

A Robust Framework for Vehicle Detection and Tracking Based on Particle Filter

Huihui Liu^{1,a}, Yong Liu^{2,b} and Haiqing Du^{3,c}

¹Beijing University of Posts and Telecommunications, Beijing, China

^adouzi_0407@163.com, ^bliuyo@bupt.edu.cn, ^cduhaiqing@bupt.edu.cn

Keywords: Vehicle detection/tracking; Foreground extraction; Particle filter; Histogram of oriented gradients; HSV color space; Occlusion segmentation;

Abstract. Vehicle detection and tracking have been promising application in traffic surveillance and vehicular network. However, the vision-based approach still remains a challenging task due to the problems of illumination variation, shadow and occlusion. In this paper, we propose a robust framework mainly concatenates on two aspects: adaptive vehicle detection with shadow removal, and vehicle tracking with occlusion handling. Firstly, for vehicle detection stage, an improved ViBe algorithm with ghost suppression is adopted to extract moving vehicle region. Then moving shadow is removed by integrating improved color with texture feature. Then, aiming to achieve the multi-vehicle tracking, we propose an enhanced histogram of the oriented gradient combined with HSV color space based on particle filter (ECHO GPF). Finally, we employ the occlusion detection and occlusion segmentation to refine our system, which are based on one-dimensional maximum entropy and the least square ellipse fitting. Experiments on popular datasets show that our proposed system has a good effectiveness, e.g., the accuracy is 95% on the vehicle tracking.

Introduction

The vehicle detection and tracking play a vital role in many vision-based intelligent transportation system applications. As existing gradual or sudden illumination changes, occurrence of shadow and vehicle occlusion, there are great difficulties among vehicle detection and tracking. We aim to design a robust framework, which could handle these problems. Our proposed system is consisted of background modeling, particle filter, one-dimensional maximum entropy and features like HSV color space, local texture, and histogram of oriented gradient.

Background modeling [1, 2] is the most commonly method used in the task of real-time detection. Compared with the GMM algorithm [4] and codebook algorithm [5], the visual background extractor [3] has got more attention due to its high precision, and fast processing speed. Li *et al.* [6] propose an improved ViBe algorithm that could suppress ghost well. As the shadow could cause objects merging, the distortion of shape and tracking loss, Cucchiara R *et al.* [7] use HSV color space, and Javed and Shah [8] apply the gradient direction to detect the shadow. However, there exists some shortcomings for the methods, which are based on HSV color space, i.e., some parameters should be adjusted depending on different traffic environments and single feature can't always work well. Therefore, we integrate the local texture feature with an improved non-parametric feature of HSV color space to remove shadow.

After detected the interest objects, we present a new appearance observation likelihood model in which the similarity of appearance between targets with a template is calculated. Feature is indispensable for appearance modeling. Sugimura *et al.* [9] uses color histogram for crowd tracking. Dalal *et al.* [10] apply histogram of oriented gradient in human detection. We combine enhanced the HOG feature with color histogram for precision and complementary description of feature. A good appearance model is of great aid to the task of tracking.

As the occlusion could result large tracking error, it has been one of the main research topics in video target tracking field for a long time. In general, occlusion is segmented by three kinds of methods: feature-based [12], model-based [13], and contour-based [14]. The feature-based approach usually costs too much time and makes vehicle tracking hard to be real-time. Kamijo *et al.* [13] prop-

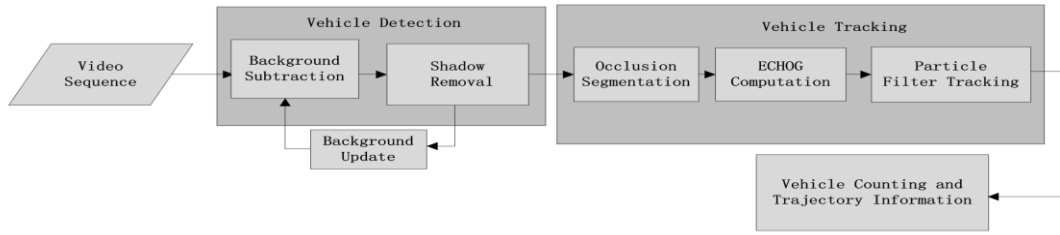


Fig. 1. The framework for vehicle detection and tracking.

