Sheng *et al. EURASIP Journal on Image and Video Processing* 2012, **2012**:2
http://jivp.eurasipjournals.com/content/2012/1/2

EURASIP Journal on Image and Video Processing
a SpringerOpen Journal

RESEARCH **Open Access**

Robust multiple-vehicle tracking via adaptive integration of multiple visual features

Hao Sheng[1,2], Qi Wei[3], Chao Li[2,3*] and Zhang Xiong[2,3]

Abstract

This article presents a robust approach to tracking multiple vehicles with integration of multiple visual features. The observation is modeled by democratic integration strategies according to the reliability of the information in the current multi-visual features to adjust their weights. The appearance model is also embedded in a particle filter (PF) tracking framework. Furthermore, we propose a new model updating algorithm based on the PF. In order to avoid incorrect results caused by "model drift" introduced into the observation model, model updating should only be controlled in a reliable manner, and the rate of updating is based on reliability. This article also presents the experiments using a real video sequence to verify the proposed method.

Keywords: multiple vehicle tracking, multiple visual features, adaptive integration, model updating

1. Introduction

With the rapid process of urbanization, the concept of developing a "smart city" has gained prominence. As an important part of this trend, Intelligent Transportation Systems (ITS) will be critical for effective management of urban traffic. Vehicle tracking under different traffic scenarios is one of the key issues in ITS. Vehicle motion parameters, such as location, velocity, orientation, and acceleration, can be obtained to further recognize and understand vehicular behavior. However, the challenges of robust tracking come from uncertain and dynamic conditions of speed, occlusion, deformation, illumination variation, background clutter, real-time restriction, etc. In order to handle these problems, great effort has been made to devise robust tracking algorithms. In general, the following three key problems should be solved in tracking: (1) an effective framework to locate vehicles in motion; (2) modeling observation of vehicles; and (3) reliable updating of vehicle models.

An ideal locating framework should be able to predict and update the motion state and observation model of an object, and even track multiple objects under various conditions. Probabilistic tracking, which is a process utilizing posterior probability density of target states in a Bayesian framework, is a highly effective approach. Kalman Filter (KF) [1], Hidden Markov Model [2], and Particle Filter (PF) [3-7] techniques have been used in different tracking applications. PF recursively constructs the posterior probability distribution function of the state space using a Monte Carlo integral. A PF-based tracking algorithm has the added advantage that any visual feature can be used for the observation model. Meanwhile, it has the ability to integrate multiple visual features.

The observation model depicts similarity measurements between a template region and the candidate region of a vehicle, and plays an even critical role in visual tracking associated with PF. Many visual features can be selected for vehicle observation modeling, including color [8], edge [3,9], feature descriptors [10], color-spatial features [11], wavelets [12], etc. For instance, it is sensitive to the variation of a given illumination environment using a color-based method when the illumination varies. An edge-based method can avoid disturbances caused by illumination variances, but it is either time-consuming or limited to a single shape model, and presents difficulties in achieving accurate real-time tracking. The algorithms based on these methods have achieved good tracking performance, but relying on a single visual feature is often inadequate and unstable in some complex tracking scenarios. Various complementary features can be combined to derive more robust tracking results. It is our interest to employ

* Correspondence: licc@buaa.edu.cn
[2]Research Institute of Beihang University in Shenzhen, Shenzhen 518057, People's Republic of China
Full list of author information is available at the end of the article

multiple visual features under a robust tracking framework. The advantage of this kind of method is that vehicle information can complement other visual features. When one visual feature fails, others can be used to maintain tracking. However, the difficulty is how to design a good strategy to integrate visual features reliably. Methods proposed in [13-15] are based on the fixed weight integration. If one visual feature with a fixed weight changes markedly, the observation model after integration will be unreliable. This leads to tracking drift away from the true location, and even tracking failure. Spengler and Schiele [16] proposed an algorithm with an adaptive integration strategy using an EM algorithm to adjust the weight of each visual feature online. However, this algorithm is based on a matching algorithm under a local search. When partial or complete occlusion occurs, tracking performance will seriously decline.

How to update a vehicle model to deal with appearance changes during tracking is very important for the robustness of an algorithm. Many algorithms assume the appearance of an object as being invariable during tracking. The appearance model of an object is usually extracted in the first frame image, and then the most probable location of the object is found in the following frame. This assumption is reasonable for short-term tracking. However, for long-term tracking, appearance changes in an object are inevitable. Jepson et al. [17] proposed an adaptive texture-based model named *WSL*. This model consists of three components to describe object appearance changes, where *W* describes the rapid changes in object appearance, *S* is used to characterize the stability in an object whose change in appearance is slow, and *L* is defined to depict abnormal variations in object appearance. A Gaussian Mixture Model (GMM) is constructed by these components, and the parameters of GMM are updated through the EM algorithm online. The proposed model has strong robustness to changes in both illumination and shape. However, it fails to track when an object is occluded by one with the same visual features for even a moment. The reason is that the same information presented by the occluding object is added into the model of the occluded object during updating. After occlusion, the appearance model cannot correctly reflect the object. This phenomenon is called "model drift". A fixed number of pre-learned exemplars are used as templates by Toyama and Blake [18]. The problem with this method is that only a fixed number of examples can be used as templates to model the appearance of an object. Yang and Wu [19] introduced a closed-form solution by "discriminative training" of a generative model to alleviate model drift. They optimize a convex combination of the generative and discriminative log likelihood functions to obtain the model. Avidan [20] treated tracking as a classification problem. The ensemble of weak classifiers is combined into a strong classifier using AdaBoost. The strong classifier is then used to label pixels in the next frame as either belonging to the object or the background, creating a confidence map. The new position of the object is found in the peak of the map by using a mean shift. However, only the color of each pixel is used to classify, and the classifier needs to update background information around the object. When two objects of a similar color are very near each other, tracking will fail.

