Vehicle Detection and Scale-Adaptive Tracking Using Tail Light Segmentation

Ming Qing, Sung-Min Yang and Kang-Hyun Jo
University of Ulsan
Ulsan, Korea
qing@islab.ulsan.ac.kr
acejo@ulsan.ac.kr

Abstract—**This paper proposes a vehicle detection and tracking system based on forward looking CCD camera. Vehicle tail light position is used to locate vehicle candidates, and distance between tail light pairs allows to adjust vehicle tracking window size. In vehicle detection step, a back propagation neural network (BPNN) trained by Gabor feature set is used. BPNN verifies vehicle candidates and ensures system robustness. In vehicle tracking step, it also uses mean shift tracking algorithm using color feature space. Tail light pairs which are determined by detection result are used to adapt tracking windows size. In our experiments, the proposed algorithm showed 84% accuracy in vehicle detection. Furthermore, its performance makes it applicable for real applications.**

Keywords-vehicle detection; vehicle tracking; color segmentation; mean shift

I. INTRODUCTION

An increasing number of car accident happened everyday, which has drawn great attention from the public, almost every minute, at least one people dies in vehicle crash [1]. Recently, many new technologies are developed to avoid or mitigate vehicle accident through sensing the significance and nature of danger which aim to improve human safety, such as intelligent transportation, intelligent driver assistance systems.

This paper proposes a vision based vehicle detection and real-time tracking system. Vehicle tail light information is employed in the entire algorithm. In vehicle detection step, many features can be selected. For instance, extracted shadow information between the vehicle and road [2,3], or edge histogram information [4,5]. However, the shadow information is very sensitive to illumination and edge histogram is influenced by the environment, which makes the algorithm quite difficult to implement. In this paper, vehicle tail light location is used to generate vehicle candidates, these candidates are verified by BP neural network trained by Gabor feature set. Vehicle detection robustness is guaranteed by BPNN classifier [6]. In vehicle tracking step, mean shift algorithm is used. The kernel window bandwidth is very essential in mean shift tracking algorithm, it determined the sampling quantity of mean shift iteration. Furthermore, it related to the tracking windows size. Original mean shift object tracking uses fixed window size. In [7] kernel bandwidth increment for scale adjustment is proposed, and this approach process one frame using 3 different bandwidth kernels, which will increase system runtime. In order to solve multiple scales tracking problem, our algorithm considers the distance between light pair which is determined by vehicle detection result for tracking windows adjustment. The main architecture of vehicle detection and tracking system is described in figure 1.

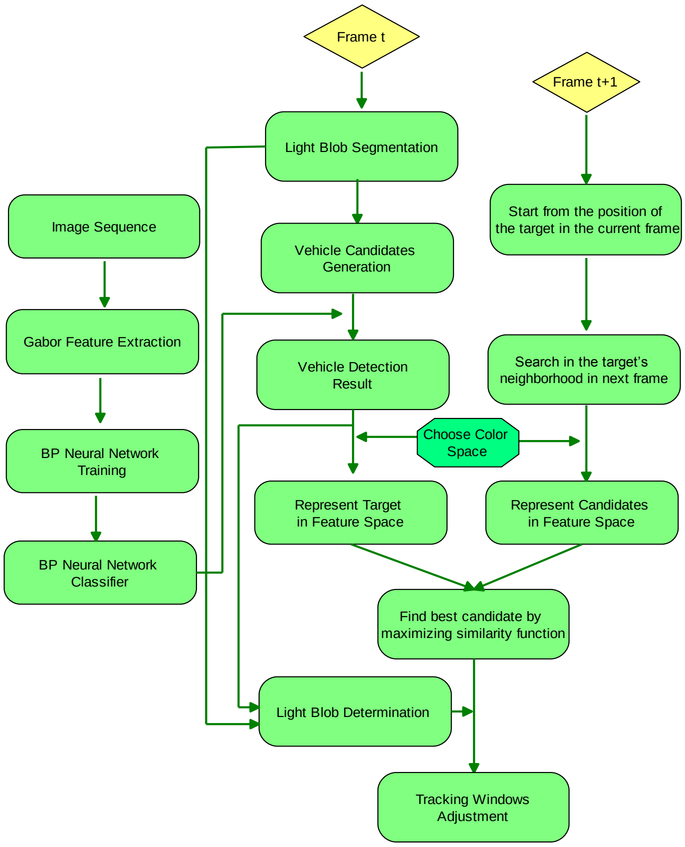
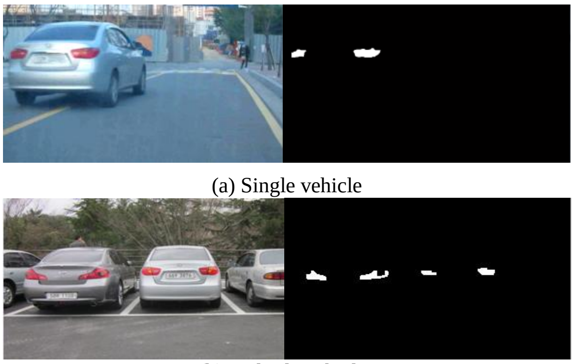


