2014 IEEE 17th International Conference on
Intelligent Transportation Systems (ITSC)
October 8-11, 2014. Qingdao, China

Visual Vehicle Tracking Based on Conditional Random Fields

Yuqiang Liu, Kunfeng Wang, and Fei-Yue Wang, *Fellow, IEEE*

***Abstract*—This paper proposes an approach to moving vehicle tracking in surveillance videos based on conditional random fields (CRF). The key idea is to integrate a variety of relevant knowledge about vehicle tracking into a uniform probabilistic framework by using the CRF model. In this work, the CRF model integrates spatial and temporal contextual information of vehicle motion, and the appearance information of the vehicle. An approximate inference algorithm, loopy belief propagation, is used to recursively estimate the vehicle region from the history of observed images. Moreover, the background model is updated adaptively to cope with non-stationary background processes. Experimental results show that the proposed approach is able to accurately track moving vehicles in monocular image sequences. Besides, region-level tracking realizes precise localization of vehicles.**

***Index Terms*—Vehicle tracking, conditional random fields, region-level tracking**

I. Introduction

Nowadays, visual traffic detection which collects various parameters of road traffic flow is attracting more and more attention in the fields of computer vision and intelligent transportation systems. Meanwhile, visual object tracking is an important and challenging task in these fields [1-3]. As an active research topic, tracking takes many forms, including automatic or manual initialization, single or multiple objects, still or moving camera, etc., each of which has been associated with an abundant literature.

Vehicle tracking is typically used to measure the vehicle trajectories in video sequences for analysis of urban traffic [4]. However, from the view of tracking results, there are mainly two types of tracking problems: trajectory-level tracking and region-level tracking. Trajectory-level tracking only gets the object trajectory (usually the center of the object in each frame), while region-level tracking aims at not only locating the object continuously, but also segmenting the object as accurately as possible. The first type of tracking receives most of the attention from researchers. In traffic surveillance, trajectory-level tracking usually use a bounding box to represent the object, which is not enough to precisely obtain the traffic parameters. In contrast, the second type of tracking enables us to locate the object and measure object attributes accurately. However, it is more difficult for us to perform region-level tracking.

The rest of this paper is organized as follows. In Section II, a review of existing methods is presented. In Section III, the CRF-based tracking framework is discussed. Section IV describes the implementation of our tracking system. In Section V, details of experimental results and discussion are proposed. Finally, Section VI draws a conclusion for this work.

II. Related Works

Numerous efforts have been devoted to addressing the tracking problem [5]. Existing tracking methods can be categorized according to how the features are used for tracking. A type of traditional region-level tracking approaches treats foreground segmentation and object tracking as two problems. They firstly obtain the foreground, then extract features from the foreground and finally track the objects based on the features [3, 6]. The main disadvantages of this processing method are that the errors of foreground segmentation always propagate forward, leading to object tracking errors. In fact, foreground segmentation and object tracking are closely related to each other. Firstly, foreground segmentation results directly determine the accuracy of feature extraction and further impact the performance of object tracking in region level. Secondly, the tracking results can provide a top-down cue for foreground segmentation. Therefore, simultaneous foreground segmentation and object tracking is able to make full use of the correlation between them and realize a bidirectional flow of information, which can greatly improve the performance of object tracking.

In [6], the foreground segmentation is firstly extracted and then the objects with occlusion are segmented based on the features of color and location. However, this method does not take the information of relationship between adjacent local blocks. Based on the observation of the candidates, Bugeau etc. [3] presents a kind of tracking and segmentation method by minimizing the energy function. Before tracking, the observation is obtained by an external object detection method. The model combines the low-level pixel-wise measures (color, motion), high-level observations obtained by an external object detection method and motion predictions. A probabilistic framework is introduced in [7] for joint segmentation and tracking based on Bayesian inference, which improves the robustness of tracking a variety of objects. However, the spatial relationship is not considered in the segmentation and there will be holes and splits in the results.

On the other hand, probabilistic graphical models are kind of mathematical tools for tracking for solving the inference problem of motion estimation. The spatial-temporal Markov random field (S-T MRF) is used in [8-10] for vehicle tracking in urban traffic scenes. The input image, which consists of 640 × 480 pixels, is divided into 80 × 60 blocks. Every block corresponds to a node in an S-T MRF. The S-T MRF is used to model the tracking problem and generate labels for the blocks.

