Rear-View Vehicle Detection and Tracking by Combining Multiple Parts for Complex Urban Surveillance

Bin Tian, Ye Li, Bo Li, and Ding Wen, *Senior Member, IEEE*

Abstract—Traffic surveillance is an important topic in intelligent transportation systems. Robust vehicle detection and tracking is one challenging problem for complex urban traffic surveillance. This paper proposes a rear-view vehicle detection and tracking method based on multiple vehicle salient parts using a stationary camera. We show that spatial modeling of these vehicle parts is crucial for overall performance. First, the vehicle is treated as an object composed of multiple salient parts, including the license plate and rear lamps. These parts are localized using their distinctive color, texture, and region feature. Furthermore, the detected parts are treated as graph nodes to construct a probabilistic graph using a Markov random field model. After that, the marginal posterior of each part is inferred using loopy belief propagation to get final vehicle detection. Finally, the vehicles' trajectories are estimated using a Kalman filter, and a tracking-based detection technique is realized. Experiments in practical urban scenarios are carried out under various weather conditions. It can be shown that our method adapts to partial occlusion and various lighting conditions. Experiments also show that our method can achieve real-time performance.

Index Terms—Kalman filter (KF), Markov random field (MRF), part-based object detection, tracking, vehicle detection.

I. Introduction

WITH rapid development of urbanization, traffic congestion, incident, and violation pose great challenges for traffic management systems. Vision, as an information collection access of real-world environments, has attracted much attention in intelligent transportation systems (ITSs). Computer vision techniques are mainly used to collect traffic parameters and analyze traffic behaviors for traffic surveillance. In [1], an overview of the background, concepts, basic methods, major issues, and current applications of parallel transportation management systems (PTMSs) was proposed. Video processing results can provide various information derived from the actual traffic world for PTMSs.

Reliable and robust vehicle detection is a fundamental component for traffic surveillance. There are still some issues for vehicle detection in ITSs. Various vehicle appearances and poses make it difficult to train a unified detection model. Complex urban environments, bad weather, illumination changes, and poor/strong lighting conditions will degrade the detection performance dramatically. In particular, for traffic congestion, vehicles are occluded by each other so that separate vehicles will easily merge into a single vehicle. The parameter learning for the vehicle detection is also a critical issue. A detection method with complex parameters is usually not practical. The advances of machine learning techniques can be used to learn the parameters. In [2], a weakly supervised approach for object detection was proposed. This method does not need manual collection and labeling of training samples. A boosting algorithm was extended to train samples with probabilistic labels.

In this paper, we aim to develop a vehicle detection and tracking system particularly for urban traffic surveillance under various environments in China. The system utilizes a rear-view stationary camera to capture the image sequence. In practical traffic scenarios, occlusion between vehicles often occurs; therefore, it is unreasonable to treat the vehicle as a whole. Much research has detected the object by detecting its parts first and measuring their spatial relationships; this is called a part-based model. In our system, we treat the vehicle as an object composed of multiple parts. Unlike other vehicle detection methods, we choose salient vehicle parts, including the license plate and rear lamps, which usually exist on each vehicle. Then, we combine these parts into a vehicle by using a Markov random field (MRF) to model their spatial relationships. We further track the detected vehicles by employing a Kalman filter (KF) to obtain vehicle trajectories. A detection-by-tracking strategy is realized to improve vehicle detection performance. The main contributions of this paper lie in the following. First, novel methods are proposed to localize vehicle parts. We localize the Chinese license plate and the rear lamp using their distinctive color, texture, and region features. Second, we propose a part-based vehicle detection model using the MRF. Detected vehicle parts are combined into a vehicle using an MRF model in which parts are treated as graph nodes. Our method can adapt to partial occlusion and various lighting conditions. The details of the system will be discussed in later sections.

B. Tian, Y. Li, and B. Li are with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: bin.tian@ia.ac.cn; ye.li@ia.ac.cn; bo.li@ia.ac.cn).

D. Wen is with the Center for Military Computational Experiments and Parallel Systems Technology, National University of Defense Technology, Changsha 410073, China (e-mail: wending2010@gmail.com).

The remainder of this paper is organized as follows. In Section II, we provide an overview of the latest research related to vehicle detection and tracking. Section III presents how to localize vehicle parts and combine them into a vehicle. Section IV describes the vehicle tracking process and a detection-by-tracking strategy. Section V provides extensive and detailed experiments to analyze the system performance. Section VI concludes this paper and discusses future work.

II. RELATED WORKS

In good and stable lighting conditions, moving object detection methods are widely used for vehicle detection in ITSs. These methods can be classified into background modeling, frame differencing, and optical flow [3], [4]. They can handle illumination change and apply to multimode and slight background change. However, there are some drawbacks in that they are unable to detect a stationary vehicle and the detected moving object is not necessarily a vehicle. Therefore, much research utilizes the visual features of the vehicle to detect it in a still image [5], [6]. Features such as Gabor, color, edge, and corner are usually used to represent the vehicle. Then, they are fed into a deterministic classifier and a generative model to identify vehicles. In addition, researchers usually employ a two-step method, including hypothesis generation and hypothesis verification [7], to locate the vehicle. This method works well during the sunny daytime but may fail during poor lighting conditions such as nighttime.

Many recent studies on part-based models have been developed to recognize vehicles. According to a human cognitive study [8], a vehicle is considered to be composed of a window, a roof, wheels, and other parts. These parts are usually learned and detected by using their appearance, edge, and shape features [9], [10]. After part detection, the spatial relationship, motion cues, and multiple models are usually used to detect vehicles. Winn et al. [11] decomposed an object into several local regions to detect them. The relationships between them were used to improve detection performance by a layout conditional random field model. Hoiem et al. [12] further expanded this method by using 3-D models to mark the learning samples. In addition, vehicle parts can be selected and learned automatically in a deformable part-based model [13]. Niknejad et al. [14] employed this model composed of five components, including the front, back, side, front truncated, and back truncated. Each component contained a root filter and six part filters, which were learned using a latent support vector machine and a histogram of oriented gradients features. To use the part-based model in parameter transfer learning, Xu and Sun [15] proposed a basic learning framework named part-based transfer learning (PBTL). All the complex tasks are regarded as a collection of constituent parts, and each task can be divided into several parts, respectively. Transfer learning between two complex tasks can be accomplished by subtransfer learning tasks between their parts. To avoid negative transfer and improve the effectiveness of transfer learning, Sun et al. [16] extended PBTL into an effective learning framework named multisource PBTL. By this, it is possible to focus on the parts that contribute more to the target task. The inference method is critical for a probabilistic graph model. In [17], Sun gave a comprehensive introduction of loopy belief propagation (LBP). The pros and cons of LBP were listed, and the improved methods were analyzed in detail with a sufficient survey of the literatures.