This article proposes a robust tracking approach with an adaptive integration of multiple visual features for vehicles. A color histogram and an edge orientation histogram (EOH) are selected as visual features to model the observation of the vehicle and integrated by a democratic integration strategy proposed by Triesch and Malsburg [21]. It is suitable for dynamic scenes due to the adaptive adjustment of the weight of each visual feature with its reliability in the current frame. However, deterministic integration is vulnerable to occlusion for a few frames because the present iteration is initialized according to the previous one. Thus, the observation model is embedded in the PF tracking framework. In order to improve the robustness of object representation, spatial information is incorporated into the observation model by dividing the object to be tracked into a number of fragments. We then analyze the reason for model drift during the model update process, and propose a new model updating method under a PF. In order to avoid errors caused by model drift, the updating process should only be implemented in a reliable manner, and the rate of updating can be controlled according to this reliability. The posterior probability density function of distribution of state vector and similarity between the candidate and reference observation of an object are used to define the valid measurement of the reliability to model updating during tracking. Experimental results in real traffic surveillance video sequences show that our approach outperforms others in vehicle tracking under complex conditions.

The remainder of this article is organized as follows. The preprocessing before tracking is described in Section 2. The state model for multiple vehicles is built in Section 3. The adaptive and robust observation model is presented in Section 4. The reliable model updating strategy is introduced in Section 5. The PF-based tracking algorithm is completely summarized in Section 6. The experimental results are given in Section 7, and finally, the conclusion is given in Section 8.

2. Preprocessing before tracking

2.1. Background modeling

In surveillance video, it can be seen that the background changes along with illumination, weather, and other conditions. So, we must process the surveillance video scene first. Our previous research has given a self-adaptive modeling for real-time background modeling with lower computational-complexity and higher accuracy [22].

In the $(n + 1)$th frame, the gray value of point p can be described as follows:

$$G(n + 1, p) = G(n, p) + L(n, p) + \text{noise}^1(n, p) \tag{1}$$

where $G(n, p)$ is the pixel p's gray value in the nth frame, $L(n, p)$ is the model to describe the change of illumination with the change of time, and $\text{noise}^1(n, p)$ is the Gaussian noise taking zero as the center. The gray value of pixel p in the input image can be described as:

$$I(n, p) = G(n, p) + \text{noise}^2(n, p) \tag{2}$$

where $\text{noise}^2(n, p)$ is Gaussian noise taking zero as the center. A comparison between (1) and (2) can easily indicate that

$$I(n + 1, p) = G(n, p) + \omega(n + 1, p) \tag{3}$$

where $\omega(n+1, p) = L(n, p)+\text{noise}^1(n, p)+\text{noise}^2(n+1, p)$. $\omega(n, p)$ is a Gaussian distribution. We use a mean value to represent $m(n, p)$ and $s(n, p)$, respectively, and use a variable to represent $\omega(n, p)$. In traffic surveillance video, illumination and noise distribution change little in a triangular region. Therefore, $m(n, p)$ and $s(n, p)$ are independent of the position of pixel p. Then, a histogram can be derived from the difference between $\{I(n+1, p)\}$ and $\{G(n, p)\}$ in a triangular region. From this histogram, the mean value of $m(n)$ and $s(n)$ can be estimated by a self-adaptive filter based on a recursive least square method.

Figure 1 gives the background in four video surveillance scenes, where the regions masked in red are not our monitoring driveways.

2.2. Detecting the ROI of a vehicle

The aim is to track vehicle targets in real time and as robustly as possible. The first step is to detect the vehicle targets automatically. The initial detection and the tracking regions are set in the field of vision, respectively, as shown in Figure 2.

Then, due to its success in vehicle detection of a real surveillance scene, a fast-constrained Delaunay triangulation (FCDT) algorithm [23] is used as follows:

(1) Extract contour information with a Canny filter
(2) Extract lines from image contours with a Hough transformer
(3) Achieve a set of corners at both ends of the lines
(4) Initialize the CDT based on all constrained edges
(5) Insert all corner points in turn to reconstruct the CDT
(6) Extract corner density, horizontal straight line density, vertical straight line density, triangle density, and average intensity of a vehicle region to construct the feature vector
(7) Put the feature vector into SVM to determine the ROI of the vehicle target

The detection results are shown in Figure 3. The blue lines construct the Delaunay triangulation net. The red rectangle bounding box is the ROI of the vehicle.

3. State modeling for multiple vehicles

According to the characteristics of vehicle motion, we build the prediction equation of the motion state using the second-order linear regression. The state model is built by a centroid and the area of a rectangular bounding box:

$$\mathbf{S} = (x, y, s)^{\mathrm{T}} \tag{4}$$

where $\mathbf{C} = (x, y)$ and s are the centroid and area bounding box, respectively.

The current state $\mathbf{S}_t$ is predicted by three parts: the previous state $\mathbf{S}_{t\text{-}1}$, the last state displacement $\mathbf{S}_{t\text{-}1}$ - $\mathbf{S}_{t\text{-}2}$, and a zero-mean Gaussian stochastic component ω_t with covariance matrix Σ:

$$\begin{cases} \mathbf{S}_t - \mathbf{S}_{t-1} = \mathbf{S}_{t-1} - \mathbf{S}_{t-2} + \omega_t \\ \omega_k \sim N(0, \Sigma) \end{cases} \tag{5}$$

Hence, the model can be denoted as a Gaussian distribution as follows:

$$p(\mathbf{S}_t | \mathbf{S}_{t-1}, \mathbf{S}_{t-2}, ..., \mathbf{S}_1) \sim N(\mathbf{S}_t; 2\mathbf{S}_{t-1} - \mathbf{S}_{t-2}, \Sigma) \tag{6}$$

For multiple vehicles, we suppose that vehicles are independent from each other and there are M vehicles in a video scene. So, the model is regarded as

$$p(\mathbf{S}_t(m) | \mathbf{S}_{t-1}(m), ..., \mathbf{S}_1(m)) \sim N(\mathbf{S}_t(m); 2\mathbf{S}_{t-1}(m) - \mathbf{S}_{t-2}(m), \Sigma(m)) \tag{7}$$

where $\mathbf{S}_t(m)$ is the state vector of the mth vehicle in the kth frame.

4. Adaptive integration-based observation model

The observation models encode the visual information of a vehicle's appearance. Since a single visual feature does not work in all cases, we utilize the Hue-Saturation-Value (HSV) color histogram to capture the color information of a vehicle, and an EOH to encode shape information, indicating that $\mathbf{O} = \{\mathbf{O}_t;\ t \in N\}$ is denoted as the vehicle's observation model.