This work is supported partly by the National Natural Science Foundation of China under Grant 61304200, MIIT Project of the Internet of Things Development Fund: R&D of Real-time Image Recognition Techniques and Application Systems for Social Security, and the Early Career Development Award of SKLMCCS under Grant 2F13Q02.

The authors (Kunfeng Wang is the corresponding author) are with Qingdao Academy of Intelligent Industries, Qingdao 266109, China, and also with The State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (phone: 86-10-82544791; e-mail: yuqiang.liu@ia.ac.cn, kunfeng.wang@ia.ac.cn).

Blocks in consecutive frames that are adjacent in spatial and temporal domain are considered neighbors for the model. The S-T MRF model estimates a current object-map of the current frame based on the current image, the previous image, and the previous object map. In [11], the S-T MRF is used in H.264/AVC-compressed video sequences for tracking moving objects. The model is established according to the motion vectors and block coding modes from the compressed bit stream. However, this method only uses the motion vector feature to track the object, which can not obtain an accurate region of the object in a complex scene. In [12], a bidirectional association graph similar to MRF is used to track regions and handle the splitting and merging of region over a sequence of images [4]. Every region in a frame corresponds to a node in the graph. The graph has two partitions represent the regions from the previous and current frame respectively. The edges between vertices in the two partitions indicate that the previous regions are associated with the current regions. The weights of edges are the area of overlap between regions in the two partitions. The region tracking problem is considered as the problem of solving the maximal weight as of the graph.

However, the above methods using S-T MRF model mostly consider the motion information, which in fact can be categorized as a kind of motion detection methods. As a result, these methods can only give a bounding box of the objects without precise region characterizing the object. Moreover, in the literature, many graph cuts-based methods have been proposed for segmentation issues, but very few works use this methodology for tracking[2]. As a result, our methods take advantage of these methods to achieve accurate region-level tracking. To track individual objects effectively, we have developed a tracking algorithm based on the CRF model. In this paper, we focus on multi-object tracking in the region level by using a static camera, by combining the advantages of tracking and segmentation. Based on these ideas, we achieve the tracking in both the temporal domain and the spatial domain. Our contribution is that we use CRF model to incorporate motion and appearance cues in a single unified framework to jointly track and segmentation objects, a key step towards traffic scene understanding. In addition, since we do not specify the unique features of the object, the method presented in this work is a general framework and can track different kinds of objects.

III. Segmentation and Tracking Based on CRF

The definition of a conditional random field (CRF) is given by Lafferty et al. in [13]. As a probabilistic framework for labeling and segmenting structured data, CRF is being widely used in computer vision. Specifically, we can define a CRF on observations X and random variables Y over the video scene. Let $G = (V, E)$ be a graph such that $\mathbf{y} = (\mathbf{y}_v)_{v \in V}$, so that $\mathbf{y}$ is indexed by the vertices of G. Then $(\mathbf{x}, \mathbf{y})$ is a conditional random field when the random variables $\mathbf{y}_v$, conditioned on $\mathbf{x}$, obey the Markov property with respect to the graph:

$$P(\mathbf{y}_{\mathrm{v}} | \mathbf{x}, \mathbf{y}_{\mathrm{w}}, \mathrm{w} \neq \mathrm{v}) = P(\mathbf{y}_{\mathrm{v}} | \mathbf{x}, \mathbf{y}_{\mathrm{w}}, \mathrm{w} \sim \mathrm{v}) \quad (1)$$

where w∼v means that w and v are neighbors in G. The conditional distribution for a CRF takes the form [14]:

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{a=1}^{A} \psi_a(\mathbf{y}_a, \mathbf{x}_a) \quad (2)$$

where ψ_a is a potential function and the x-dependent partition function $Z(\mathbf{x}) = \sum_{\mathbf{y}} \prod_{a=1}^{A} \psi_a(\mathbf{y}_a, \mathbf{x}_a)$ is a normalization factor which ensures that the distribution $P(\mathbf{y}|\mathbf{x})$ given by (1) is correctly normalized. In the next sections, we will describe our tracking model based on the CRF model.