Vehicle detection at nighttime is a dramatic challenge for traffic surveillance. Generally, vehicle headlights and taillights are used to represent the vehicle [18]. Robert [18] detected bright blobs as candidate headlights. To filter out false detection, they assumed that two headlights were aligned horizontally. Inspired by [18], Wang et al. [19] proposed a two-layer nighttime detection method. In the first layer, the headlight detection process was the same as in [18]. In the second layer, Haar features based on the AdaBoost cascade method were employed to recognize vehicle frontal views. Zhang et al. [20] modeled the reflection intensity map, the reflection suppressed map, and image intensity into an MRF model to distinguish light pixels from reflection pixels. For taillight detection, its color information is mostly utilized. O'Malley et al. [21] proposed a system to detect and track vehicle taillight pairs. Taillight candidates were extracted with a hue–saturation–value (HSV) color threshold and paired by their symmetry features. To detect nighttime brake lights, Chen et al. [22] modeled taillights using the Nakagami distribution.

Vehicle tracking is used to obtain trajectories of moving vehicles, enabling higher level tasks such as traffic incident detection and behavior understanding. A detailed survey about object tracking is shown in [23]. Vehicle tracking can be classified into four categories: model-based tracking, region-based tracking, deformable template-based tracking, and feature-based tracking. Various filtering algorithms are used in tracking tasks, such as the Bayesian filter, the KF, and the particle filter [14], [18]. Object tracking based on the mean-shift algorithm is an appearance-based tracking method and performs well in tracking moving objects even in dense traffic [24]. However, this method needs manual initialization of a target model. In [25], kinematic variables including position, speed, and size of the vehicle were estimated with a projective KF to initialize the mean-shift tracker.

Different from the aforementioned methods, we treat the vehicle as an object composed of multiple salient parts, including the license plate and rear lamps. With the advance of the MRF model, the visual features and structure of the parts are modeled into probability distributions. Hence, the method proposed in this paper can adapt to partial occlusions and various lighting conditions.

III. PART-BASED VEHICLE DETECTION

A. Outline of Vehicle Detection

For practical urban traffic surveillance, reliable and robust vehicle detection is a fundamental component. Occlusions between vehicles occurs frequently so that it is unreasonable to treat the vehicle as a whole. In this paper, we treat the vehicle as an object composed of multiple salient parts, including the license plate and rear lamps, which usually exist on each vehicle. These parts are localized using their distinctive color, texture,
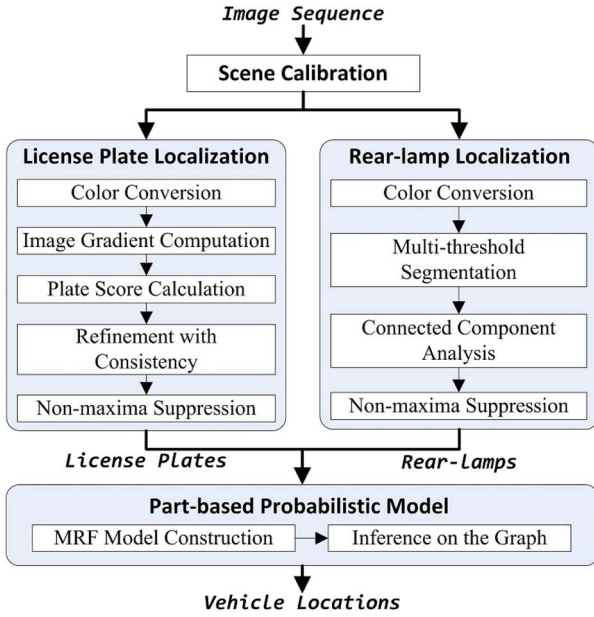
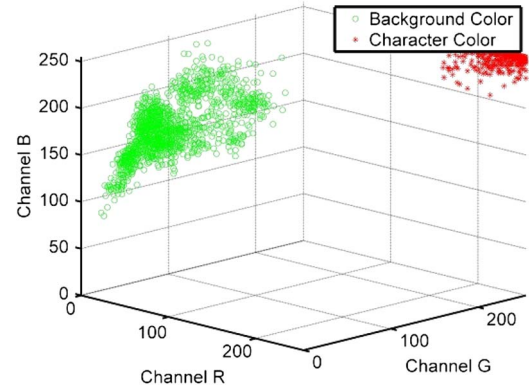Fig. 1.   Flowchart depicting the vehicle detection pipeline.

Fig. 2.   RGB scatter plot of pixels from a database of license plate images for a blue background/white character plate.
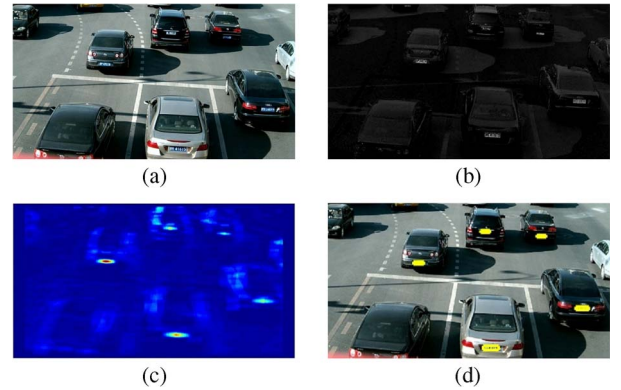

Fig. 3.   Example of the license plate location process. (a) Input image. (b) Converted license plate color image. (c) Score image obtained by gradient statistics. (d) License plate localization results.

and region features. Then, an MRF model is constructed to model the spatial relationships among these parts to infer and localize the vehicle. Even if some parts are occluded, vehicles can be correctly detected. Meanwhile, our method can cope with several weather and lighting conditions. An illustration of the vehicle detection pipeline is shown in Fig. 1.

B. License Plate Localization

License plate localization is a critical technology in ITS applications for traffic management and analysis. In this section, we localize the Chinese license plates with a coarse-to-fine strategy. It has its unique texture features that are distinguished from the other vehicle parts. The Chinese license plates include a blue background/white character plate, a yellow background/black character plate, a white background/black-and-red character plate, and a black background and white character plate.

Our license plate localization process is shown in Fig. 1. Here, we take a white–blue pair as an example. This type of plate is the most common in China. To get the width of the plate for each image coordinate, we utilized the calibration toolbox in the OpenCV library for traffic scene calibration.

1) Deriving Plate Color Converting Model: We mainly utilized the color difference between plate characters and plate background. The vast majority of plate characters are white or black, whose values are not certain in HSV space. Consequently, we chose the red–green–blue (RGB) color space. A database of 50 license plate images were created for the blue–white pair. The color distributions of plate pixels are shown in Fig. 2. As observed, the plate background and plate characters have some unique characteristics. For the plate background pixel, its value of the blue channel is far greater than those of the other two channels. Moreover, the values of the green and red channels are both relatively small. For the plate character pixel, the values of all three channels are relatively

large. The characteristics were analyzed to find the appropriate color conversion for the plate.