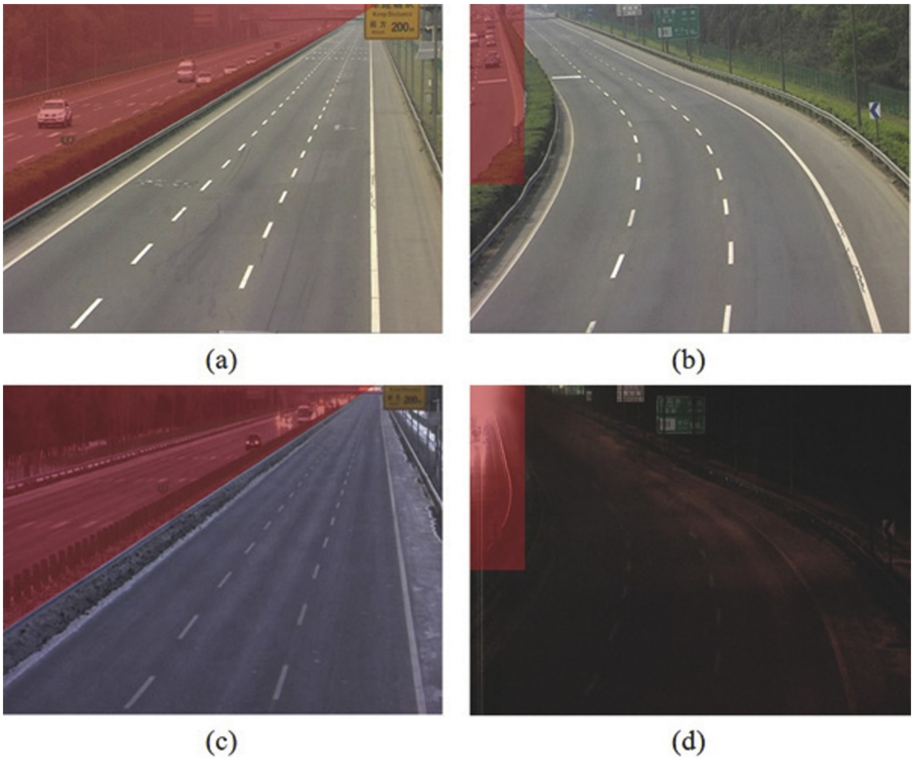


Figure 1 The results of background estimation. **(a)** Straight driveway. **(b)** Turning driveway. **(c)** Straight driveway in the evening. **(d)** Turning driveway late at night.

4.1. Color features

We obtain the color information of a vehicle by a two-part color histogram based on the HSV color space. We use the HSV color histogram because it decouples the intensity from Hue and Saturation, and thus it is less sensitive to illumination effects than a histogram from the RGB color space. The exploitation of the spatial layout of the color is also crucial due to the fact that different vehicles usually have different colors.

In the non-Gaussian state space, state model $\mathbf{S}$ is assumed to be a hidden Markov process, with an initial distribution $p(\mathbf{S}_0)$ and a transfer distribution $p(\mathbf{S}_t|\mathbf{S}_{t\text{-}1})$. A color histogram-based observation model $\mathbf{O}_t^c$ is obtained through the marginal distribution $p(\mathbf{O}_t^c|\mathbf{S}_t)$. Our color observation model is composed of a 2D histogram based on Hue and Saturation and a 1D histogram based on value. Both histograms are normalized such that all bins sum to one. We assign the same number of bins for

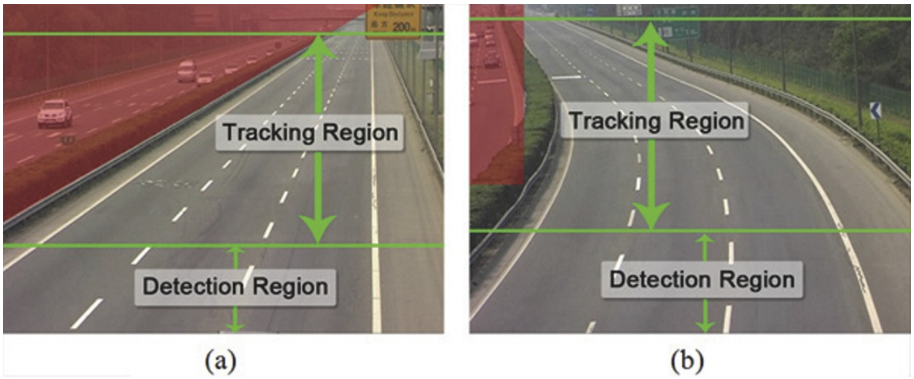


Figure 2 Initial detection region and tracking region.

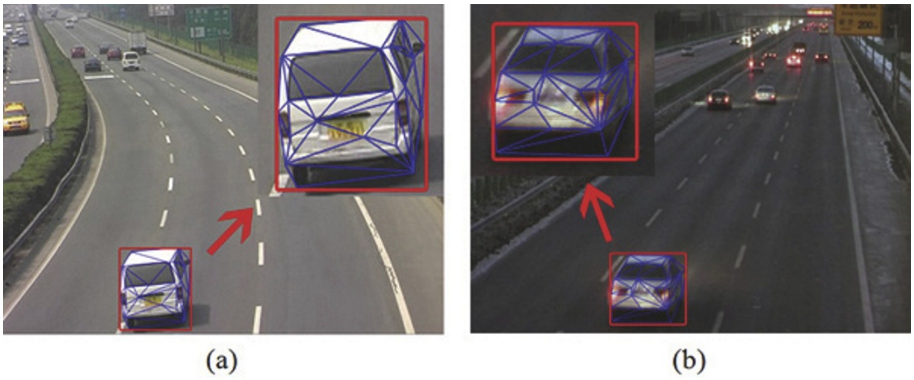


Figure 3 Results of detecting the ROI of the vehicle targets using FCDT.

each color component, i.e., $N_{\mathrm{h}} = N_{\mathrm{s}} = N_{\mathrm{v}} = 10$, resulting in an $N = N_{\mathrm{h}} \times N_{\mathrm{s}} + N_{\mathrm{v}} = 110$-dimensional HSV histogram.