A. *Definition of the Tracking Model*

Generally speaking, the moving object is characterized by its spatial and temporal characteristics. For example, the appearance usually does not change during the tracking, the object often has relatively consistent motion features and the region occupied by the object has similarity of motion. In addition, the object will not disperse across different parts of the frame. In the spatial domain, if the vehicles in two adjacent frames are the same one, the spatial distance is close. Based on all these characteristics, our CRF model incorporates the motion and appearance for tracking the vehicles.

Tracking can be treated as a problem of finding the most likely assignment of every pixel, that is inferring maximum a posteriori (MAP) solution of the CRF model. The CRF model uses the graph structure in Fig. 1. More specifically, given an image sequence $\{\mathbf{x}_t\}$. As shown in Fig.1, the node $\mathbf{y}_i$ represents the object labels, and the node $\mathbf{x}_i$ represents the observation. For the object labels $\mathbf{y}$, the label for a pixel with the index $i = 1, \dots, D$ within the tth image is denoted by $y_t(i)$. We want to infer the tracking label $y_t(i) \in \{0, 1, \dots, N\}$ given the observed image sequence $\{\mathbf{x}_t\}$, which is, in fact, the inference of the MAP of CRF.

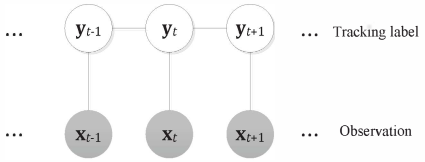


Figure 1. Graphical representation of CRF, where the $\mathbf{y}_i$ represents the object labels, and the $\mathbf{x}_i$ represents the observation. Note that $\mathbf{x}_i$ is not generated by the model.

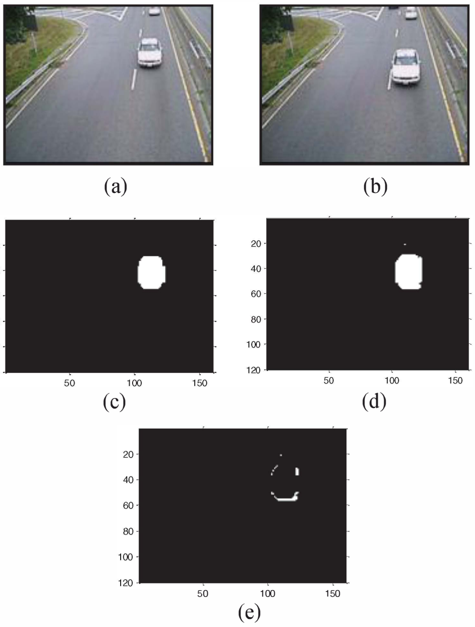
In this work, we define the conditional probability of the foreground segmentation and tracking process given an image sequence $\{\mathbf{x}_t\}$ as

$$\log P(\mathbf{y}_t | \mathbf{y}_{t-1}, \mathbf{x}, \theta) \propto \sum_i \big(\psi_i(y_t(i), \mathbf{y}_{t-1}, \mathbf{x}_{t-1:t}; \theta_\psi) + \omega_i(y_t(i), \mathbf{x}_{t-1:t}; \theta_\omega) + \rho_i(y_t(i), \mathbf{x}; \theta_\rho) + \textstyle\sum_{(i,j) \in N_i} \tau_i(y_t(i), y_t(j), \mathbf{x}_{t-1:t}; \theta_\tau)\big) \quad (3)$$

where ψ_i, ω_i, ρ_i and τ_i are potential functions, $\boldsymbol{\theta} = \{\theta_\psi, \theta_\omega, \theta_\rho, \theta_\tau\}$ are the model parameters, and N_i denotes the 8-pixel spatial neighborhood. Note that the model consists of three unary potentials and one pairwise potential. In following sections, we will give a description of the energy functions and their parameters used for tracking objects.

B. *Temporal Association Relationship*

In order to establish the association relationship in temporal domain, we employ dense optical flow as the simple motion estimation of the tracked object based on the approach proposed in [15]. Note that the energy used in foreground segmentation also implicitly uses motion information, but only considers the general motion characteristic. Compared to the method prosed in [16, 17], we think the optical flow can give a better representation of the corresponding relationship between pixels in consecutive frames. An intuitive idea is that we can use the optical flow to backwards-project the pixel $\mathrm{x}_t(i)$ in current frame. If the corresponding pixel $\mathrm{x}_{t-1}(j)$ in the previous frame is labeled as object, then the pixel $\mathrm{x}_t(i)$ should be labeled as object. The intuitive example is shown in Fig.2.