used a solution based on random field models. However, the model-based method introduces unknown variables and usually could not adapt to the changed environment. The contour-based method could be failed if the foreground is incomplete. Comaniciu *et al.* [15] proposed a new object appearance based on mean-shift for partial occlusion problems. Wu *et al.* [16] proposed a partial occlusion detection approach based on the reduction of particle weights within the occluded regions. Many related occlusion segmentation methods are based on specific tracking algorithm. In this paper, we propose an adaptive occlusion segmentation method, which can handle multi-vehicle occlusion and be applied to other tracking framework. Our framework is based on the well-known particle filter [11], which is robust and flexible, and which has the ability to deal with the non-Gaussian and nonlinear models. As shown in Fig. 1, our system consists of the following modules: vehicle detection (ViBe background modeling, update and shadow removal), vehicle tracking (occlusion detection and segmentation, ECHOG appearance model, and particle filter tracking), and access to traffic information.

Vehicle Detection and Tracking Framework

Adaptive Background Modeling for Vehicle Detection. The algorithm of background modeling ViBe (visual background extractor) introduces randomization. The algorithm mainly comprises of three steps. The first step is to initialize the model using the strategy of random selection. The second step is to update the background model. The third step is to randomly propagate the pixel of the background image to its neighborhood based on the principle of pixel spatial relativity.

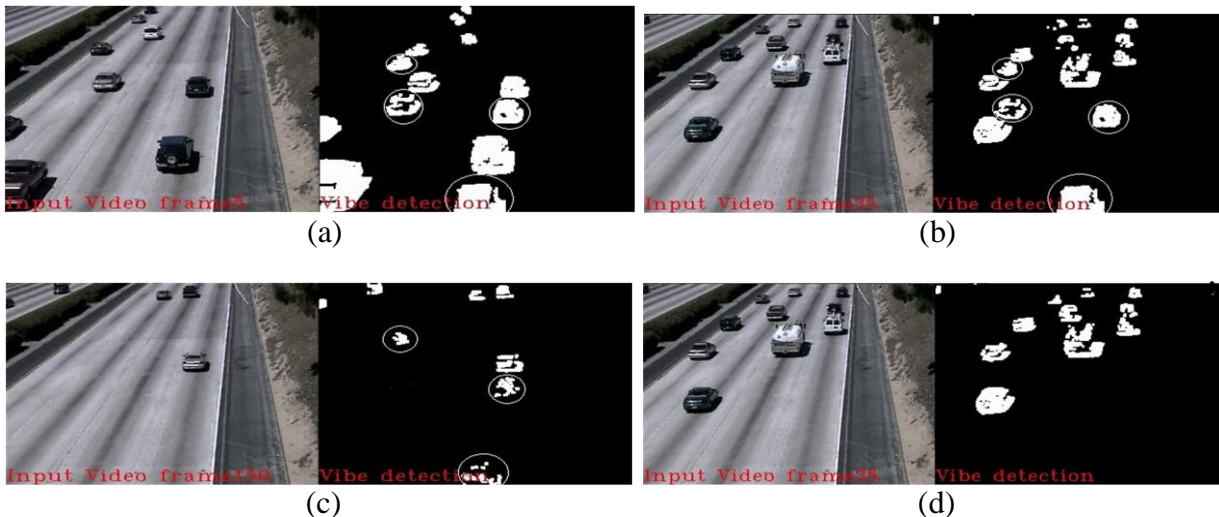


Fig. 2. The results of foreground extraction: (a) Frame 5 with ViBe detection, (b) Frame 35 with ViBe detection, (c) Frame 150 with ViBe detection, (d) Frame 35 with improved ViBe detection.

How to initialize the background model is a significant aspect for vehicle detection. There exists two ways to initialize in the previous method. The one method which uses the first frame of the video sequence to initialize the background model and this method achieves fast speed of initialization. However, the ghost will appear when the target exists in the first frame. In this case, the real

background pixel could not match the background model once the target left and this region becomes ghost. Therefore, we adopt multi-frame initialization algorithm which is based on ViBe detection [6]. The improved algorithm can suppress ghost well. An intuitive description is shown in Fig. 2. Fig. 2(c) shows the result for original ViBe algorithm, and Fig. 2(d) shows a better result which is obtained by our method.