Assume that $R(\mathbf{S}_t)$ is the candidate region of vehicle at time t, the kernel density estimation of color distribution is

$$k(n;\mathbf{S}_t) = \kappa \sum_{\mathbf{d}\in R(\mathbf{S}_t)} \delta[b_t(\mathbf{d}) - n] \tag{8}$$

where $b_t(\mathbf{d}) \in \{1, ..., N\}$ is the index of color bins of a pixel at position $\mathbf{d}$; $\delta[\cdot]$ is the delta function; κ is a normalized factor to subject to $\sum_{n=1}^{N} k(n;\mathbf{S}_t) = 1$; position $\mathbf{d}$ is a pixel in the candidate region $R(\mathbf{S}_t)$. Suppose that $\mathbf{K}^* \triangleq \{k^*(n;\mathbf{S}_0)\}_{n=1,\ldots,N}$ is the reference template and $\mathbf{K}(\mathbf{S}_t) \triangleq \{k(n;\mathbf{S}_t)\}_{n=1,\ldots,N}$ is the candidate model, the similarity measurement is defined based on Bhattacharyya coefficient:

$$\rho_c(\mathbf{K}^*, \mathbf{K}(\mathbf{S}_t)) = \left(1 - \sum_{n=1}^{N} \sqrt{k^*(n;\mathbf{S}_0)k(n;\mathbf{S}_t)}\right)^{\frac{1}{2}} \tag{9}$$

Therefore, the color-based observation model is denoted as follows:

$$p(\mathbf{O}_t^c|\mathbf{S}_t) \propto e^{-\lambda_c \rho_c^2(\mathbf{K}^*, \mathbf{K}(\mathbf{S}_t))} \tag{10}$$

where λ_c is a factor determined by the variation of color Gaussian distribution. Figure 4 shows the HSV color histograms of two vehicles.

4.2. Shape features

We apply an EOH to describe shape information of a vehicle. In order to detect the edge, the color image must be converted to grayscale at first. The gradient at pixel (x, y) in the image I can be computed by the Sobel operator mask:

$$G_h(x,y) = \mathrm{Sobel}_{\mathrm{h}} \times I(x,y) \tag{11}$$

$$G_v(x,y) = \mathrm{Sobel}_{\mathrm{v}} \times I(x,y) \tag{12}$$

where $\mathrm{Sobel}_{\mathrm{h}}$ and $\mathrm{Sobel}_{\mathrm{v}}$ are horizontal and vertical masks of the Sobel operator. The strength of an edge is computed as follows:

$$G(x,y) = \sqrt{G_h^2(x,y) + G_v^2(x,y)} \tag{13}$$

In order to suppress noise we threshold $G(x, y)$ such that

$$G'(x,y) = \begin{cases} G(x,y) & \text{if } G(x,y) \geq T \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

where the value of T was suggested to be set between 80 and 110 in [24]. The orientation of the edge is

$$\theta(x,y) = \begin{cases} \arctan\left(\dfrac{G_v(x,y)}{G_h(x,y)}\right) & \text{if } G_h(x,y) \neq 0 \\ \dfrac{\pi}{2} & \text{if } G_h(x,y) = 0 \end{cases} \tag{15}$$

Then, the edges are divided into K bins. The value of the kth bin is denoted as

$$\psi_k(x,y) = \begin{cases} G'(x,y) & \text{if } \theta(x,y) \in \mathrm{bin}_k \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

Figure 5 shows the EOHs of the vehicles in Figure 4.

Levi and Weiss [25] introduced three extended features based on EOH. However, direct use of these features for vehicles has some limitations. First, the values of both the ratio of edge strength of any two orientations and the dominant orientation feature have a large

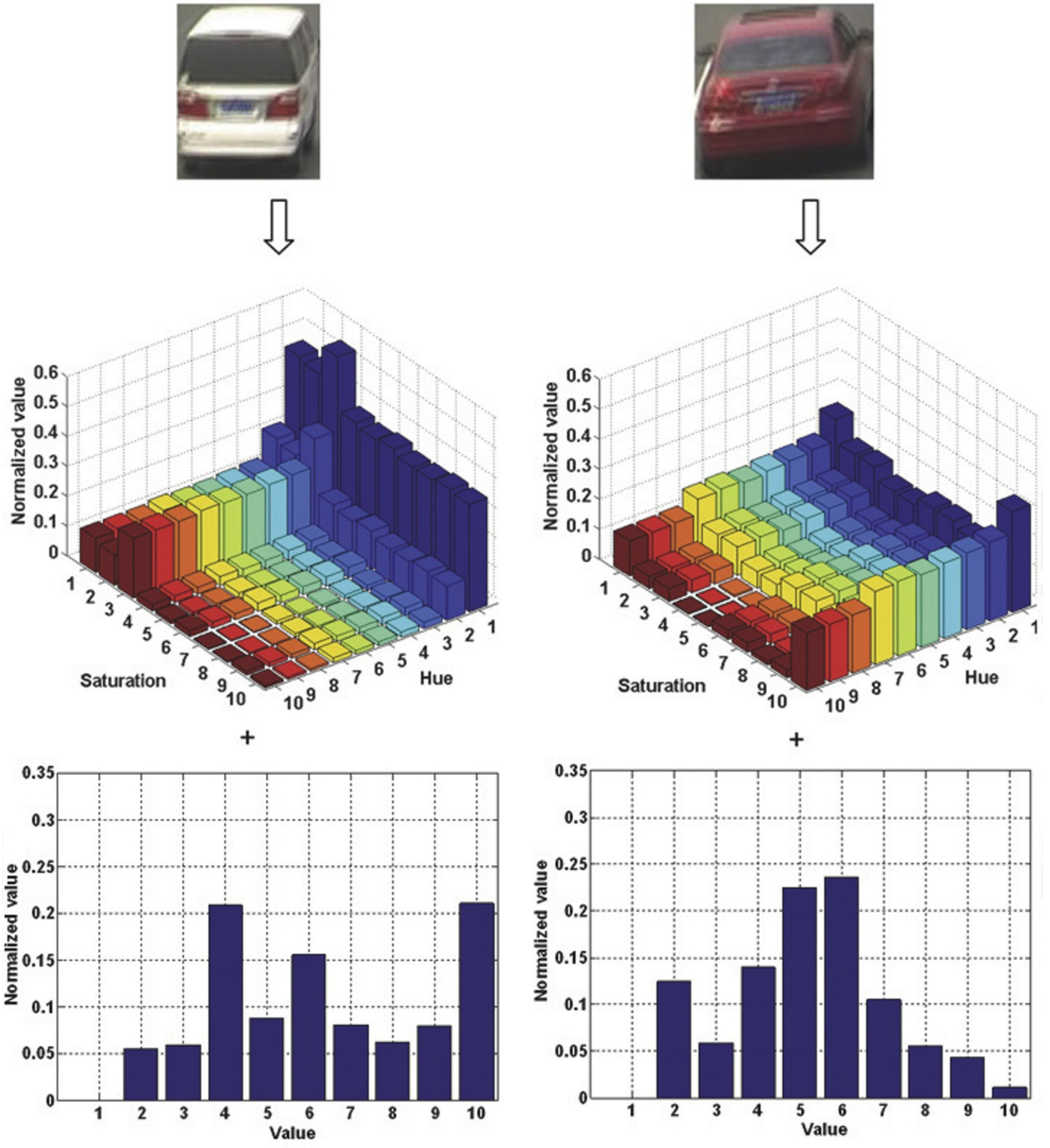


Figure 4 HSV color histograms of vehicles.