(a) (b)

(c) (d)

(e)

Figure 2. Projection result using the optical flow: (a) previous frame, (b) current frame, (c) ground truth image, (d) backwards-projection image, (e) difference between the backwards-projection and the ground truth.

From the results in Fig. 2(e), we can see that the simple backwards-projection can get a good performance except the boundary. However, it is usually sensitive to the noise if we only consider the corresponding pixel in the previous frame, which will result in lots of holes and split in the tracking. Inspired by the work proposed in [11] for single object tracking, the temporal association relationship is measured by the overlap, which can be determined by the backwards-projected candidate labeling for the current frame and the labeling of the previous frame.

As shown in Fig. 3, after getting the optical flow, we backwards-project the pixel in the tth frame back to the pixel in the $(t-1)$th frame. Specifically, consider a pixel P in the current frame and we can project it backwards into the pixel P' in the previous frame along its optical flow $\mathbf{v}_t(i) = \left(v_x(i), v_y(i)\right)$, that is $(x + \Delta x, y + \Delta y)$, where Δx and Δy is the fraction of P'. The degree of overlap R is computed as the fraction of pixels within the backwards-projected pixel in the previous frame which carries the object label according to the labeling $\mathbf{y}_{t-1}$.

This temporal association relationship potential can be defined as:

$$\psi_i\left(y_t(i), \mathbf{y}_{t-1}, \mathbf{x}_{t-1:t}; \theta_\psi\right) = \theta_\psi R(i) \delta\left(\lambda\left(x_t(i)\right) - i\right) \quad (4)$$

where $R(\cdot)$denotes the overlap, $\delta(\cdot)$ is the Kronecker delta function. We now introduce a labeling function $\lambda: \mathrm{x}_t \longmapsto [0; \mathrm{N}]$, which associates each pixel of the current image with an object or the background.

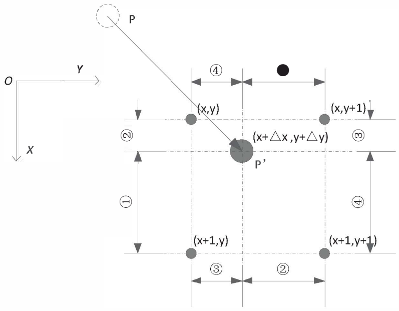
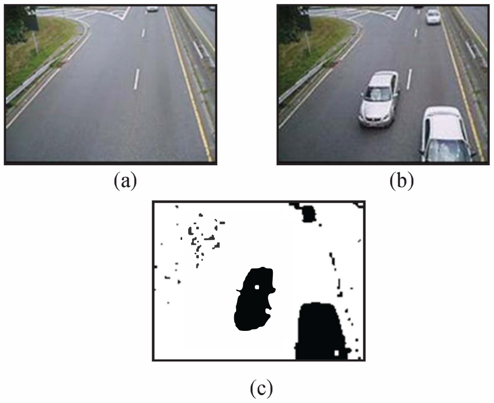


Figure 3. Degree of overlap R. The point P is the candidate point in current frame, while the projected point P' is the point with the coordinate $x_{t-1}(i) + \mathbf{v}_t(i)$, that is$(x + \Delta x, y + \Delta y)$. Δx and Δy is the fraction of P'.

C. Foreground Segmentation

Temporal association relationship is a powerful cue for object tracking and can provide relatively accurate results in many cases. However, it highly depends on the optical flow which may be inaccurate especially near object boundaries, which is more evident in Fig. 5. The dense optical flow calculation does not always work well for we often choose a region to search and for the low rate video this assumption is not appropriate. Moreover, the motion vector can be affected by the illumination and complex background, which will result in a lot of holes and that the boundary is usually not precise. Besides the errors from the optical flow, the errors in previous labeling will transfer to the current frame for we use the overlap based on labeling of the previous frame. The accumulative error would lead to a bad tracking result.