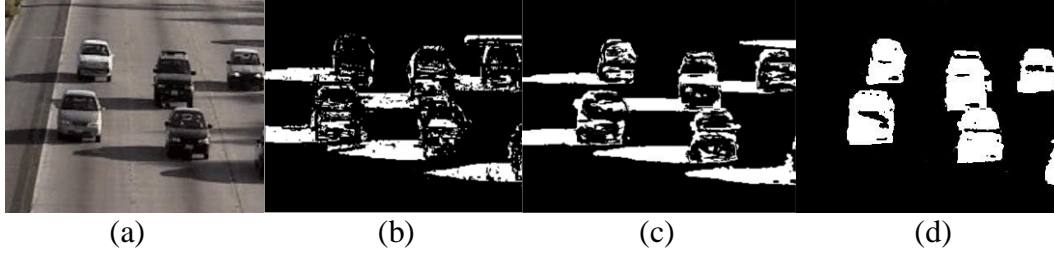


Fig. 3. Example of shadow detection for foreground extraction. (a) The original frame. (b) Mask of shadow using LBP feature. (c) Mask of shadow using improved HSV color space. (d) Foreground image.

Moving Shadow Detection Integrating Improved Color and Texture. Using the background modeling, each pixel could be classified into two classes: foreground and background. However, the foreground image is likely to contain the shadow. It causes incorrect object information and may lead to occlusion. Therefore, shadow detection is significant during tracking. Many previous studies have been discussed about the shadow elimination. However, the existing models, which are based on HSV color space or based on texture, i.e., LBP, all have some drawbacks. For example, if the color between moving object with background is similar, the region of the object may be detected as shadow mistakenly. For the most methods, the threshold parameters should be constantly optimized [15]. Therefore, we combine the feature of LBP and improved non-parametric HSV color space. The Fig. 3 is an intuitive description for the shadow detection.

In the region covered by shadow, the change of luminance is bigger than hue and saturation. The mask of shadow C_t^{HSV} is calculated by Eq. 1.

$$C_t^{HSV}(x,y) = \begin{cases} 1, & \text{if } 0 \leq \frac{I_t^V(x,y)}{B_t^V(x,y)} \leq 1 \wedge |I_t^h(x,y) - I_{t-1}^h(x,y)| \leq 1 \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

where $I_t^V(x,y)$ and $B_t^V(x,y)$ are pixel value of luminance channel for the given foreground image and background image of frame t . We utilize the $I_t^h(x,y)$ between two adjacent frames to discriminate shadow. The advantage of this method is that there are no parameters in the above process. Then, we use LBP operators to compute the texture mask L^{LBP} . Texture similarity ρ between the regions of shadow with background is measured by histogram intersection using the Eq. 2.

$$\rho = \sum_{n=0}^{N-1} \min(h_n^I, h_n^B) \quad (2)$$

where h_n^I denotes histogram of current frame, and h_n^B denotes histogram of background image. Once obtaining the likelihood, the foreground pixels could be classified into shadow and target.

$$L_t^{LBP}(x,y) = \begin{cases} 1, & \text{if } \rho \geq T_{LBP} \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

where T_{LBP} is set by the user according to actual video surveillance environment. L^{LBP} is shown in Fig. 3(b) and C^{HSV} is shown in Fig. 3(c). To get the result of shadow segmentation, we need to AND C^{HSV} to L^{LBP} . Certain areas of vehicles are detected as shadow mistakenly. So we refine the final mask though connected-components analysis. Fig. 3(d) shows the result of foreground image.

ECHOGPF Vehicle Tracking Algorithm. The output of the vehicle detection is a binary object mask which is used to perform tracking. Tracking is a consecutive task which aims to search the best-match object in the current frame and update feature of the template for an accurate match in the subsequent frame. Our proposed method, ECHOGPF, mainly consists of three parts. The first part is color histogram and enhanced HOG (ECHOG). The second part is particle filter tracking over time based on ECHOG appearance model. The third part is the appearance template matching used for checking whether the candidate bounding box matches the target in previous frame or not. In this paper, the template is progressively updated at each time instant.

ECHOG Appearance Model. Color histogram represents the distribution of colors in an image. It is the most popular appearance model in multi-object tracking system because of its effectiveness on obtaining statistical information. We compute a 3D HSV color histogram of concerned vehicles. One point worth noting is that the number of bin should be balanced because of bad discriminant effect when being too large or small. In our model, we choose 8 bins for h-channel, 12 bins for s-channel, and 3 bins for v-channel. Therefore, the total number of elements of HSV color histogram descriptor, which is used to compute similarity distance, is 288. However, the color histogram has a drawback due to the loss of the spatial information. Therefore, we improve the appearance model through combining the enhanced HOG descriptor [16].