range, but the values of the discriminative features distribute into a relatively small scope. It cannot reflect characteristics of the majority of edges. Second, the orientations of symmetrical edges should be complementary instead of equal, because of the symmetry of two regions. Hence, we provide an enhanced feature set. These features are used to improve robustness in Section 4.3.

(1) Edge Strength Features in any two orientations φ:

$$\phi_{i,j}(q^l) = \arctan\left(\frac{E_i(q^l) + \varepsilon}{E_j(q^l) + \varepsilon}\right) \tag{17}$$

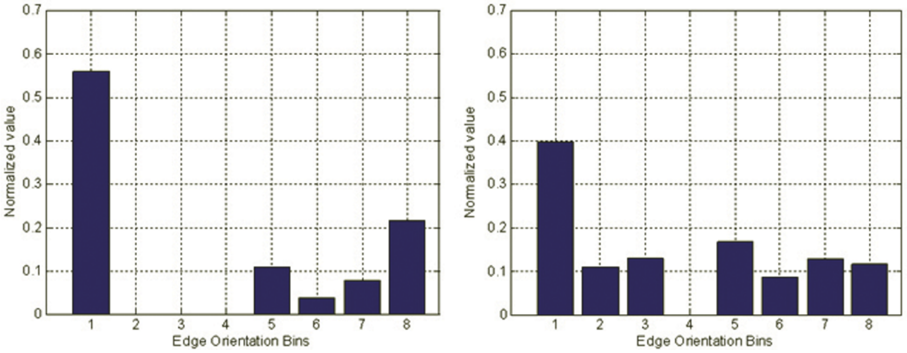


Figure 5 EOHs of vehicles.

(2) Dominant orientation features ϕ:

$$\varphi_i(q^l) = \arctan\left(\frac{E_i(q^l) + \varepsilon}{\sum\limits_{j\in K} E_j(q^l) + \varepsilon}\right) \tag{18}$$

(3) Symmetry features ζ:

$$\zeta_1(R_1, R_2) = \arctan\left(\frac{E_i(R_1) - E_{\pi(i)}(R_2) + \varepsilon}{\sum\limits_{j\in K}(E_j(R_1) + E_j(R_2)) + \varepsilon}\right) \tag{19}$$

$$\zeta_2(R_1, R_2) = \arctan\left(\frac{E_i(R_1) + E_{\pi(i)}(R_2) + \varepsilon}{\sum\limits_{j\in K}(E_j(R_1) + E_j(R_2)) + \varepsilon}\right) \tag{20}$$

where R_1 and R_2 are regions of the same size and are positioned at opposite sides of the symmetry axes. $\pi(i) = (M_\zeta - i)\%M_\zeta$; M_ζ is interval numbers of $[0, \pi]$ and $M_\zeta = 6$ in the experiment.

Suppose that $\mathbf{E}^* \triangleq \{e^*(n; \mathbf{S}_0)\}_{n=1,\ldots,K}$ is the reference template and $\mathbf{E}(\mathbf{S}_t) \triangleq \{e(n; \mathbf{S}_t)\}_{n=1,\ldots,K}$ is the candidate model of EOH, the similarity measurement is defined as follows:

$$\rho_e(\mathbf{E}^*, \mathbf{E}(\mathbf{S}_t)) = \left(\sum_{n=1}^{K}(e^*(n; \mathbf{S}_0)e(n; \mathbf{S}_t))\right)^{\frac{1}{2}} \tag{21}$$

Therefore, the color-based observation model is denoted as follows:

$$p(\mathbf{O}_t^e|\mathbf{S}_t) \propto e^{-\lambda_e\rho_e^2(\mathbf{E}^*,\mathbf{E}(\mathbf{S}_t))} \tag{22}$$

where $\mathbf{O}^e$ is denoted as the observation model based on an EOH, and λ_e is a factor determined by the variation of the EOH distribution.

4.3. Improving robustness

Both visual features introduced above are based on histograms, while all spatial information is discarded. This may lead to false objects and local minima, and even tracking failure under occlusion. On the other hand, methods incorporating the spatial information are computationally intensive. Motivated by the approaches proposed in [26,27], spatial information is incorporated into the observation model by dividing the vehicle to be tracked into a number of fragments.

The reference observation of a vehicle is represented by multiple fragments using multiple feature histograms $\{q^l\}_{l\ =\ 1,\ \ldots,\ L}$ instead of one global histogram, where L is the number of fragments. Let the target candidate centered at position $\mathbf{C}$ be represented by $\{p^l(\mathbf{C})\}_{l\ =\ 1,\ \ldots,\ L}$, where $p^l(\mathbf{C})$ is built in the same manner as the observation model. With this definition, we propose the similarity function as follows:

$$\rho(\mathbf{C}) = \lambda^{(1)}\rho^{(1)} + \cdots + \lambda^{(L)}\rho^{(L)} = \sum_{l=1}^{L}\lambda^{(l)}\rho^{(l)} \tag{23}$$

where $\lambda^{(l)}$ describes the important weight of each fragment and subjects to $\sum_{l=1}^{L}\lambda^{(l)} = 1$. The similarity function of each fragment is calculated by similarity measurements of different features between $p^l(\mathbf{C})$ and q^l. During tracking, each fragment should play a role at different levels due to occlusions or other kinds of appearance changes. A higher value $\lambda^{(l)}$ means that the tracking algorithm will refer more to the lth fragment. Conversely, a fragment with little weight will count less for the final tracking result. Here, we regard a fragment as being more important if it is more similar with the reference fragment, and at the same time less similar with the background:

$$\lambda^{(l)} = \gamma \lambda_{\mathrm{fg}}^{(l)} + (1-\gamma)\lambda_{\mathrm{bg}}^{(l)} \tag{24}$$

where γ tunes the proportion of $\lambda_{\mathrm{fg}}^{(l)}$ and $\lambda_{\mathrm{bg}}^{(l)}$, that we set it 0.8 in the following experiments. The background region for each fragment is selected as the neighborhood surrounding region with a double size excluding the fragment. Accordingly, the feature histogram of the background region is extracted. To measure the similarity more properly, we use the metric proposed by Nummiaro et al. [28]:

$$\lambda_{\mathrm{fg}}^{(l)} = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\left(d_{\mathrm{fg}}^{(l)}\right)^2}{2\sigma^2}\right) \tag{25}$$

$$\lambda_{\mathrm{bg}}^{(l)} = 1 - \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\left(d_{\mathrm{bg}}^{(l)}\right)^2}{2\sigma^2}\right) \tag{26}$$

where $d^{(l)}$ is the distance of two feature histograms.

Many suggested methods [29-31] divide an object into multiple non-overlapping fragments. Note that both the number of fragments and their delineation have an impact on tracking efficiency and accuracy. Although the robustness increases with the number of fragments, too many fragments mean an increased processing time for each frame. Since the computation required for each frame greatly depends on the size of each fragment, which also needs to be restricted. Further, selecting very small fragments will result in tracking drift, or some information about the vehicle being discarded. So, a trade-off is required. We prompt the use of some overlapping fragments. A set of non-overlapping horizontal fragments and a set of non-overlapping vertical fragments are overlapped. Horizontal and vertical fragments are obtained by the dominant orientation features and the symmetry feature introduced in Section 4.2, respectively. When the size of the fragment is less than 8 × 8, it will be discarded. The satisfactory results of fragmentation are shown in Figure 6.

Hence, the color and EOH-based observation with fragmentation are denoted as follows:

$$p(\mathbf{O}_t^c|\mathbf{S}_t) \propto e^{-\lambda_c \sum_{l=1}^{L} \left(\rho_c^{(l)}(\mathbf{K}^*, \mathbf{K}(\mathbf{S}_t))\right)^2} \tag{27}$$

$$p(\mathbf{O}_t^e|\mathbf{S}_t) \propto e^{-\lambda_e \sum_{l=1}^{L} \left(\rho_e^{(l)}(\mathbf{E}^*, \mathbf{E}(\mathbf{S}_t))\right)^2} \tag{28}$$

4.4. Adaptive integration

We employ an adaptive integration of the multiple visual features mentioned above, i.e., democratic integration. This integration strategy changes each feature's weight adaptively, according to its reliability in the previous frame, and improves the performance robustness of the visual features.

The complete observation model is defined as

$$p(\mathbf{O}|\mathbf{S}) = \alpha^c p(\mathbf{O}^c|\mathbf{S}) + \alpha^e p(\mathbf{O}^e|\mathbf{S}) \tag{29}$$

where α^c and α^e are the weights of color histogram and EOH features, respectively, and $\alpha^c + \alpha^e = 1$. The final state vector can be obtained by the maximum likelihood estimation:

$$\hat{\mathbf{S}} = \arg\max_{\mathbf{s}}\{p(\mathbf{O}|\mathbf{S})\} \tag{30}$$

In order to verify the consistency between results by integration of multiple visual features and by a single feature, a quality function γ_t^f is introduced and normalized as follows:

$$\bar{\gamma}_t^f = \frac{\gamma_t^f}{\sum_f \gamma_t^f} \tag{31}$$

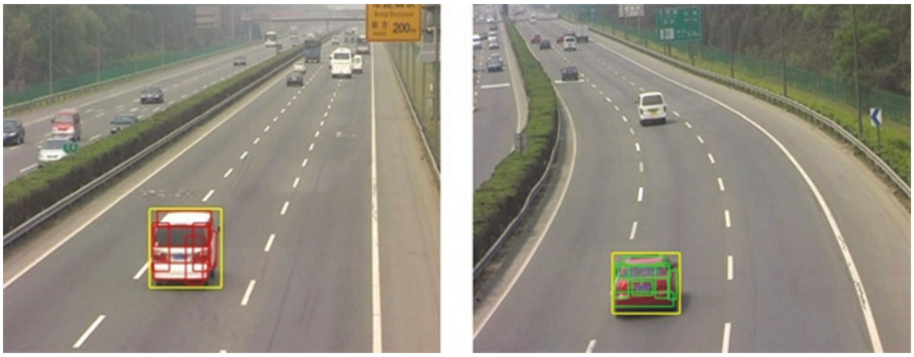


Figure 6 The satisfactory results of fragmentation.

where f is a sign to indicate the type of feature, i.e., color or EOH. In general, the change between two adjacent frames is small, so the weight of a feature can be predicted by

$$\tau\frac{\alpha_t^f - \alpha_{t-1}^f}{\Delta t} = \bar{\gamma}_{t-1}^f - \alpha_{t-1}^f \qquad (32)$$

where τ is a constant to determine the adaptive rate of change of weight; Δt is a continuous time interval between two frames. From Equation (32), the weight of feature whose current weight is less than the value of $\bar{\gamma}_{t-1}^f$ may be increased. That is to say that this strategy always increases the weight of a feature with a high reliability and reduces the weight with a low reliability.