According to these observations, we converted the image into a specific color space as in

$$C_{x,y} = B_{x,y} - \min\{R_{x,y}, G_{x,y}\} \tag{1}$$

where $C_{x,y}$ is the converted color of pixel (x, y), and $R_{x,y}$, $G_{x,y}$, $B_{x,y}$ are the red, green, and blue channel values. The converted color image is shown in Fig. 3(b), which is also called the plate color image. The conversion can enhance the difference between the plate background and characters. Furthermore, it suppresses the values of the nonplate pixels. Comparing with a grayscale image, the gradient of the plate region is significantly enhanced, whereas the gradient of the other regions is suppressed. For the yellow–black pair, a similar process was taken to obtain the corresponding plate color image. For the white–black pair and black–white pair, we did not perform color conversion and use the grayscale image immediately.

2) Plate Hypothesis Score Calculation: After color conversion, the image gradient was calculated for the plate color image. We calculated the image gradient by computing the difference between the maximum and minimum of the neighboring pixels, as calculated in

$$\text{Grad}_{x,y} = \max\{\text{Nb}_{x,y}\} - \min\{\text{Nb}_{x,y}\} \tag{2}$$

where $\mathrm{Grad}_{x,y}$ is the gradient of the pixel $(x\ y)$ and $\mathrm{Nb}_{x,y}$ is the neighboring pixels. Then, we scanned the gradient image with a sliding window to compute the average gradient. Using the previous calibration result, we could get the plate size and scan with a plate-size window, as in

$$S_{x,y} = \sum_{x'=x-\mathrm{pw}/2}^{x+\mathrm{pw}/2} \sum_{y'=y-\mathrm{ph}/2}^{y+\mathrm{ph}/2} \mathrm{Grad}_{x',y'}/(\mathrm{pw}*\mathrm{ph}) \quad (3)$$

where $S_{x,y}$ is the plate score at pixel $(x,\ y)$. pw and ph are the plate width and height, respectively. Considering the processing speed, we scanned the gradient sparsely. As a result, we obtained a score image, as shown in Fig. 3(c).

Finally, we used a nonmaxima suppression (NMS) [26] strategy to seek local maximums in the score image. If a local maximum is greater than a predefined threshold α, it is recognized as a candidate plate region.

3) Cascade Plate Refining: Except for the plate regions, there may be some plate-like regions on the vehicle. They have strong gradient and meet the color feature, which may lead to false detection. As we observed, the gradient of a plate region is not only strong but also evenly distributed. Consequently, we further used the consistency of textures on the gradient image to improve algorithm efficiency. We chose texture consistency for texture measurement as computed in

$$U = \sum_{i=0}^{L-1} p^2(z_i) \quad (4)$$

where U is the consistency of an image region. L indicates the possible gray levels. $p(z)$ is the histogram of the gray level in an image region. The consistency of the real plate region is relatively consistent (0.7 in this paper). We scanned the candidate plate regions to obtain refined localization results, as shown in Fig. 3(d).

C. Vehicle Rear-Lamp Localization

A vehicle rear lamp is an obvious feature of the vehicle. According to the Chinese national standard, the color of the vehicle rear lamp is red and falls within a specified range. We adopted the multithreshold segmentation and connected component analysis to extract as many candidate rear lamps as possible. In this section, we focus on candidate rear-lamp localization without pairing them. The localization process is shown in Fig. 1.

1) Deriving the Color Converting Model: Similar to license plate localization, we operated rear-lamp localization under the RGB color space. As observed, a rear-lamp pixel satisfies some unique properties. The value of the red channel is large. The values of the green and blue channels are small. The differences between the values of the green and blue channels are small. The characteristics were analyzed to find the appropriate color conversion for the rear lamp. The conversion can suppress the values of the nonrear-lamp pixels.
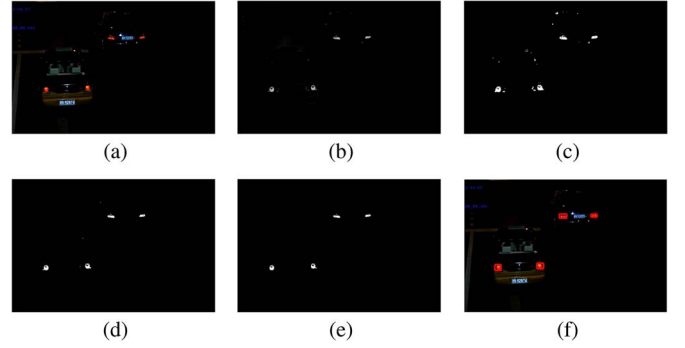

Fig. 4. Example of the rear-lamp localization process. (a) Input image. (b) Converted rear-lamp color image. (c)–(e) Binary images obtained by three thresholds ($\sigma_1 = 20$, $\sigma_2 = 40$, $\sigma_3 = 80$). (f) Rear-lamp localization results.

The RGB image was converted into a color grayscale image, which is called the rear-lamp color image, as calculated in

$$C_{x,y} = R_{x,y} - \max\{G_{x,y}, B_{x,y}\} - 2 * \|G_{x,y} - B_{x,y}\| \quad (5)$$

where $C_{x,y}$ is the converted color of pixel (x, y), whereas $R_{x,y}$, $G_{x,y}$, $B_{x,y}$ are the red, green, and blue channel values. The converted rear-lamp color image is shown in Fig. 4(b).

2) Rear-Lamp Candidate Generation by Multithreshold Segmentation: Motivated by the maximally stable extremal region detector [27], we adopted a multithreshold segmentation method to segment the rear-lamp color image. To adapt to relatively dark and bright lamps, we used three thresholds to execute binarization. Three binary images were obtained, as shown in Fig. 4(c)–(e).

Finally, a connected component analysis was used to extract candidate rear lamps on the binary images as follows.

Step 1) Get the connected regions by multithreshold segmentation: Q_1, Q_2, Q_3 ($Q_i \subset Q_{i+1}$).

Step 2) Select the rear-lamp regions with appropriate areas in the threshold ascending order, i.e., $\mathrm{Area}(Q_i) \in [\gamma_1, \gamma_2]$.

Step 3) Clear the remaining regions Q_k ($k < i$) and analyze the next connected region to Step 1.

An NMS strategy was used to remove overlaid rear lamps. The final localization results are shown in Fig. 4(f).

D. Vehicle Detection Using MRF

Earlier, we localized the vehicle parts, including the license plate and rear lamps. Then, it became a challenge how to efficiently combine these parts into a vehicle. In this section, to make better use of the relationships among vehicle parts, we adopted an MRF model to localize the vehicle.

1) Probability Model Representation: At first, we constructed an MRF graph model and defined its model parameters. Vehicle parts were treated as the graph nodes, and the relationships among them were the graph edges. First, we selected one detected license plate as a graph node in the current frame. Then, neighboring vehicle rear lamps were added into the graph if they were close enough to the license plate. In Section IV-C, we will mention that tracking results are used to improve the detection results by adding predictions of vehicle location into the graph.