Figure 1. system architecture

II. VEHICLE DETECTION

A. *HSV color model based light blob segmentation*

HSV stands for hue, saturation and value, and is also often called HSB (B stands for brightness); these are often used by human for color object description. Based on HSV color model, red light blob can be detected easily with appropriate threshold. Threshold was obtained by 68 vehicle tail light images HSV statistical value. After that, morphological method is employed to remove some small points which come from color segmentation. Figure 2 shows vehicle tail light blob segmentation results for single (a) and multiple vehicles (b).



(a) Single vehicle

(b) multiple vehicles

Figure 2. Vehicle rear light blob segmentation

B. Light pairs candidates generation

The main goal of this step is to find the corresponding light pairs. Vehicle tail light pair candidates can be extracted by the following conditions. See figure 3, denote the left tail light as c1 and right side tail light as c2.

Figure 3. Vehicle tail light pair

a) The distance between c_1 and c_2 should be limited to a range.

$$w_{\min} \le w_{c_1c_2} \le w_{\max} \qquad (1)$$

w_{c1c2} represents the width between vehicle light pair. In this paper, w_{min} and w_{max} equal to 5 pixel length and 100 pixel length respectively, the thresholds selection depend on the image size.

b) The position of the corresponding light pair almost locate the same horizontal line.

$$\left| h_{c_1} - h_{c_2} \right| \le d \qquad (2)$$

Where h_{c1} and h_{c2} represent the height of c_1 and c_2 respectively, d is a constant, 6 pixels were chosen in this process.

Due to the parameters which are mentioned above step, this algorithm is able to extract vehicle candidates based on tail light distribution. However, in some cases where cars are next to each other, as shown in figure 2(b), this process will extract non tail light pairs. As shown in figure 4, each rectangle determines one light pair.

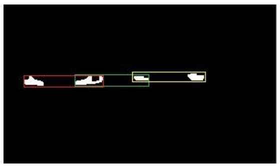


Figure 4. Light pair candidate

Because the proportion of the same type vehicle is almost the same, therefore, based on location of pairs of lights and distance information, vehicle candidates can be extracted as figure 5 shows:

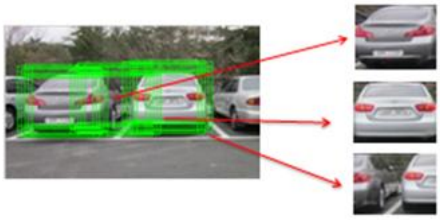


Figure 5. Vehicle candidates generation

C. Gabor feature extraction and training process

To obtain vehicle feature for BPNN training process, Gabor feature is investigated. Gabor feature can effectively describe local feature of image with different direction and scales. Gabor transform [8] can be formulated by the following equation:

$$G(\mathrm{x,y},\omega,\sigma) = \frac{1}{2\pi\sigma_x\sigma_y} \exp^{-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}\right)} \exp^{-j\omega(x+y)} \qquad (3)$$

This paper chose 8 orientation and 5 scales for Gabor filters to build the database.