In fact, γ_t^f can be treated as the feedback of the tracking result $\hat{\mathbf{S}}_t$. The weight of each visual feature is adaptively calculated by the normalized quality function in the previous frame. In order to define γ_t^f, we employ the probabilistic distribution map in [31]: $p_f(\mathbf{x}_i, t) \propto p_f(Z_i| M_{f,\ F})$. Z_i is the observation at pixel i; $M_{f,\ F}$ is the foreground model of feature f, $p_f(Z_i|M_{f,\ F})$ represents the observation likelihood of the pixel i given the foreground model $M_{f,\ F}$ of feature f. The higher the pixel's value in $p_f(\mathbf{x}_i, t)$ is, the higher the likelihood of pixel i belongs to the foreground. Hence, γ_t^f is defined as the ratio between the numbers of probabilistic pixels of foreground and background in the probabilistic distribution map:

$$\gamma_t^f = \frac{\mathrm{Sum}(p_f(\mathbf{x}_i, t), \hat{\mathbf{S}}_t)}{\mathrm{Sum}(p_f(\mathbf{x}_i, t), \hat{\mathbf{S}}_t' - \hat{\mathbf{S}}_t)} \qquad (33)$$

where the background is defined as the area between the tracking box and a larger window $\hat{\mathbf{S}}_t'$, which shares the same centroid of a bounding box. $\mathrm{Sum}(\cdot,\cdot)$ is the sum of probabilistic pixels in the window W:

$$\mathrm{Sum}(p(\mathbf{x}_i), W) = \sum_i p(\mathbf{x}_i), \quad \mathbf{x}_i \in W \qquad (34)$$

5. Model updating

Tracking is usually performed by searching for a location in the image that is similar to a given reference model. The updating of the observation model is implemented by the new appearance and a previous observation model O_1, ..., O_t to estimate to the observation model $\mathbf{O}_{t+1}$ in the next frame. Assume that the appearance of a vehicle remains the same during tracking, the observation model in the coming frame is

$$\mathbf{O}_{t+1} = \mathbf{O}_t \qquad (35)$$

It is reasonable for short-term tracking under some conditions. However in reality, vehicles will change appearance due to a variety of factors, such as turning, scale, camera angle, etc. Therefore, this assumption will eventually lead to some errors where the observation model cannot correctly represent the actual appearance of the vehicle. In order to obtain the latest and real observation model of a vehicle, a simple model updating strategy is proposed where the observation model in the next frame is estimated by the state vector of tracking results from the previous frame:

$$\mathbf{O}_{t+1} = p(\bar{\mathbf{S}}_t) \qquad (36)$$

where $\bar{\mathbf{S}}_t$ is the state of the vehicle at time t, and $p(\bar{\mathbf{S}}_t)$ is the observation estimation covered by $\bar{\mathbf{S}}_t$.

This updating strategy can make the observation model of a vehicle respond to appearance changes, but it easily leads to model drift when the vehicle is occluded by other vehicles or tracking errors, or the rapid deviations from the ground-truth of vehicle observation present during the updating process. Thus, we have created an adaptive update method to maintain stability over observation changes:

$$\mathbf{O}_{t+1} = (1 - \beta_t)\mathbf{O}_t + \beta_t p(\bar{\mathbf{S}}_t) \qquad (37)$$

where β_t is named as a forgetting factor, and it is used to minimize the impact on the observation model by specific frames and to control the speed of model updating. It is inevitable that some kinds of errors will be made during tracking. There exist two kinds of errors: errors caused by accumulation, and errors caused by object distortion. The former is caused by the accumulation of small errors from frequent updating; the latter is usually a fatal error which is induced by maintaining the same observation model during tracking. Therefore, the key problems are when to update the model and the rate of updating.

In a traffic scene, the changes of tracked vehicles usually fall into two categories: change of a vehicle's scale and changes in appearance. Therefore, we define two factors, $\eta_1(t)$ and $\eta_2(t)$, to determine the forgetting factor β_t at time t:

$$\beta_t = k\eta_1(t)\eta_2(t) \qquad (38)$$

where k is a constant.

First, when the appearance of a vehicle is obviously changed by occlusion, illumination, etc., a significant difference appears between the reference observation and the candidate one. At this time, updating should be avoided. Thus, $\eta_1(t)$ is defined using the similarity measurement between the candidate and the reference observation:

$$\eta_1(t) = \begin{cases} \rho(\mathrm{O}_t, \bar{\mathrm{S}}_t), & \rho(\mathrm{O}_t, \bar{\mathrm{S}}_t) > Th_1 \\ 0, & \text{otherwise} \end{cases} \tag{39}$$

where $\rho(\cdot,\cdot)$ is the similarity measurement between $\mathbf{O}_t$ and $\bar{\mathbf{S}}_t$. Th_1 is empirical and is set to 0.8 in the experiments. The bounding box scale changes due to the vehicle's motion trajectory. We employ the bounding box scale recursion introduced by McKenna et al. [32]:

$$\mu_{t+1} = c\mu_t + (1-c)s'_{t+1} \tag{40}$$

$$\sigma_{t+1}^2 = c(\sigma_t^2 + (\mu_{t+1} - \mu_t)^2) + (1-c)(s'_{t+1} - \mu_{t+1})^2 \tag{41}$$

where μ_{t+1} and σ_{t+1}^2 represent the new mean and the new variance of the recursive bounding box scale, respectively, and s'_{t+1} represents the newly detected bounding box scale. C is used to control the forgotten rate of the recursive bounding box scale of the vehicle. If c is large, the history of the bounding box scale will fade out slowly. This is good for a vehicle as a rigid object with a fixed shape, and the history of the bounding box scale will be kept through the large c. In the experiments, c is set to 0.9. Here, $\eta_2(t)$ is defined according to the new mean and variance:

$$\eta_2(t) = \begin{cases} CV(t), & 0 < CV(t) < Th_2 \\ 0, & \text{otherwise} \end{cases} \tag{42}$$

where $CV(t) = \sigma_t/\mu_t$ is the dispersion coefficient. Th_1 is empirical and is set to 0.2 in the experiments.