(a) (b)

(c)

Figure 4. Foreground segmentation result: (a) background image, (b) current frame, (c) foreground mask where black pixels denote foreground.

On the other hand, we can employ temporal or dynamic information to handle the evolution of the scene. Moreover, in order to handle non-stationary background processes, we can update the background model in an adaptive way based on the recent history of observed images [16]. A background image is constructed using GMM, to give an mask of the tracked object. The importance of using foreground segmentation cues to address the inaccuracy of optical flow will become more evident in the experimental results (Section 5.A). As a result, we use background model to filter the noise and make the tracking system more robust. In this paper, we construct the background model by using GMM[18], as shown in Fig. 4. In addition, morphology is used to remove noise and join disparate element.

This foreground segmentation potential can be defined as:

$$\omega_i(y_t(i), \mathbf{x}_{t-1:t}; \theta_\omega) = \theta_\omega \delta(g(x_t(i)) - i) \quad (5)$$

where θ_ω is the weight of the potential. A labeling function is introduced here $g: \mathbf{x}_t \longmapsto [0,1]$. The labeling function associates each pixel of the current image with the object or the background via GMM.

D. *Motion Coherence*

In natural video, one characteristic of rigid object is the relative coherence of the motion characteristics of pixels belonging to the object, as shown in (c) and (d) of Fig. 5. In this work, we will use the Gaussian mixture model (GMM) to model the distribution of the optical flows. The clusters of optical flows are performed in an unsupervised manner using k–means and the number of components of the GMM is typically set to 5. As shown in (e) of Fig.5, we can calculate the probability of pixels to belong to each object exploiting the motion distribution of objects. In Fig.5(f), we set the posterior probability of the background to a constant, which makes the posterior probability of the objects more clear. We can see that different objects often have different motion model, which can be used as a cue for classification of different objects.

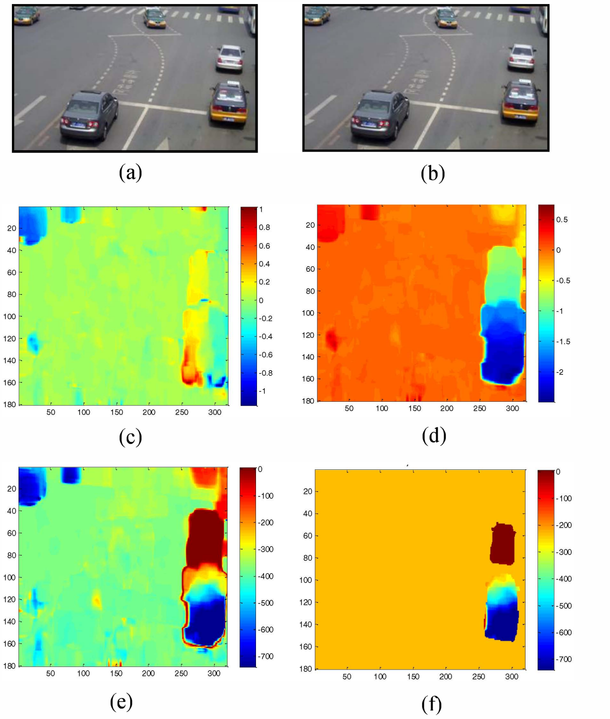

(a)
(b)
(c)
(d)
(e)
(f)

Figure 5. Dense optical flow of the vehicle. (a) Previous frame. (b) Current frame. (c) and (d) show the dense optical flow in x and y direction. (e) and (f) are the posterior probability using GMM modeling and (f) set the posterior probability of the background to a constant.

The potential function is defined as follows:

$$\rho_i(y_t(i), x; \theta_\rho) = \theta_\rho \log P(y_t(i)|x) \quad (6)$$

where θ_ρ is the weight and $P(y_t(i)|x)$ denotes the probability of each pixel to belong to the object based on the motion model.