Fig. 5.   MRF graph construction and inference process. (a) Treat vehicle parts with green rectangles as inputs. (b) Construct MRF graphs. Red points and lines denote nodes and links in MRF, respectively. (c) Infer these on the graphs. (d) Vehicle detection results. Detected vehicles are labeled with green rectangles.

Graph $G = \{V, E\}$ was constructed with one license plate and multiple rear-lamp candidates, where $V = \{v_1, v_2, \ldots, v_n\}$ are nodes denoting vehicle parts and $E = \{e_1, e_2, \ldots, e_m\}$ are edges denoting relationships among neighboring nodes. G is a complete graph and each pair of nodes is connected by an edge. Each node in G corresponds to a random variable F_i. Order $f = \{f_1, f_2, \ldots, f_n\}$ is a configuration of F. f_i belongs to $Q = \{1, 2, 3, 0\}$. In our MRF graph, there are four types of nodes:

- $q = 1$, license plate node;
- $q = 2$, left rear-lamp node;
- $q = 3$, right rear-lamp node;
- $q = 0$, false detection node.

Our MRF model is a pairwise model with the distribution in

$$P(F) = \frac{1}{Z} \prod_{c \in C} \psi_c(f_c) = \frac{1}{Z} \underbrace{\prod_{i \in V} \varphi(f_i)}_{\text{node potential}} \underbrace{\prod_{(i,j) \in E} \phi(f_i, f_j)}_{\text{edge potential}} \quad (6)$$

$$Z = \sum_F \prod_{i \in V} \varphi(f_i) \prod_{(i,j) \in E} \phi(f_i, f_j) \quad (7)$$

where Z is a partition function. $\varphi(f_i)$ is the node potential, representing detection confidence of each node. $\phi(f_i, f_j)$ is the edge potential, representing the relationship between each pair of nodes. The model is shown in Fig. 5(b). Then, we will determine the node potentials and edge potentials.

a) Node potentials: The node potential $\varphi(f_i = p)$ indicates the probability of node i to be part p without considering other nodes. It depends on the scores of part detectors S_i. To integrate the discriminative detectors into the MRF model, it is necessary to give a probabilistic meaning to the outputs. We introduce the sigmoid function to normalize the scores of the vehicle parts. The final node potential is

$$\varphi(f_i = p) = \begin{cases} \frac{1}{1 + \exp\{A_p * S_i + B_p\}}, & \text{Node } i \text{ is detected as } p \\ 1 - \frac{1}{1 + \exp\{A_p * S_i + B_p\}}, & p = 0 \\ \lambda, & \text{otherwise.} \end{cases} \quad (8)$$

The probability of the node to be false detection is obtained by computing the complementary probability of the detection. If node i is not detected as q, its probability is set to be a small constant λ (10^{-2} in this paper).

The parameters A_p and B_p for each part were learned by the sigmoid fitting method introduced in [28]. For the rear lamps, there are no discriminative features, except their color. Therefore, its detection score was set to be a constant 0.8 by careful
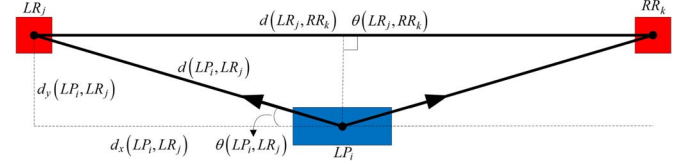

Fig. 6.   Spatial relationship among vehicle parts.

selection and cross validation. The corresponding sigmoid parameters were set to be $A_p = -1$ and $B_p = 0$.

b) Edge potentials: The edge potential $\phi(f_i = p, f_j = q)$ evaluates the compatibility between the neighboring nodes. It represents the relationship between the node pairs connected by an edge. A vehicle model is defined to model the edge potentials, as shown in Fig. 6. In the model, there are two main spatial relationships between vehicle parts, i.e., the relationship between the plate and a rear lamp, and the other one between both rear lamps as follows:

1) The distance between vehicle parts, i.e., $d(\text{LP}_i, \text{LR}_j)$ and $d(\text{LR}_j, \text{RR}_k)$;
2) The angle between vehicle parts, i.e., $\tan^{-1}(d_x(\text{LP}_i, \text{LR}_j)/ d_y(\text{LP}_i, \text{LR}_j))$ and $\tan^{-1}(d_x(\text{LR}_j, \text{RR}_k)/d_y(\text{LR}_j, \text{RR}_k))$.

Since the poses of vehicles on the road vary greatly, we chose the Gaussian to model the spatial relationship between parts. In addition, since a variety of types of vehicles exists, the relationships between nodes might be multimodal. Based on the two considerations, the Gaussian mixture model (GMM) was selected to model the part relationships. The edge potential is

$$\phi(f_i = p, f_j = q)$$
$$= \begin{cases} \sum_{k=1}^{M} \alpha_k p_k(x|\theta_k), & (p \neq 0 \text{ and } q \neq 0) \\ & \text{and } p \neq q \\ \frac{1}{2}\left(1 - \sum_{k=1}^{M} \alpha_k p_k(x|\theta_k)\right), & (p = 0 \text{ or } q = 0) \\ & \text{and } p \neq q \\ \lambda, & p = q. \end{cases} \quad (9)$$

The relationships between the part nodes are represented by the GMM. Because the left and right rear lamps are horizontally symmetrical on both sides of the license plate, we learned the relationship model of the plate and the left rear lamp and obtained that of the plate and the right rear lamp by symmetry. The relationship between the false detection node and the part node ($p = 0$ or $q = 0$) is difficult to be represented by a probability distribution; hence, they are modeled using the complementary
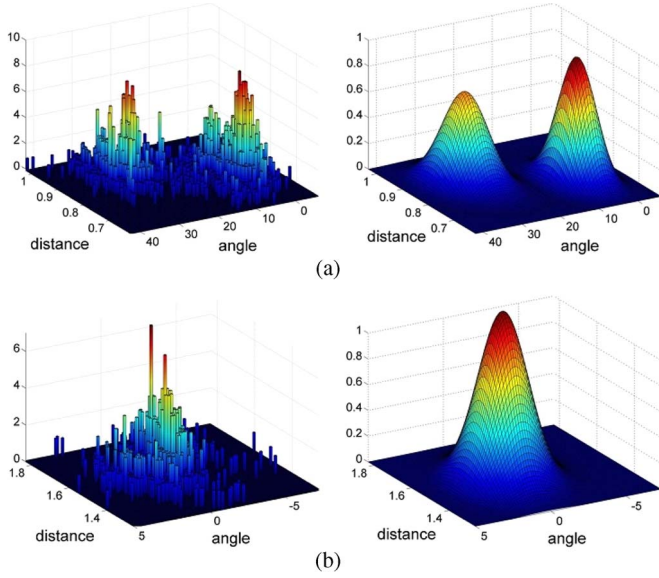
Fig. 7. Part relationships modeled by GMMs. The left column provides the statistical histograms of the part relationships including distance and angle. The right column shows the learned models of the part relationships. (a) Relationship between the plate and the rear lamp. (b) Relationship between the both rear-lamps.

probabilities of part relationships. In this paper, the probability of the pairwise part relationship is $\sum_{k=1}^{M} \alpha_k p_k(x|\theta_k)$, and the complementary probability of the node pair with a false detection node is supposed to be $(1/2)(1 - \sum_{k=1}^{M} \alpha_k p_k(x|\theta_k))$. For the case where two nodes share the same labels ($p = q$), the probability takes a small constant λ (10^{-2} in this paper).

