, *Parameter values for the HDTV standards for production and international programme exchange*. Geneva: Electronic Publication, 05 2002.

[22] M. Gabb, O. Lohlein, M. Oberlander, and G. Heidemann, "Efficient monocular vehicle orientation estimation using a tree-based classifier," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*. IEEE, 2011, pp. 308–313.