E. *Local Smoothness*

Generally speaking, we assume that the object is spatially compact. To be more specific, for the candidate pixel, the more neighbors there are belonging to the object , the more likely the pixel belongs to the object. We take this neighboring relationship into account by connecting each pixel variable to its nearest neighbors in the image plane to encode a local smoothness constraint. Specifically, let $y_t(j) \in N_i$ be the neighborhood pixel, we define a potential function term with the type as follows:

$$\begin{aligned} &\tau_i(y_t(i), y_t(j), \mathbf{x}_{t-1:t}; \theta_\tau) \\ &= \theta_\tau \frac{1}{\|\mathbf{x}_t(i) - \mathbf{x}_t(j)\|^2} \left(1 - \delta(y_t(i) - y_t(j))\right) \end{aligned} \quad (7)$$

where θ_τ is the weight and we tend to give a larger weight to the closer neighbors. Thus, two neighboring pixels are more likely to have the same label.

IV. Implementation

For the accuracy of optical flow will have an effect on the performance of our tracking method, we use a filter to eliminate the outliers by using statistical approach. In this paper, we use 3σ rule to detect the outliers and then use a median value to replace them.

As for the tuning of parameters, we think that the contribution of every energy function is different, which is influenced by its reliability. Therefore, when we determine the parameters we measure its reliability to tune the parameters. However, learning the CRF parameters proves difficult. In our work, we use a more pragmatic solution based on piecewise training [19] to determine the parameters. Piecewise training divides the CRF model into pieces corresponding to the different terms in (3). These pieces in our model are then trained independently, as if each of them was the only term in the conditional model. Finally, we recombine these pieces with weights. When we use piecewise training in our work, the parameters $\theta_\psi, \theta_\omega, \theta_\rho$ and θ_τ can be learned by maximizing the conditional likelihood in each of the four models. In each case, only the factors in the model which contain the relevant parameter vector are retained. In addition, this training method minimizes an upper bound on the log partition function.

Given the structure of the CRF model and the learned parameters, we can infer the MAP solution of the model, i.e., the labeling that maximizes the conditional probability of (3). That is to find the most probable labeling $\mathbf{y}^*$ as shown in (8),

$$\mathbf{y}^* = \operatorname{argmax}_{\mathbf{y}} \log P(\mathbf{y}_t | \mathbf{y}_{t-1}, \mathbf{x}, \boldsymbol{\theta}) \quad (8)$$

The optimal labeling is found by applying the loopy belief propagation. Although belief propagation is exact only when the structure has no loop, in practice it has been proved to be a successful approximate inference method for general graphical models [20].

V. Experiment and Discussion

We implement the CRF tracking in Matlab and C++ without any code optimization and also use some functions in OpenCV. The experiments are carried out on an Intel 3.1GHz PC platform with 4 GB memory. The reference parameters used in this paper are presented as follows. The weight parameters in the foreground segmentation layer are $\theta_\psi = 0.55, \theta_\omega = 1, \theta_\rho = 0.5$, and $\theta_\tau = 0.5$. We conduct our experiment on the sample video in MATLAB and the video of ours captured at Zhongguancun Road. The video shots include vehicles under different scales and poses. Because our method is a general tracking method, the unique appearance and decoration have little effect on the final results. In the first frame, we use the hand-made initialization.

A. Tracking vehicle with large scale change

In this experiment, the video is from the sample video in MATLAB, from the results we can see that the algorithm can deal with the tracking problem on the highway. The reason is that the scene is simple and there is not much interference and the background is stable. From the results, we can see that the vehicle can be tracking in region level while the size of the car changes dramatically during the tracking. However, there are some errors near the boundary, which are caused partly by the difference between the car and the background and partly by the accurate manual annotation.