For the parameter learning process, we used the expectation–maximization (EM) algorithm to estimate the parameters of the GMM using the maximum-likelihood estimation. The weights α_k, mean, and covariance θ_k for each Gaussian component were the parameters for the GMM. The vehicle parts were carefully labeled artificially. As shown in Fig. 7, the relationship between the plate and the rear lamp is modeled to be a GMM with two components. The relationship between both rear lamps is modeled to be one Gaussian component.

2) Inference: Node potentials $\varphi(f_i)$ of variables and edge potentials $\phi(f_i, f_j)$ between the graph nodes determine the probabilistic assumptions of the MRF model. Here, we perform probabilistic inference to estimate each $f_i \in F$, to satisfy the maximum *a posteriori* (MAP) in

$$Q^* = \arg \max \ P(F)$$
$$= \arg \max \prod_{i \in V} \varphi(f_i) \prod_{(i,j) \in E} \phi(f_i, f_j). \quad (10)$$

Pearl *et al.* [29] proposed belief propagation inference, which is a probabilistic inference method. For the acyclic graph, it can provide the exact solution. For a loopy graph, it can provide an approximate solution, which is called LBP. Here, we initialize all messages $m_{ij}(f_j)$ to 1. After certain updating steps, all of the messages tend to converge. Then, the marginal probability of each variable, which is called beliefs, are read out. Beliefs are normalized so that they approximate the marginal probability.
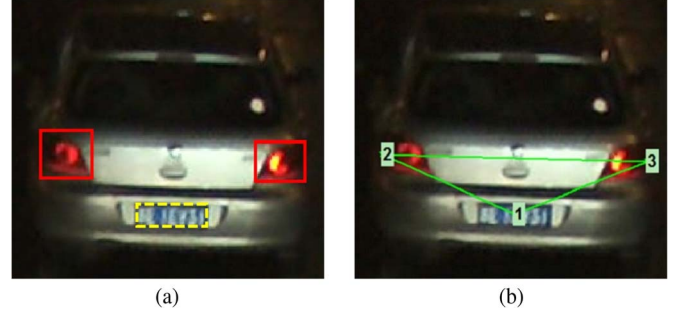

Fig. 8. Example of a detection result improved by tracking. (a) License plate with the dashed rectangle is lost localized. The rear lamps are localized correctly with red rectangles. (b) Vehicle parts are correctly inferred using the prediction of the tracker as a graph node.

After vehicle inference, optimal labels are assigned for all of the MRF nodes, as shown in Fig. 5(c). The vehicle detection results are shown in Fig. 5(d). During the vehicle detection process, if more than two parts are correctly inferred, we recognize it as a vehicle. In case some parts are not detected or are wrongly detected, our method can still localize them, as in Fig. 9.

IV. VEHICLE TRACKING

Vehicle detection has been described in the earlier sections. In this section, detected vehicles are tracked to obtain vehicle trajectories. A simple detection-by-tracking strategy was used to improve vehicle detection accuracy.

A. Outline of the Tracking Process

Vehicle tracking is used to predict vehicle positions in subsequent frames, match vehicles between adjacent frames, and ultimately obtain the trajectory of the vehicle. In our system, we used the KF to track vehicles. In statistics, the KF [30] is a minimum variance estimation of linear movement. Each KF corresponds to a tracked object, which is called a tracker. It estimates the true values of observations with noise.

B. KF

In our system, we treated the center of the vehicle license plate and the speed of the vehicle as the state vector, which is shown as

$$x = [p_x, \ p_y, \ s_x, \ s_y]^t$$

where p_x and p_y are x and y coordinates of the vehicle, respectively. s_x and s_y are the speeds of the x-axis and y-axis directions. KF is a recursive estimation method and can be split into two steps, i.e., prediction and update.

1) Prediction: In the prediction step, we predicted state vector x and state error covariance matrix P at the current time k, as in

$$\hat{x}_k = F x_{k-1}$$
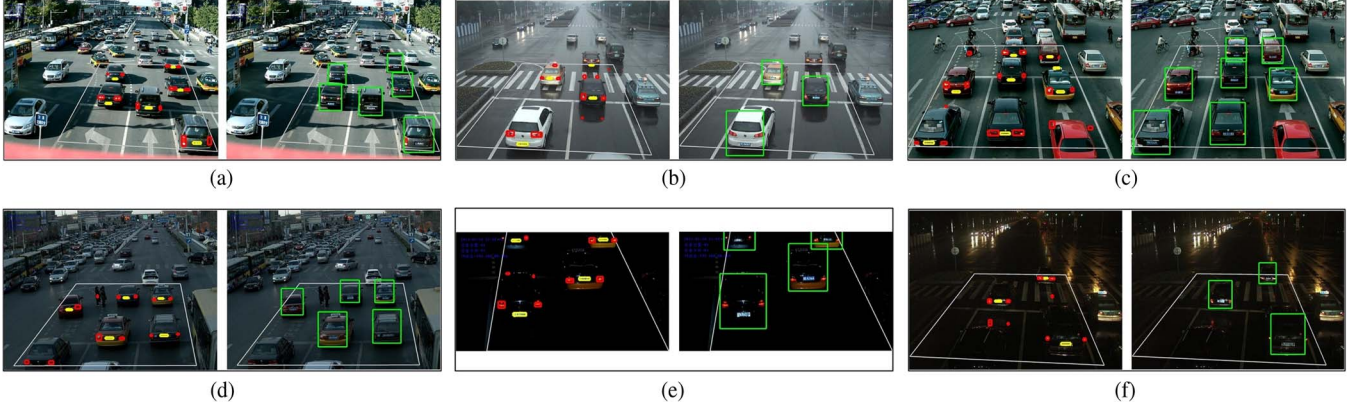$$\hat{P}_k = F P_{k-1} F^t + Q \quad (11)$$

Fig. 9. Vehicle detection results for different scenarios. For each scenario, the left side is the vehicle part detection result, and the right side is the vehicle detection result. The license plates are marked with yellow rectangles. The rear lamps are labeled with red rectangles. Green boxes are bounding boxes of the detected vehicles. (a) Sunny daytime. (b) Rainy daytime. (c) Cloudy. (d) Dusk. (e) Nighttime. (f) Rainy nighttime.

where x_{k-1} is the state of previous time $k - 1$; F is the state transition matrix; $\widehat{x}_k$ is the state of current time; P_{k-1} and $\widehat{P}_k$ are error covariance matrices of the previous time $k - 1$ and current time k, respectively; and Q is the process noise covariance matrix.