Histogram of oriented gradient (HOG) is one kind of gradient descriptor, which is usually employed to image processing and object detection. Firstly, color-image is preprocessed by graying. We compute gradient by convolving gray image with a vertical kernel $[-1,0,1]^T$ and a horizontal kernel $[-1,0,1]^T$. The size of all object image patches is resized to 32×32 . Next, the image is divided into small cells with the size of 8×8 . The orientation of gradient ranged between 0° with 180° is uniformly divided into nine bins. In each cell, we compute the magnitude of gray value in all bins. A block is formed by 4 each close cells. So there are 36 bins in one block and nine blocks in one image. The nine blocks forma 324-dimension feature vector that is used as a HOG descriptor. The RDHOG descriptor used for enhancing the description of the object reflects difference between the central block and its neighbor blocks. The difference is computed by the Eq. 4.

$$RD_j(i) = S_{\text{center}}(i) - S_j(i), \quad j = 1,2 \dots \dots 9, \quad j \neq \text{center} \quad (4)$$

where $S_{\text{center}}(i)$ is i th bin in the central block and $S_j(i)$ is i th bin in j th block. $RD_j(i)$ is the i th bin of RDHOG descriptor in j th block. In this way, 288 bins make up a 288-dimension feature vector. By combining 3D HSV histogram with HOG and RDHOG descriptor, the model becomes more robust. It can not only work well in normal light condition but also the low contrast environment. The above descriptors are integrated into the particle filter framework for vehicle tracking.

Particle Filter Framework. Particle filter provides a probabilistic framework for recursive dynamic state estimation [17]. The goal is to estimate the posterior distributions of state variables that are represented as sets of weighted particles. A particle is one possible state of target. The filter is initialized by the state of target at the start time. The steps are shown as follows. The first stage is *prediction*. Particles are propagated from frame $t-1$ to frame t using the following motion model

$$S(i, t) = A \times S(i, t - 1) + W(i, t - 1) \quad (5)$$

where $S(i, t)$ denotes the state of i th particle in frame t . The parameter A denotes state-transition matrix and W denotes Gaussian distribution of random variables. The second stage is to *update*. We compute a weight for each propagated particle using our appearance model and normalize all of weights to sum to one. The third stage is to *resample* to avoid particle degeneracy phenomenon. The particles with low weight are deleted and high weight particles are replicated. As a result, a new set of equal weight particles can be generated [18].

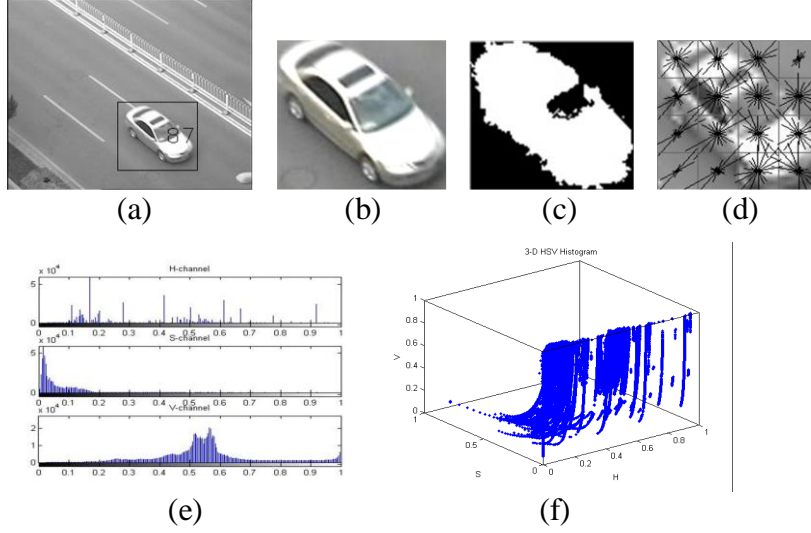


Fig. 4. (a) Frame 1224 of Video1. (b) An extracted target from frame 1224. (c) Target binary foreground image. (d) Target HOG image. (e) - (f) Target HSV histogram.