In vehicle verification step, back propagation neural network is used. BPNN is applied extensively in practical application, in particular, superior effectiveness in model recognition [9]. It can identify vehicle in vehicle candidates, non vehicle candidates are dislodged, In this case, the robustness of vehicle detection is guaranteed by a strong BPNN classifier. Figure 6 shows vehicle detection result example.

Figure 6. Vehicle detection examples

III. REAL-TIME VEHICLE TRACKING

In order to track detected vehicles in video, this system implements mean shift algorithm. It is a nonparametric statistical method for seeking the nearest mode of point sample distribution [10] which has been proven to be effective for object tracking.

A. *Color feature space representation*

In the current frame, the detected vehicle can be represented based on RGB color model, see figure 7 (b). We assume the target color model can be split into several uniform histogram bins; target vehicle can be represented in RGB color space by the color probability density function (PDF) q which is calculated by the following equation, this paper used 16x16x16 bins, the result is shown in figure 7(c).

$$q_u = C\sum_{i=1}^{n} g(\|\frac{x_i}{h}\|^2)\delta_{ui} \tag{4}$$

Where g is kernel function, C is normalization factor, x_i represents target location, δ is the Kronecker delta function, u=1…m.

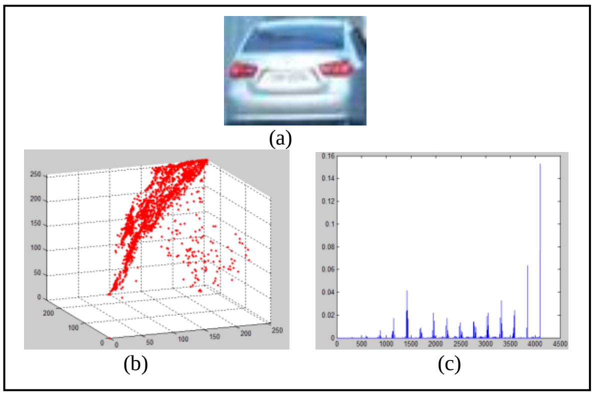


Figure 7. (a) Target vehicle, (b) RGB color space, (c) color PDF

In the next frame, starting from the position of the target in the previous frame, the same process is applied. The candidate color PDF is defined as p. In order to find the best matching candidate in the target's neighborhood, similarity function between q and p is used.

B. *Similarity function*

To define similarity, Bhattacharyya coefficient is used in this paper. It is an approximate measurement of the amount of overlap between two statistical samples. This coefficient can be used to describe the similarity of two discrete and normalized distributions. In the tracking system, $p(y)$ is the distribution of the candidate object at position y. So the Bhattacharyya coefficient function can be formulated as [11]:

$$\rho[p(y),q] = \sum_{u=1}^{m}\sqrt{p_u(y)q_u} \tag{6}$$

Best candidate can be found by maximizing similarity function.

C. *Mean shift vector*

Mean shift is a non-parametric feature-space analysis technique. Let's denote model location as y_0 and candidate location as y. The similarity function can be linearized by the Taylor series as:

$$\rho[p(y_0,q)] \approx \frac{1}{2}\sum_{u=1}^{m}\sqrt{p_u(y_0)q_u} + \frac{1}{2}\sum_{u=1}^{m} p_u(y)\sqrt{\frac{q_u}{p_u(y_0)}}$$
$$where: \quad p_u(y) = C_h \sum_{b(x_i)=u} k\left(\left\|\frac{y-x_i}{h}\right\|^2\right) \tag{7}$$

Where C_h is a normalization coefficient and h is a kernel bandwidth. From the above equation, the first term is independent of candidate location y; the second term is kernel probability density estimation with profile function k. So the similarity function maximization can be realized by the following iteration.

$$y_{k+1} = y_k + \frac{\sum_{i=1}^{n_h}\omega_i(x_i - y_k)k'\left(\left\|\frac{y_k - x_i}{h}\right\|^2\right)}{\sum_{i=1}^{n_h}\omega_i k'\left(\left\|\frac{y_k - x_i}{h}\right\|^2\right)}$$
$$where \quad \omega_i = \sum_{i=1}^{m}\sqrt{\frac{p_u}{q_{u(y_k)}}} \tag{8}$$

x_i (i=1,…,n_h) denotes the candidate spatial coordinates, h denotes the scale of the kernel. The best candidate which is most similar to the detected vehicle can be sought by mean shift iterations. The mean shift iteration example is shown in figure 8.