2) Update: We selected the closest vehicle around the prediction location. If the distance between them was below the threshold, it was considered the observation y_k. If an observation was unavailable, the update step would be skipped, and multiple prediction steps were performed. To perform the update, the Kalman gain K_k is computed in

$$K_k = \widehat{P}_k H^t (H \widehat{P}_k H^t + R)^{-1} \tag{12}$$

where H is the measurement matrix and R is the measurement noise covariance matrix. Then, we updated the state vector and the error covariance matrix

$$x_k = \hat{x}_k + K_k (y_k - H \hat{x}_k)$$
$$P_k = (I - K_k H) \widehat{P}_k. \tag{13}$$

After vehicle tracking, the vehicles corresponded between adjacent frames and vehicle trajectories were estimated, as shown in Fig. 12.

C. Improved Detection by Tracking

The camera view is for rear-facing vehicles. When the vehicle enters the camera field of view, the vehicle parts can be seen without any occlusion. With the vehicle moving forward, the camera perspective is lower, which may cause severe occlusion of the license plate. Coupled with lower resolution in the distance, the plate is easily lost. However, the rear lamps are usually localized. We wish to detect the vehicle robustly in this case.

We added the prediction of the tracker into the MRF graph as a license plate node. In case that the plate was not localized, as shown in Fig. 8(a), the MRF model was constructed using the predictions and rear lamps. The vehicle parts are then correctly inferred, as in Fig. 8(b). Finally, according to the node marginal probabilities, we used the NMS method to eliminate repeated

inferences. In addition, in poor lighting conditions, the license plate is difficult to see, particularly due to the limited scope of the supplemental lighting equipment at night. On the other hand, rear lamps are a more prominent vehicle feature. This technique performs more effectively at night.

V. EXPERIMENTAL RESULTS AND ANALYSIS

A. Data

To test and evaluate our method, we applied it to the urban traffic road for vehicle detection and tracking. We selected a busy traffic intersection in China as the actual testing environment. The intersection included four lanes and one crosswalk. We collected image sequences using a high-resolution charge-coupled device camera.

We selected six scenarios including sunny daytime, rainy daytime, cloudy, dusk, nighttime, and rainy nighttime, as shown in Fig. 9. Each data set is a continuous video recording. The frame rate is 8.3 frames/s, except for the nighttime data set, which is 11.1 frames/s. For each data set, there were sedan vehicles, sports utility vehicles (SUVs), and buses, and we labeled them with ground-truth rectangles manually. In each environment, vehicles were occluded at different levels, and their pose transformations ranged from about $-45°$ to $45°$. Traffic congestion was divided into three levels: heavy traffic corresponding to frequent vehicle occlusion; moderate traffic, which means usual vehicle adhesion; and slight traffic with no vehicle occlusion. In particular, in the night condition, stroboscopic supplemental lighting equipment was used. Detailed information for all data sets is listed in Table I.

B. Performance

Recall and precision of our method were evaluated for both the vehicle and its parts detection performance, as shown in Table II. The performance of the part detection is highly related to that of the vehicle detection. We tried to ensure high recall rates of the vehicle parts since they determine the upper limits of the vehicle detection rate. As in Table II, the average recall rate of the license plate is higher than 98%, and the recall

TABLE I
TEST DATA SET DESCRIPTION

| Scenarios | Resolution | Frames | Vehicles | | Congestion |
			Pass	Total	
Sunny daytime	2592*1936	500	48	1777	Heavy
Rainy daytime	2592*1936	330	24	595	Moderate
Cloudy	2592*1936	500	18	2265	Heavy
Dusk	2592*1936	500	44	1844	Heavy
Nighttime	1920*1072	257	17	312	Moderate
Rainy nighttime	2592*1936	417	10	550	Slight

TABLE II
SYSTEM PERFORMANCE

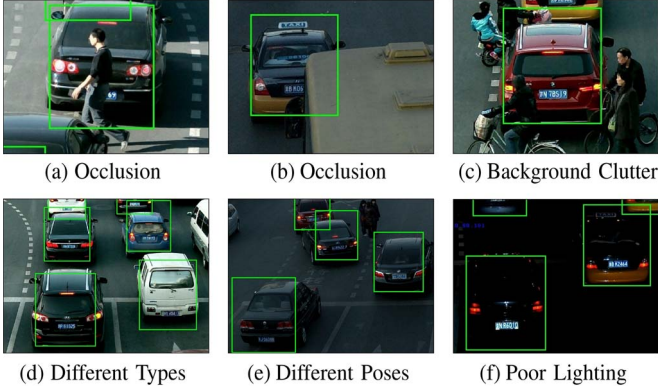| Datasets | Vehicle Part | Detection | | Tracking |
		Recall	Precision	MT
Sunny daytime	License Plate	95.0%	85.7%	89.6%
	Rear-lamp	91.4%	73.5%	
	Vehicle	94.3%	96.5%	
Rainy daytime	License Plate	99.8%	82.3%	91.7%
	Rear-lamp	90.2%	76.6%	
	Vehicle	93.8%	95.2%	
Cloudy	License Plate	99.1%	84.7%	94.4%
	Rear-lamp	88.3%	75.8%	
	Vehicle	97.9%	90.9%	
Dusk	License Plate	98.8%	78.2%	90.9%
	Rear-lamp	87.8%	74.5%	
	Vehicle	94.5%	97.5%	
Nighttime	License Plate	98.0%	92.0%	88.2%
	Rear-lamp	91.8%	90.9%	
	Vehicle	95.2%	90.5%	
Rainy nighttime	License Plate	99.3%	97.7%	90.0%
	Rear-lamp	90.3%	88.7%	
	Vehicle	95.8%	85.7%	

Fig. 10. Vehicle detection results in some challenging cases. Green boxes are bounding boxes of the detected vehicles. (a) Occlusion by a pedestrian. (b) Occlusion by a bus. (c) Cluttered background with several pedestrians and bicyclists. (d) Different vehicle types including sedan vehicles, vans, and SUVs. (e) Different vehicle poses. (f) Poor lighting condition during nighttime.

Fig. 11. ROC curve for vehicle detection.

rate of the rear lamp is higher than 89%. The final detection performance of the vehicle is highly superior to those of the parts. Our method achieved an average vehicle recall rate of 95% with an average precision rate of 92% for all the scenarios.