Object Matching. Object matching aims to determine which object O_n^j in current frame is the most relevant with the object O_{n-1}^j in previous frame. Therefore, we need to measure the similarity between the observation and template. The color or HOG feature is used to describe in many other measurements. In this paper, we build a new combination evaluation for tracking vehicles. Firstly, we resize the bounding box of the target to the same size of template. Fig. 4(c) shows the target binary foreground image of target in Fig. 4(b). The HOG image and HSV histogram of the target are shown in Fig. 4. The Bhattacharyya similarity $f(h^t|x_{j,i}^t, h_j)$ between model histogram and observed histogram is calculated by the Eq. 6.

$$f(h^t|x_{j,i}^t, h_j) \propto e^{-d(h^j, h^t)} \quad (6)$$

Where

$$d(h^j, h^t) = \sum_{i=0}^{N-1} \sqrt{h_i^t h_i^j} \quad (7)$$

Using the HSV and HOG descriptors, we can get HSV weight $d_{\text{HSV},n}^j$ and HOG weight $d_{\text{HOG},n}^j$ of j th particle in frame n . h_i^j denotes i th bin in the observed histogram of the j th particle and h_i^t denotes the i th bin in the histogram of template of the j th particle. In this paper, we set N is 288 for HSV descriptor and 324 for HOG descriptor. The RDHOG weight $D_{\text{RDHOG},n}^j$ is calculated by Eq. 8

$$D_{\text{RDHOG},n}^j = \sum_{i=0}^{D-1} |b_i^j - b_i^t| \quad (8)$$

where b_i^j and b_i^t denotes the i th bin of RDHOG descriptor of the observed object and template. The number of D is 288. $D_{\text{RDHOG},n}^j$ denotes the difference between the central block and its neighbor blocks. The weight of j th particle in frame n is given in Eq. 9.

$$w_n^j = (d_{\text{HSV},n}^j \times d_{\text{HOG},n}^j) / D_{\text{RDHOG},n}^j \quad (9)$$

The weight of particle reflects the degree of similarity between candidate bounding box with the template. It will increase when $d_{\text{HSV},n}^j$ or $d_{\text{HOG},n}^j$ increases and decrease when $D_{\text{RDHOG},n}^j$ increases.

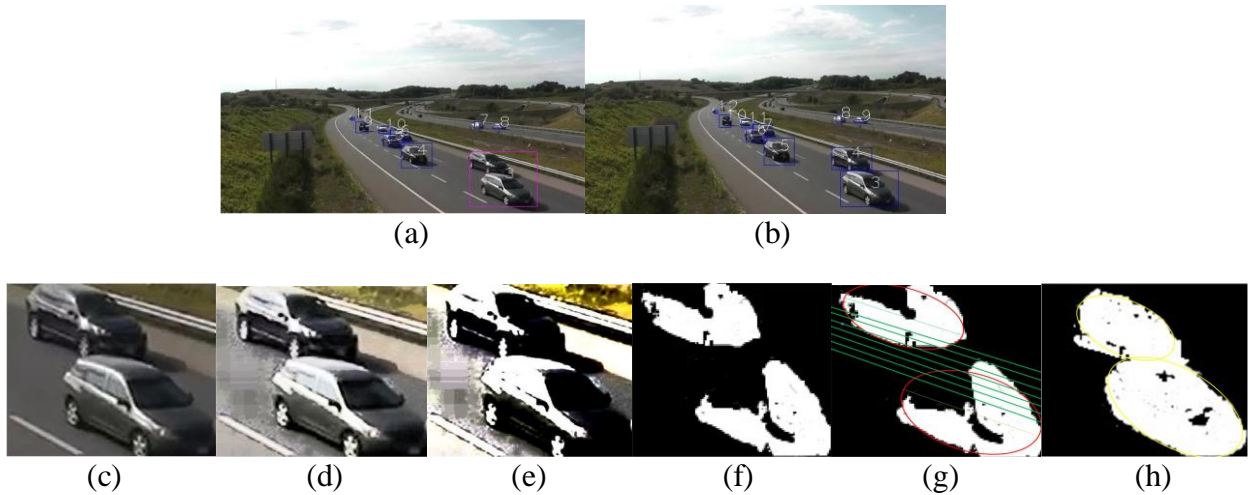


Fig. 5. The procedure of occlusion segmentation. (a) Vehicle tracking without handling occlusion. (b) Vehicle tracking with handling occlusion (c) Occlusion detection and candidate region extraction. (d) Histogram equalization. (e) Gray-scale transformation. (f) One-dimensional maximum segmentation. (g) Ellipse fitting. (h) Line scanning and foreground segmentation.