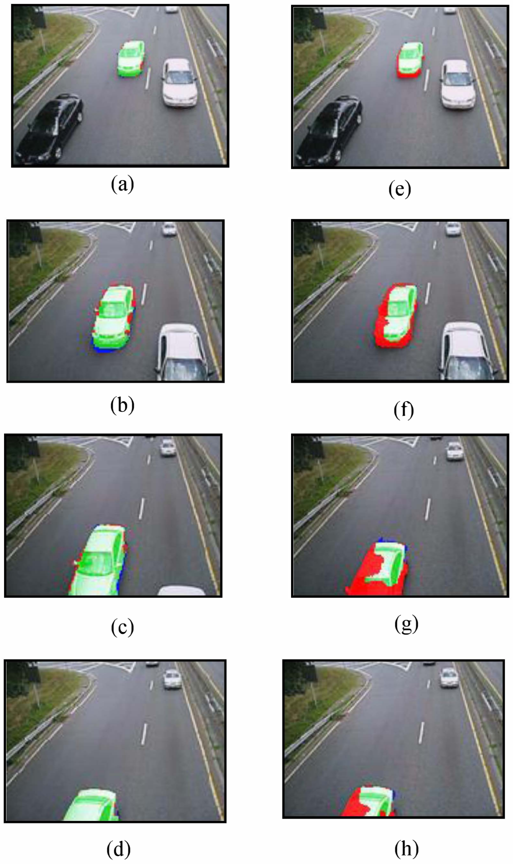


Figure 6. Result of vehicle tracking with large scale change. (a)-(d) are results of tracking for Frame #73, #77, #80, and #81 respectively. (e)-(h) are results of tracking for Frame #73, #77, #80, and #81 respectively after setting the wight θ_ω to zero.True positives (TPs) are shown as green, false positives (FPs) as blue, and false negatives (FNs) as red.

On the other hand, we propose an analysis of the influence on the results of the foreground segmentation term of the energy defined in (5). If the parameter θ_ω is set to zero, it means that no foreground segmentation is applied to the tracking. The final tracking result of the vehicle then only depends on the optical flow and the probability of each pixel to belong to the object. That is the reason why the vehicle is not well segmented in (e)-(h) of Fig. 6.

B. Tracking vehicle on urban road

In this experiment, we evaluate our method in the urban scene at Zhongguancun Road. In this traffic scene, the background is more complex and has more noise, which makes it more difficult to achieve precise region-level tracking of the vehicle.

From the results shown in Fig. 7, we can see that our algorithm can track the object in region level. However, if the difference between the background and the boundary of the vehicle is small, the segmentation will contain errors. From the results, we can see that there are errors in the boundary, resulting from inaccurate optical flow and low image resolution.

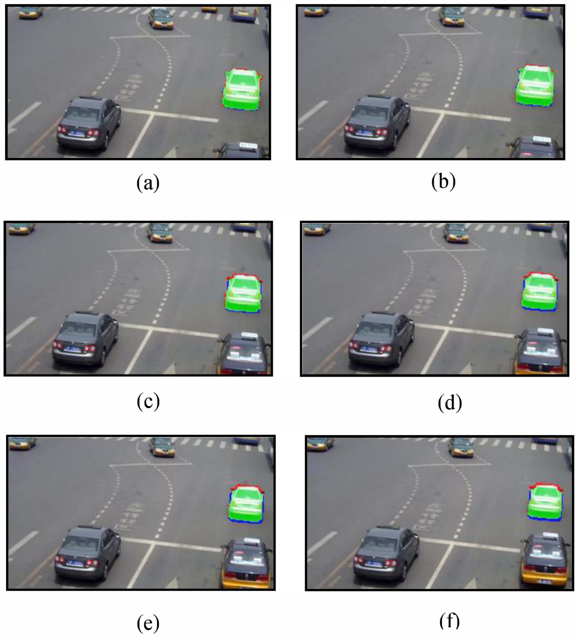


Figure 7. Result of vehicle tracking on urban road. (a)-(f) are Frame #2060, #2062, #2068, #2070, #2072, and #2074 respectively. True positives (TPs) are shown as green, false positives (FPs) as blue, and false negatives (FNs) as red.

C. Quantitative Analysis

Table 1. shows a quantitative analysis in terms of Accuracy, Recall, and False alarm, which are defined as in (9), (10) and (11).

$$Accuracy = \frac{TP+TN}{TP+FN+FP+TN} \tag{9}$$

$$Recall = \frac{TP}{TP+FN} \tag{10}$$

$$False\ alarm = \frac{FP}{FP+TN} \tag{11}$$

As shown in Table. 1, we can get a good performance for the vehicle tracking through our method. However, as mentioned before, if the background and the boundary of the vehicle have the similar appearance, the segmentation will contain errors. Moreover, for the tracking in urban road, the recall is not as good as in the sample video in MATLAB. The main reason of this is that the environment in urban road is more complex.