Fig. 9 shows some detection results of the vehicle and its parts in various scenarios. For each scenario, the left side is the part detection result, and the right side is the vehicle detection result. The detected plates are indicated by yellow rectangles, and rear lamps are indicated by red rectangles. Most of the vehicle parts were detected. A few false positives of the rear lamps were caused by red cars in Fig. 9(c) and vehicle reflection on a rainy day and nighttime in Fig. 9(b), (e), and (f). However, they did not affect the final vehicle detection because we also utilized the license plate part in the MRF model. The detected vehicles are indicated by green rectangles. As shown in Fig. 9(a)–(f), the proposed method could detect vehicles in various weather and lighting conditions. Even in some challenging situations, the method can robustly detect vehicles, as shown in Fig. 10. The main challenges included occlusion, background clutter, different vehicle types, different vehicle poses, and poor lighting. Since our vehicle detection method is based on detection of multiple salient parts and the probability model, it can deal with the challenging cases.

The receiver operating characteristic (ROC) curve of vehicle detection is drawn by adjusting the scoring thresholds in the license plate localization. We tested all data sets in different scenarios to get the summary ROC curve. As shown in Fig. 11, the true positive (TP) rate of vehicle detection increased with
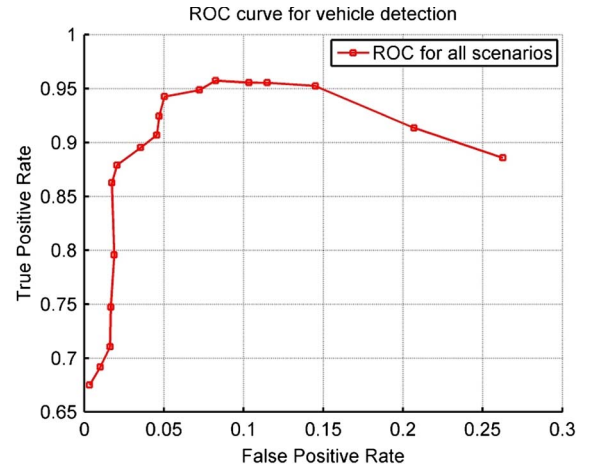
the false positive (FP) rate at the beginning. When the FP rate of the plate increased, vehicle detection was degraded because of using incorrectly detected plates. Therefore, the TP rate decreased when the FP rate exceeded a certain limit. With the ROC curve, we can choose a relatively good scoring threshold for all scenarios.

We evaluated the tracking performance using the indicator of mostly tracked trajectories (MT). If more than 80% of a ground-truth trajectory is correctly tracked, it is called an MT. The tracking results are listed in Table II. It is shown that our tracking method achieved an average tracking rate of 90%. Tracking results are better during nighttime in slight traffic congestion than those during sunny daytime for heavy traffic. Fig. 12 shows the tracking results of an intersection during sunny daytime. The proposed method could robustly detect preceding vehicles that change its direction. It can robustly track the vehicle even with pedestrian occlusion, as shown by
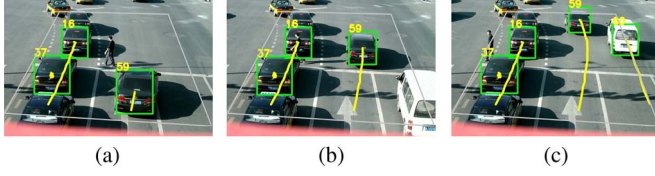
(a) (b) (c)

Fig. 12. Tracking example during sunny daytime. The vehicles are marked with green rectangles. Each vehicle is assigned with a yellow number. The trajectories are drawn with yellow lines. (a) Frame 299. (b) Frame 307. (c) Frame 315.

TABLE III
CRITICAL PARAMETERS OF THE ALGORITHM

Processes	Parameters	Parameter Selection
License Plate	Score Threshold α	Empirical Value, $[10, 60]$
Rear-lamp	Area Threshold γ_1 and γ_2	Depending on Traffic Scenes, $[10 * 10, 150 * 150]$
Node Potential	Sigmoid Parameters, A_p and B_p	Maximum Likelihood Estimation
Edge Potential	GMM Parameters, α_k and θ_k	Expectation Maximization

TABLE IV
AVERAGE RUN TIME PER FRAME BY PROCESS

Processes	Sub-processes	Run-time
License Plate Localization	Color Conversion	5 ms
	Image Gradient Computation	17 ms
	Plate Score Calculation	37 ms
	Refinement with Consistency	3 ms
Rear-lamp Localization	Color Conversion	3 ms
	Multi-threshold Segmentation	13 ms
	Connected Component Analysis	9 ms
Vehicle Inference	-	5 ms
Vehicle Tracking	-	3 ms
Total Run-time	-	95 ms

the vehicle labeled as number 16. As discussed in Section IV, we introduced the vehicle predictions of the previous frame for vehicle detection. Therefore, the distant vehicles can be correctly tracked even if their plates are not detected.

The learning of critical parameters for the vehicle detection algorithm is an important issue in practice. Here, we list the critical parameters of the algorithm and their learning methods, as shown in Table III.

C. Computational Cost

Our algorithm was mainly implemented with C++ and OpenCV libraries. The platform was configured with a 2.5-GHz Intel Core i5 3210M processor and a 4-GB memory device. To evaluate the running time of our algorithm, we divided it into the following steps: 1) license plate localization; 2) rear-lamp localization; 3) vehicle inference; and 4) vehicle tracking. Detailed computation time of each step is listed in Table IV. Our algorithm only processed the region of interest (ROI), which was about two million pixels. The running time per
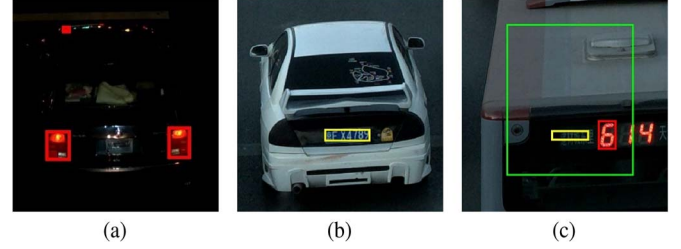

(a) (b) (c)

Fig. 13. Failure of detection cases. (a) and (b) Two false negative examples. (c) False positive example.

frame was no more than 100 ms, and this satisfied real-time performance on this processing platform.

D. Detection Error Analysis

Some of the failure results of the proposed method are shown in Fig. 13. False negatives tend to occur in the case of vehicle parts missing detection. For example, the license plate of the vehicle is blurred or there is no plate, as in Fig. 13(a). Rear lamps may not be detected if they are repainted with other colors, as in Fig. 13(b). These false negatives can be reduced by only utilizing detected parts. However, this causes a tradeoff by increasing false positives. An example of a false positive is shown in Fig. 13(c). There are some plate-like and lamp-like regions on the vehicle. False positives tend to occur on long vehicles and repainted vehicles. They can be eliminated by introducing more vehicle parts.

VI. CONCLUSION

In this paper, we have proposed a rear-view vehicle detection and tracking method based on a high-resolution camera. First, we localized the salient parts of the vehicle, including the license plate and rear lamps. Then, we constructed an MRF model by treating vehicle parts as graph nodes. LBP was used to infer the MRF graph to obtain the vehicle locations. After vehicle detection, we implemented vehicle tracking using KF. We realized a detection-by-tracking technique in which the prediction locations of KF were added into the MRF model as graph nodes. We carried out experiments in practical urban scenarios. Our method could adapt to partial occlusion and various weather conditions. The experiments showed that the proposed method could achieve real-time performance. This method could provide good results even in complex scenarios such as in a busy traffic intersection.