A Novel Occlusion Handling Method. Due to the phenomenon of interaction and overlapping between vehicles, the detected candidate bounding box may contain multiple vehicles. The appearance of a vehicle will exhibit different in two adjacent frames when occlusion exits, and thus make it difficult to associate the same vehicle extracted from the frames. The two main problems should be solved in occlusion segmentation that are occlusion detection and multi-vehicle segmentation.

A lot of study work has been done for occlusion detection. Our occlusion detection is based on geometrical shape of area because the adaptable algorithm needs less prior knowledge and processing speed is relatively fast. When occlusion happens, the duty ratio $R_d = R_V/R_B$ and aspect ratio $R_a = R_H/R_w$ become abnormal where R_V denotes the area of the vehicle region and R_B denotes the envelop rectangle. R_H and R_w are the height and width of the rectangle. The detailed detection is designed as follows. The smaller value of R_d , the more likely the region is abnormal and R_a is on the contrary. If $R_d < Y_d$ or $R_a > Y_{HW}$, we regard this region as abnormal. Under the basis of vast analysis and careful studies, we set Y_d as 0.75 and Y_{HW} as 1.55. In non-occlusion cases, we use particle filter to predict the positions of vehicles in current frame. In occlusion cases, we must segment all vehicles in one candidate box, and track them in particle filter framework.

To solve the problem of the segmentation of vehicles with occlusion, we first adopt the one-dimensional maximum entropy method to split the image into a binary segment image, and some result are shown in Fig. 5(f). In this paper, the candidate region is a rectangle surrounding the vehicles. For better splitting, we apply histogram equalization and image gray-scale transformation to enhance the contrast of candidate bounding box. Then the least square ellipse is utilized to fit the connected region of the binary image. The long axis of ellipses is in the direction of lane. Lines that are parallel to the major axis of these ellipses divide the foreground boundary into several sub-boundaries depending on the number of ellipses. For the case of multi-vehicle occlusion, three or more ellipses can be detected with the one-dimensional maximum entropy method. The lines scan the region between the two each close centers of ellipses to find the least sum of distance from the centroids of sub-boundaries (Fig. 5(g) and Fig. 5(h)). In contrast to many algorithms which mainly focus on occlusion of two vehicles, three or more vehicles can be segment effectively. Fig. 5(b) shows the result of segmentation and tracking for object#3 in Fig. 5(a).

Experiments and Results

Experiment setup. In this section, we evaluate our proposed system based on several video sequences. These video sequences contain various traffic environments respectively, and we could detail describe each video in the following. Vehicle video1 denotes low or medium traffic flow. The highway II video denotes the medium traffic flow with the occurrence of shadow. Vehicle video2 denotes a high traffic flow with serious occlusion.

The experiments are performed in PC platform with Intel Core i5 with a 3.3-GHz central processing unit and 3GB of RAM. Our approach is developed in C++ using OpenCV Library [19]. OpenCV is a library of programming functions mainly focuses on image processing and computer vision. The obtained experiment results are compared with the other two similar methods proposed by Lien *et al.* [20] and Scharcanski *et al.* [21]. The method proposed by Lien *et al.* [20] mainly focuses on texture-based object segmentation and tracking. The method proposed by Scharcanski *et al.* [21] is designed to detect occlusion and track vehicle using a Rayleigh distribution based on particle filter.

Performance analysis. For the evaluation of tracking performance, the measurement includes three factors: correct tracking which denotes the number of correctly tracking frames, missing tracking which denotes the number of missing tracking frames and false tracking which denotes the number of false tracking frames.

Table 1 shows the comparison between our proposed method with the other two methods. Aim to evaluating the effectiveness of our framework, we evaluate all of them on three videos. Although the method of Lien could effectively detect vehicle without background model, the texture-based target tracking method may fail when the texture feature of vehicles is similar. Compared with the method of Lien, our appearance model could hold essential characteristic of vehicles more accurately. From the table, we could see that the system of Lien has lower correct fraction in video2 due to it can't deal with occlusion. The method proposed by Scharcanski could handle occlusion and the correct fraction is slightly higher. However, this algorithm has limitations in dealing with occlusion because it can't be applied to other tracking system. And it performs worse in poor lighting or shadow environments because the method can't detect shadow well. Our system has high correct fraction in traffic scene with shadow which shows the robustness.