TABLE I
COMPARISON OF TRACKING METHODS IN TERMS OF THE AVERAGE ACCURACY, RECALL, AND FALSE ALARM

	Accuracy	*Recall*	*False Alarm*
Section A	99.6%	94.7%	0.2%
Section B	99.5%	91.8%	0.3%

VI. CONCLUSION

In this paper, we use a conditional random field (CRF) model to build the conditional distribution over vehicle tracking given an image sequence $\{\mathbf{x}_t\}$. The application of conditional random fields allows us to incorporate various motion and appearance information into a single unified framework. For this model, we employ dense optical flow to establish the relationship between the objects in consecutive frames, GMM background model to segment foreground and background, and local smoothness to impose constraints between the neighboring pixels. By jointly employing these cues, we succeed in achieving accurate vehicle tracking.

In the future, we will extend the tracking model by endowing it with the abilities of automatic initialization, shadow suppression, and occlusion handling. These extensions will make the approach more effective and useful in practice.

REFERENCES

[1] *PASCAL. http://pascallin.ecs.soton.ac.uk/challenges/VOC/.*

[2] N. Papadakis and A. Bugeau, "Tracking with occlusions via graph cuts," *IEEE transactions on pattern analysis and machine intelligence,* vol. 33, pp. 144-157, 2011.

[3] B. Aurélie and P. Patrick, "Track and cut: simultaneous tracking and segmentation of multiple objects with graph cuts," *EURASIP Journal on Image and Video Processing,* vol. 2008, 2008.

[4] N. Buch, S. A. Velastin, and J. Orwell, "A review of computer vision techniques for the analysis of urban traffic," *IEEE Transactions on Intelligent Transportation Systems,* vol. 12, pp. 920-939, 2011.

[5] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Comput. Surv.,* vol. 38, p. 13, 2006.

[6] L. Zhu, J. Zhou, and J. Song, "Tracking multiple objects through occlusion with online sampling and position estimation," *Pattern Recognition,* vol. 41, pp. 2447-2460, 2008.

[7] C. Aeschliman, J. Park, and A. C. Kak, "A probabilistic framework for joint segmentation and tracking," in *2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010, pp. 1371-1378.

[8] S. Kamijo, K. Ikeuchi, and M. Sakauchi, "Vehicle tracking in low-angle and front-view images based on spatio-temporal markov random field model," in *8th World Congress on ITS*, 2001.

[9] S. Kamijo, Y. Matsushita, K. Ikeuchi, and M. Sakauchi, "Traffic monitoring and accident detection at intersections," *IEEE Transactions on Intelligent Transportation Systems,* vol. 1, pp. 108-118, 2000.

[10] S. Kamijo, K. Ikeuchi, and M. Sakauchi, "Event recognitions from traffic images based on spatio-temporal markov random field model," in *8th World Congress on ITS, Sydney*, 2001.

[11] S. Khatoonabadi and I. Bajic, "Video Object Tracking in the Compressed Domain Using Spatio-Temporal Markov Random Fields," *IEEE Transactions on Image Processing,* vol. 22, pp. 300-313 January 2013.

[12] S. Gupte, O. Masoud, R. F. K. Martin, and N. P. Papanikolopoulos, "Detection and classification of vehicles," *IEEE Transactions on Intelligent Transportation Systems,* vol. 3, pp. 37-47, 2002.

[13] J. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *the 18th International Conference on Machine Learning 2001 (ICML 2001)*, 2001, pp. 282-289.

[14] C. Sutton, "An Introduction to Conditional Random Fields," *Foundations and Trends® in Machine Learning,* vol. 4, pp. 267-373, 2012.

[15] G. Farnebäck, "Two-Frame Motion Estimation Based on Polynomial Expansion," in *Scandinavian Conference on Image Analysis*, 2003, pp. 363-370.

[16] Y. Wang, K.-F. Loe, and J.-K. Wu, "A dynamic conditional random field model for foreground and shadow segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 28, pp. 279-289, 2006.

[17] Y. Wang and Q. Ji, "A dynamic conditional random field model for object segmentation in image sequences," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005)*, 2005, pp. 264-270.

[18] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1999, p. 252.

[19] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context," *International Journal of Computer Vision,* vol. 81, pp. 2-23, 2009.

[20] C. Sutton, A. McCallum, and K. Rohanimanesh, "Dynamic conditional random fields: Factorized probabilistic models for labeling and segmenting sequence data," *The Journal of Machine Learning Research,* vol. 8, pp. 693-723, 2007.