In the future, we will localize more salient parts, such as the windshield and front cover, aiming to deal with more severe occlusion and vehicle posture variation. The inference process will also be improved for better inference performance according to the comprehensive description in [17]. In addition, the vehicle detection and tracking process can be integrated into an embedded camera platform as a low-cost implementation.

REFERENCES

[1] F.-Y. Wang, "Parallel control and management for intelligent transportation systems: Concepts, architectures, and applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 630–638, Sep. 2010.
[2] W. Wang, Y. Wang, F. Chen, and A. Sowmya, "A weakly supervised approach for object detection based on soft-label boosting," in *Proc. IEEE Workshop Appl. Comput. Vis.*, 2013, pp. 331–338.
[3] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 1999, vol. 2, pp. 246–252.
[4] R. Cucchiara, M. Piccardi, and P. Mello, "Image analysis and rule-based reasoning for a traffic monitoring system," *IEEE Trans. Intell. Transp. Syst.*, vol. 1, no. 2, pp. 119–130, Jun. 2000.
[5] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection using gabor filters and support vector machines," in *Proc. IEEE Conf. Digital Signal Process.*, 2002, vol. 2, pp. 1019–1022.
[6] H.-Y. Cheng, C.-C. Weng, and Y.-Y. Chen, "Vehicle detection in aerial surveillance using dynamic bayesian networks," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 2152–2159, Apr. 2012.
[7] M. Cheon, W. Lee, C. Yoon, and M. Park, "Vision-based vehicle detection system with consideration of the detecting location," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 1243–1252, Sep. 2012.
[8] W. W. L. Lam, C. C. C. Pang, and N. H. C. Yung, "Vehicle-component identification based on multiscale textural couriers," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 4, pp. 681–694, Dec. 2007.
[9] L. Lin, T. Wu, J. Porway, and Z. Xu, "A stochastic graph grammar for compositional object representation and recognition," *Pattern Recogn.*, vol. 42, no. 7, pp. 1297–1307, Jul. 2009.
[10] B.-F. Lin, Y.-M. Chan, L.-C. Fu, P.-Y. Hsiao, L.-A. Chuang, S.-S. Huang, and M.-F. Lo, "Integrating appearance and edge features for sedan vehicle detection in the blind-spot area," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 737–747, Jun. 2012.
[11] J. Winn and J. Shotton, "The layout consistent random field for recognizing and segmenting partially occluded objects," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2006, vol. 1, pp. 37–44.
[12] D. Hoiem, C. Rother, and J. Winn, "3d layoutcrf for multi-view object class recognition and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–8.
[13] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.
[14] H. T. Niknejad, A. Takeuchi, S. Mita, and D. McAllester, "On-road multivehicle tracking using deformable object model and particle filter with improved likelihood estimation," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 748–758, Jun. 2012.
[15] Z. Xu and S. Sun, "Part-based transfer learning," in *Proc. Int. Conf. Adv. Neural Netw.*, 2011, pp. 434–441, ser. ISNN'11.
[16] S. Sun, Z. Xu, and M. Yang, "Transfer learning with part-based ensembles," in *Proc. Multiple Classifier Syst.*, 2013, pp. 271–282.
[17] S. Sun, "A review of deterministic approximate inference techniques for bayesian machine learning," *Neur. Comput. Appl.*, pp. 1–12, 2013.
[18] K. Robert, "Night-time traffic surveillance: A robust framework for multivehicle detection, classification and tracking," in *Proc. IEEE Conf. Adv. Video Signal Based Surv.*, 2009, pp. 1–6.
[19] W. Wang, C. Shen, J. Zhang, and S. Paisitkriangkrai, "A two-layer nighttime vehicle detector," in *Proc. Digit. Image Comput., Tech. Appl.*, 2009, pp. 162–167.
[20] W. Zhang, Q. M. J. Wu, G. Wang, and X. You, "Tracking and pairing vehicle headlight in night scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 1, pp. 140–153, Mar. 2012.
[21] R. O'Malley, E. Jones, and M. Glavin, "Rear-lamp vehicle detection and tracking in low-exposure color video for night conditions," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 2, pp. 453–462, Jun. 2010.
[22] D.-Y. Chen, Y.-H. Lin, and Y.-J. Peng, "Nighttime brake-light detection by nakagami imaging," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 4, pp. 1627–1637, Dec. 2012.
[23] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Comput. Surv.*, vol. 38, no. 4, pp. 1–45, 2006.
[24] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 5, pp. 564–577, May 2003.
[25] P. L. M. Bouttefroy, A. Bouzerdoum, S. L. Phung, and A. Beghdadi, "Vehicle tracking by non-drifting mean-shift using projective kalman filter," in *Proc. Int. IEEE Conf. Intell. Transp. Syst.*, 2008, pp. 61–66.
[26] F. Devernay, "A non-maxima suppression method for edge detection with sub-pixel accuracy," INRIA, SophiaAntipolis, France, Tech. Rep. rR-2724, 1995.
[27] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," *Image Vis. Comput.*, vol. 22, no. 10, pp. 761–767, Sep. 2004.
[28] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in Large Margin Classifiers*. Cambridge, MA, USA: MIT Press, 1999, pp. 61–74.
[29] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA, USA: Morgan Kaufmann, 1988.
[30] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Trans. ASME, J. Basic Eng.*, vol. 82, pp. 35–45, 1960.
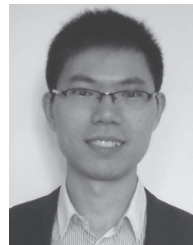
Bin Tian received the B.S. degree from Shandong University, Jinan, China, in 2009. He is currently working toward the Ph.D. degree in control theory and control engineering with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China.

His research interests include intelligent transportation systems, pattern recognition, and computer vision.

Ye Li received the B.S. degree in automation from University of Science and Technology Beijing, Beijing, China, in 2009. She is currently working toward the combined Master's and Doctoral degree in control theory and control engineering with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing.

Her research interests include intelligent transportation systems, computer vision, and pattern recognition.

Bo Li received the B.S. degree from Southeast University, Nanjing, China, in 2009. She is currently working toward the Ph.D. degree in control theory and control engineering with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China.

Her research interests include image processing and computer vision and their applications in intelligent transportation systems.

Ding Wen (M'95–SM'99) is a Professor with National University of Defense Technology (NUDT), Changsha, China, and a Senior Advisor with the Research Center for Military Computational Experiments and Parallel Systems Technology, NUDT. He has published extensively and received numerous awards. His main research interests include behavioral operation management, human resource management, management information systems, and intelligent systems.