As expected, our approach outperforms the other two systems. Experimental results show that our proposed framework was running at about 20 frames per second and the accuracy of tracking can approach 95%. However, when the vehicles in the video are so far away that appear too small, their local appearance characteristics and spatial relationships can't be captured completely. That is the reason of high missing tracking ratio and false tracking ratio. One solution is to set up the region of interest in the central region of the video. In this way, we can discard the region near the border and improve our system performance.

Table 1. Performance comparison of vehicle tracking

Video	Tracking Algorithm	Correct Tracking	Missing Tracking	False Tracking	Correct Fraction(%)	Error Fraction(%)
Video1	Our system	3146	55	49	96.8%	3.2%
	Lien <i>et al.</i> [20]	2935	234	81	90.4%	9.6%
	Scharcanski <i>et al.</i> [21]	3020	138	92	92.9%	7.1%
Highway II	Our system	450	8	4	97.4%	2.6%
	Lien <i>et al.</i> [20]	436	7	19	94.3%	5.7%
	Scharcanski <i>et al.</i> [21]	424	25	13	91.7%	9.3%
Video 2	Our system	280	8	12	93.3%	6.7%
	Lien <i>et al.</i> [20]	260	11	29	86.6%	13.4%

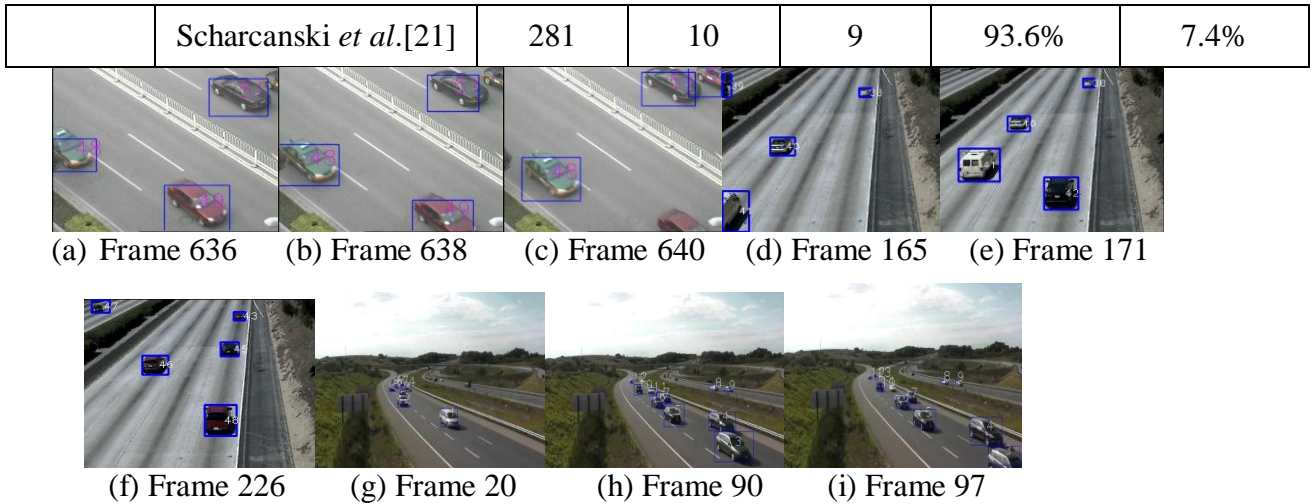


Fig. 6. The tracking results of test sequences. The results of vehicle tracking for video1 are shown in (a), (b) and (c) that present our proposed method is able to track multiple vehicles in real traffic scene. The results for highway II video are shown in (d), (e) and (f) that present the results of real-time shadow scene. The results for video2 are shown in (g), (h) and (i) that present the tracking results before, during and after vehicles are occluded. It shows that our framework can handle occlusion problem and track vehicles accurately.

Summary

In this paper, based on particle filter, we present a robust processing system for vehicle detection and tracking while satisfying the real-time constraints, which can handle shadow and occlusion effectively. The improved feature fusion is efficient for shadow detection. And the enhanced appearance model works well in different environments. Our proposed occlusion segmentation can handle occlusion accurately. Experiments under different illumination intensity and traffic flow demonstrate the robustness and efficiency of our framework.