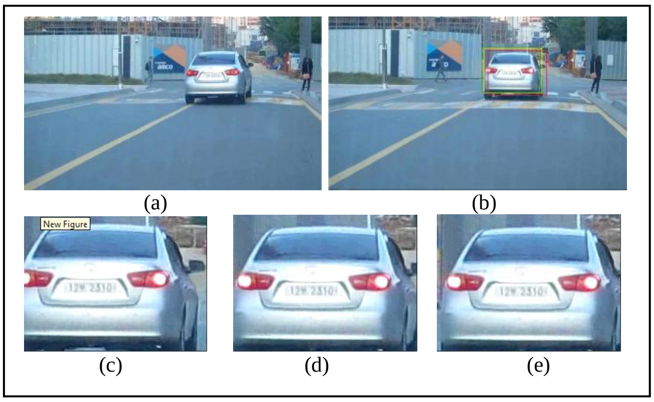


Figure 8. Mean shift iteration

Figure 8 (a) and (b) is frame t and frame $t+1$ respectively. The red rectangle is the first iteration, it starts from initial position which is target vehicle position at frame t, and the candidate is shown in figure 8(c). Yellow rectangle and green rectangle represent the forth iteration and ninth iteration respectively, the candidates are shown in figure 8 (d) and (e), and the ninth iteration is the final iteration which means figure 8 (e) is the best candidate for matching with the target vehicle.

D. *Scale adjustment*

The size of the target vehicle varies while the distance between target vehicle and camera change. Consequently, window scale adjustment is essential. In order to obtain a dynamic tracking window, distance between corresponding light pairs is investigate. Light pairs are segmented in vehicle candidate generation step, and it can be finally determined in vehicle verification step by trained BPNN. After apply vehicle tracking process, the best candidate is obtained. Based on this candidate, the window scale is enlarged 10%, light blob is detected by color segmentation on HSV space which has been introduced in section 2 A and B. In order to determine the corresponding light pair, the following constraints is proposed, c_1 and c_2 represent left and right light which is determined before. c_1' and c_2' represent left and right light which is detected after tracking process.

a) The area of the corresponding light is almost proportional

$$\frac{S(c_1)}{S(c_1')} \approx \frac{S(c_2)}{S(c_2')} \tag{9}$$

b) The shift of the corresponding light's position almost same

$$P(c_1) - P(c_1') \approx p(c_2) - P(c_2') \tag{10}$$

After corresponding light pairs are found, the distance between lights which is determined in vehicle detection step is defined as d_0, the distance between light which is detected after mean shift iteration is defined as d_t, so the size of the tracking window at frame t can be obtained by following equation:

$$L_t = \frac{d_t}{d_0} L_0 \tag{11}$$

L_0 is initial window size, L_t represents the window size at frame *t*.

IV. EXPERIMENTS

In this section, the experimental results obtained for the proposed method is laid out. The algorithm implemented by MATLab 7.8.0(R2009a), computer processor is Intel(R) Core(TM) 2 Quad CPU 2.66 GHz, RAM 3.00 GB. 104 images are tested for vehicle detection; vehicle movement video is tested for combine with real-time vehicle tracking. The images sequence which comes from vehicle movement video has 175 frames of 180x320 pixels for each. The target vehicle histogram has been derived in the RGB space with 16x16x16 bins. The detected vehicle window size was 78x102 pixels. The vehicle detection result is shown in figure 9.

Figure 9. Vehicle detection result at 1st frame

The tracking algorithm start from the 1st frame, the tracking results are presented in figure 10.

Figure 10. Vehicle tracking results. The frames 25, 55,95,100,143,175

These above results show that the proposed system is effective for vehicle detection and real-time tracking in the day time. The tracking window scale is able to self-adapt to the changing of the target vehicle size. Vehicle center position and tracking window size is displayed in top left corner of each frame.

Figure 11 shows the number of mean shift iteration for each frame, from frame 103 to frame 125, because the vehicle position did not shift in horizontal direction, just scale change with the distance variety, so the best matching candidate is sought only used one time iteration.