References

- [1] Kamijo, S., Matsushita, Y., Ikeuchi, K., and Sakauchi, M., "Traffic monitoring and accident detection at intersections," *IEEE Transaction on Intelligent Transportation Systems*, Vol.1, No.2, pp. 703-708(2000).
- [2] Zhou, J., Gao, D., and Zhang, D., "Moving vehicle detection for automatic traffic monitoring," *IEEE Transaction On Vehicular Technology*, Vol.56, No.1, pp.51-59(2007).
- [3] O.Barnich, M. Van Droogenbroeck. ViBe: A powerful random technique to estimate the background in video sequences [C].*Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 945-948 (2009).
- [4] C. Stauffer, W. Eric, L. Grimson. Learning patterns of activity using real-time tracking [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8): 747-757(2000).
- [5] K. Kim, T. Chalidabhongse, D. Harwood, and L. Davis, "Real-time foreground-background segmentation using codebook model," *Special Issue on Video Object Processing, Real-Time Imag.*, Vol. 11, pp.172-185 (2005).
- [6] Yongqing Li, Wanzhong Chen, Rui Jiang. The integration adjacent frame difference of improved ViBe for foreground object detection [C]. *Proceedings of 7th International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 1-4 (2011).

- [7] Cucchiara R, Grana C, Piccardi M, et al. Improving shadow suppression in moving object detection with HSV color information [C]. Proceedings of IEEE Intelligent Transportation Systems, pp. 334-339 (2001).
- [8] JAVED O, SHAH M. Tracking and object classification for automated surveillance [C]. Proceedings of IEEE European Conference on Computer Vision. [S.I.], pp. 343-357 (2002).
- [9] Daisuke Sugimura, Kris Makoto Kitani, Takahiro Okabe, Yoichi Sato, and Akihiro Sugimoto. Using individuality to track individuals: clustering individual trajectories in crowds using local appearance and frequency trait .In ICCV. 1467-1474(2009).
- [10] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In CCPR(2005).
- [11] J. Scharcanski, A. B. de Oliveria, P.G. Cavalcanti, and Y. YARI, "A particle-filtering approach for vehicular tracking adaptive to occlusions," IEEE Transaction on Vehicular Technology, Vol.60, No.2, pp.381-389 (2011).
- [12] LINDERBERG T. Feature detection with automatic scale selection [J]. International Journal of Computer Vision, 30(2):79-116 (1998).
- [13] KAMIJO S, SAKAUGHI M. Segmentation of vehicles and pedestrians in traffic scene by spatio-temporal Markov random field model [C]. Proceedings of the 21st International Conference on Data Engineering Workshops, pp.1-8 (2005).
- [14] ZHANG Xin, JI Xiu-hua. An improved Harris corner detection algorithm for noised images [J]. Advanced Materials Research, 433: 6151-6156(2012).
- [15] D. Comaniciu, V. Rameshand, P. Meer, Real-time tracking of non-rigid objects using mean-shift, in: Computer Vision and Pattern Recognition, pp. 142-149 (2000).
- [16] WU B, KAO C, JEN C, et al. A relative discriminative histogram of oriented gradients-based particle filter approach to vehicle occlusion handling and tracking [J]. IEEE Transactions on Industrial Electronics, 61 (8): 4228-4237 (2014).
- [17] M.S.Arulampalam, S.Maskell, N.Gordon, and T.Clapp.A tutorial on particle filters for online nonlinear/non-gaussian Bayesian tracking. IEEE Transactions on Signal Processing, 50(2): 174-188 (2002).
- [18] Rui, Y, Chen, Y.: Better proposal distributions: Object tracking using unscented particle filter. In: IEEE Conference on Computer Vision and Pattern Recognition, Vol 2, pp.786-793(2001).
- [19] OpenCV on SourceForge: <http://sourceforge.net/projects/opencvlibrary/>.
- [20] C.C. Lien, Y.T. Tsai, M.H. Tsai, and L.G Jang, "Vehicle counting without background modeling," 17th International Conference on Advances in Multimedia Modeling, vol. Part 1, Springer-Verlag, Berlin, Heidelberg, pp.446-456 (2011).
- [21] Scharcanski, J.; de Oliveira, A.B.; Cavalcanti.P.G.; Yari, Y. (2011). "A Particle-Filtering Approach for Vehicular Tracking Adaptive to Occlusions", IEEE Transactions on Vehicular Technology, Vol. 60, No.2, pp.381-389 (2011).