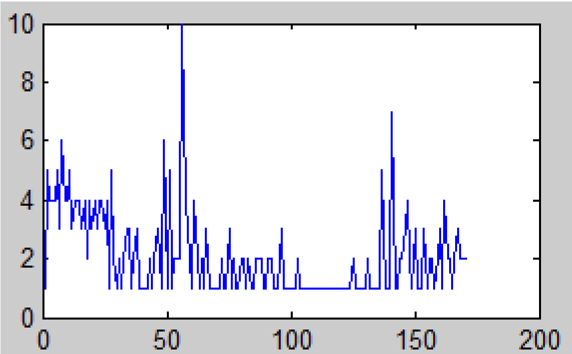


Figure 11. The number of mean shift iteration

V. CONCLUSIONS

An effective system for vehicle detection and vehicle tracking is presented in this paper. Vehicle tail light detection is used in the proposed algorithm, not only for a vehicle candidate generation, but also for vehicle tracking window adjustment. Back propagation neural network which is trained by 8 orientation and 5 scales Gabor feature set is used in vehicle candidate verification step. Mean shift tracker is implemented in vehicle tracking step.

The result shows that this system has good performance for vehicle detection and tracking, especially for target vehicle nearby. When target vehicle is very far from camera (more than 50m), the detection result performance is not quite good, because in that case, the color information influence from illumination easily, and the tail light feature is not strong enough. Also in vehicle tracking process, because of view angle of the camera, the tracking window may not be perfect matching with target vehicle.

For future work, it is necessary to utilize any other features which are stronger and more suitable for vehicle representation both in detection and tracking algorithm.

REFERENCES

[1] Zehuang Sun, George Bebis, Ronald Miller “On- Road Vehicle Dection: A Review,” IEEE Transactions on Pattern Analysis And Machine Intelligence, Vol. 28, No. 5, pp694-711,May 2006.

[2] Jeong-Gwan Kang, Se-Young Oh, “Real Time Multiple Vehicle Detection Using Neural Network with Local Orientation Coding and PCA,” 3rd ed., vol. 2. Oxford: Clarendon, pp68–73, 1892.

[3] Sungji Han, Youngjoon Han and Hernsoo Hahn , “Vehicle Detection Method Using Haar-Like Frature on Real Time System,” World Academy of Science, Engineering and Techonology 59, 2009.

[4] Z.Sun, R.Miller, G.Bebis, and D.DiMeo, “A Real-Time Precrash Vehicle Detection System,” 6th, IEEE Workshop on Applications of Computer Vision, December 03 - 04, Orlando, Florida, 2002.

[5] C.Goerick, D.Noll and M.Werner, “Artificial Neural Networks in Real-Time Car Detection And Tracking Applications,” *Pattern Recognition Letters*, Vol. 17, pp335-343, 1991.

[6] Qing Ming, Kang-Hyun Jo, “Vehicle Detection Using Tail Light Segmentation,” International Forum on Strategic Technology, August 22-24, Harbin, 2011.

[7] Comaniciu D, Ramesh V,Meer P. “Kernel-Based Object Tracking,” IEEE Trans.on Pattern Analysis and Machine Intelligence, Vol.25,No.5, pp564-575, May 2003.

[8] Tai Sing LEE, “Image Representation using 2D Gabor Wavelets,”IEEE Transactions on Parrern Analysis and Machine Intelligence, Vol 18,No.10, pp959-971,October 1996.

[9] C.Goerick, D.Noll and M.wener, “Artificial Neural Networks in Real-Time Car Detection And Tracking Applications,” Machine Vision and Applications, Vol 12, pp69-83, 2000.

[10] Cheng, Yizong. "Mean Shift, Mode Seeking, and Clustering," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.17,No.8, pp790-799, August 1995.

[11] Dorin Comaniciu, Visvanathan Ramesh, P eterMeer, "Real-Time Tracking of Non-Rigid Objects using Mean Shift, " IEEE Conference on Computer Vision and Pattern Recognition, June 13 -15, Hilton Head, SC, USA, 2000