6. Robust tracking under PF

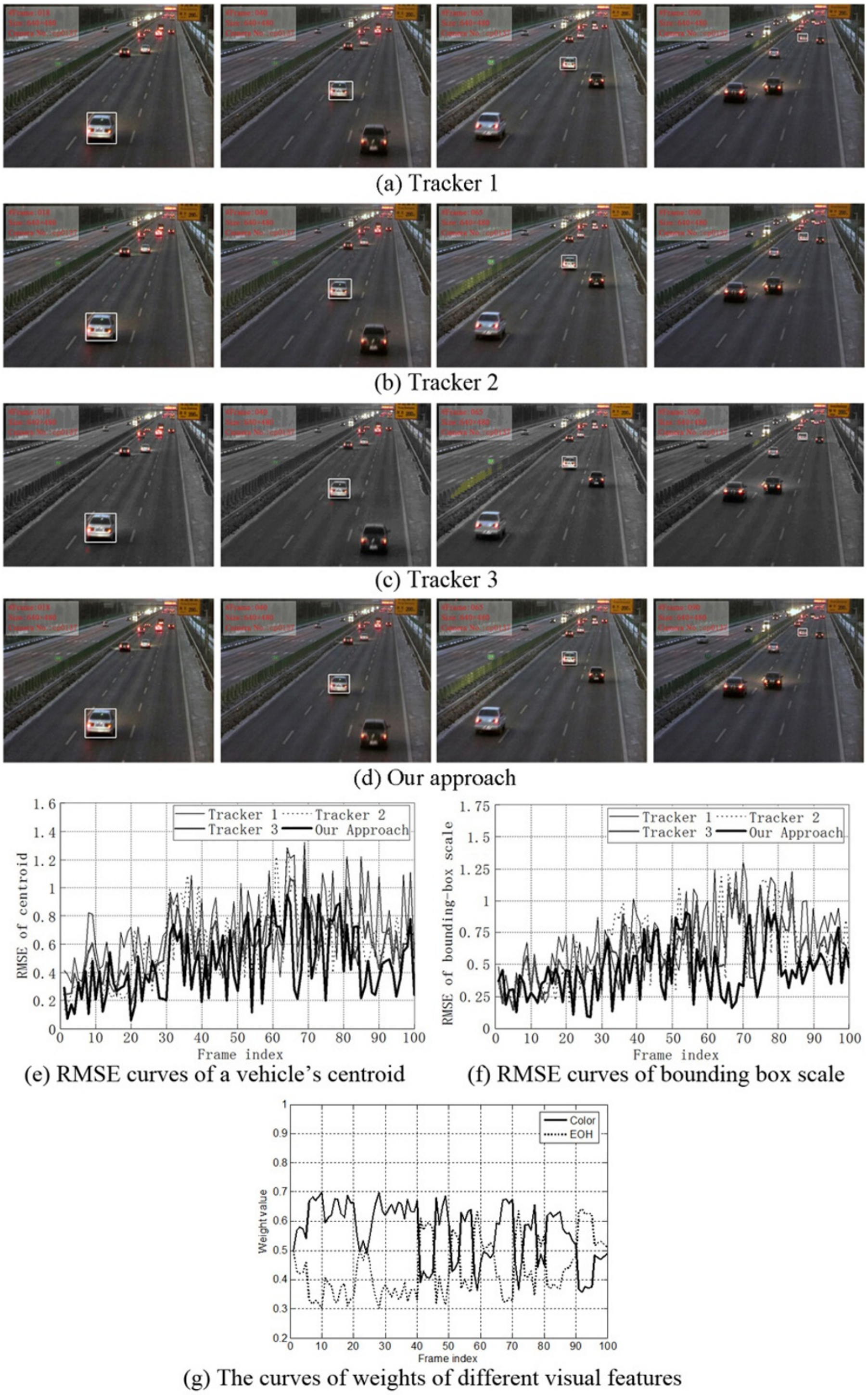
According to the state and observation model, multi-vehicle tracking is performed by running multiple-independent PFs for every vehicle in the scene. Algorithm 1 summarizes the fully automatic multi-vehicle tracking algorithm.

Algorithm 1. Robust Tracking under PF

```
Input: $\{I_t\}_{t = 1, ..., T}$;
  Output: $\{\hat{\mathbf{S}}_t(m)\}_{t=1,...,T;m=1,...,M}$;
  1. Detect the ROI of vehicle;
  2. Divide (0, 1] into N independent intervals, and N is the number of initial particles, i.e.
     $(0, 1] = \left(0, \frac{1}{N}\right] \cup \cdots \cup \left(\frac{N-1}{N}, 1\right]$, where N is the number of initial particles;
  3. For each initial particle set $\{S^i\}_{i=1,2,...,N}$, which is independent identical distribution, $S^i$ is denoted as
     $\mathbf{S}^i = U\left(\left(\frac{i-1}{N}, \frac{i}{N}\right]\right)$, where $U((u, v])$ is uniform distribution in $(u, v]$;
  4. The vehicle is fragmented according to the set of features generated by the EOH;
  5. Compute the initial HSV color histogram of each fragments of vehicle;
  6. Compute the initial EOH histogram of each fragments of vehicle;
  7. Initialize the weights of integration of the color and EOH features: $\alpha^c = \alpha^e = 0.5$;
  8. For t = 1,2,...
     For i = 1,...,N
        Predict the state of the vehicle by Equation (5):
        $\bar{\mathbf{S}}_t^i = E(\mathbf{S}_t^i) = 2\mathbf{S}_{t-1}^i - \mathbf{S}_{t-2}^i$;
        Compute the observation likelihood of color $p(\mathbf{O}_t^c|\mathbf{S}_t^i)$ by Equation (27)
        Compute the observation likelihood of EOH $p(\mathbf{O}_t^e|\mathbf{S}_t^i)$ by Equation (28)
        Generate the observation likelihood integrating both color and EOH $p(\mathbf{O}_t|\mathbf{S}_t^i)$ by Equation (29);
        Update the importance weights: $\omega_t^i = \omega_{t-1}^i p(\mathbf{O}_t|\mathbf{S}_t^i)$;
     End For
  9. If it is necessary to do re-sampling
     Obtain a new set of particles: $\{\mathbf{S}_t^i, 1/N\} \sim \{\mathbf{S}_t^i, \omega_t^i\}$;
  End if
  10. Generate the final state vector by Eqn. (30);
  11. Compute the quality function $\gamma_t^f$ of color and EOH by Equation (33), respectively;
  12. Compute the integrated weight $\alpha_t^f$ of color and EOH by Equation (32), respectively;
  13. According to probability density distribution of the posterior of a vehicle's state, compute the two factors $\eta_1(t)$ and $\eta_2(t)$ by Equations (39) and (42), respectively;
  14. Obtain the forgetting factor $\beta_t$ by Equation (38) to update the vehicle's observation model by Equation (37);
  End For
```

7. Experimental results

In this section, the proposed approach is used to track vehicles on the road. In our experiments, the dataset is composed of video sequences which were obtained from a real surveillance camera. The camera is fixed on a pole in highway and has a high-angle shot to one side of a driveway. All the experiments were carried out on 640 × 480 pixel sequences with an Intel$^{®}$ Core™ Duo CPU T7500



(a) Tracker 1

(b) Tracker 2

(c) Tracker 3

(d) Our approach

(e) RMSE curves of a vehicle's centroid

(f) RMSE curves of bounding box scale

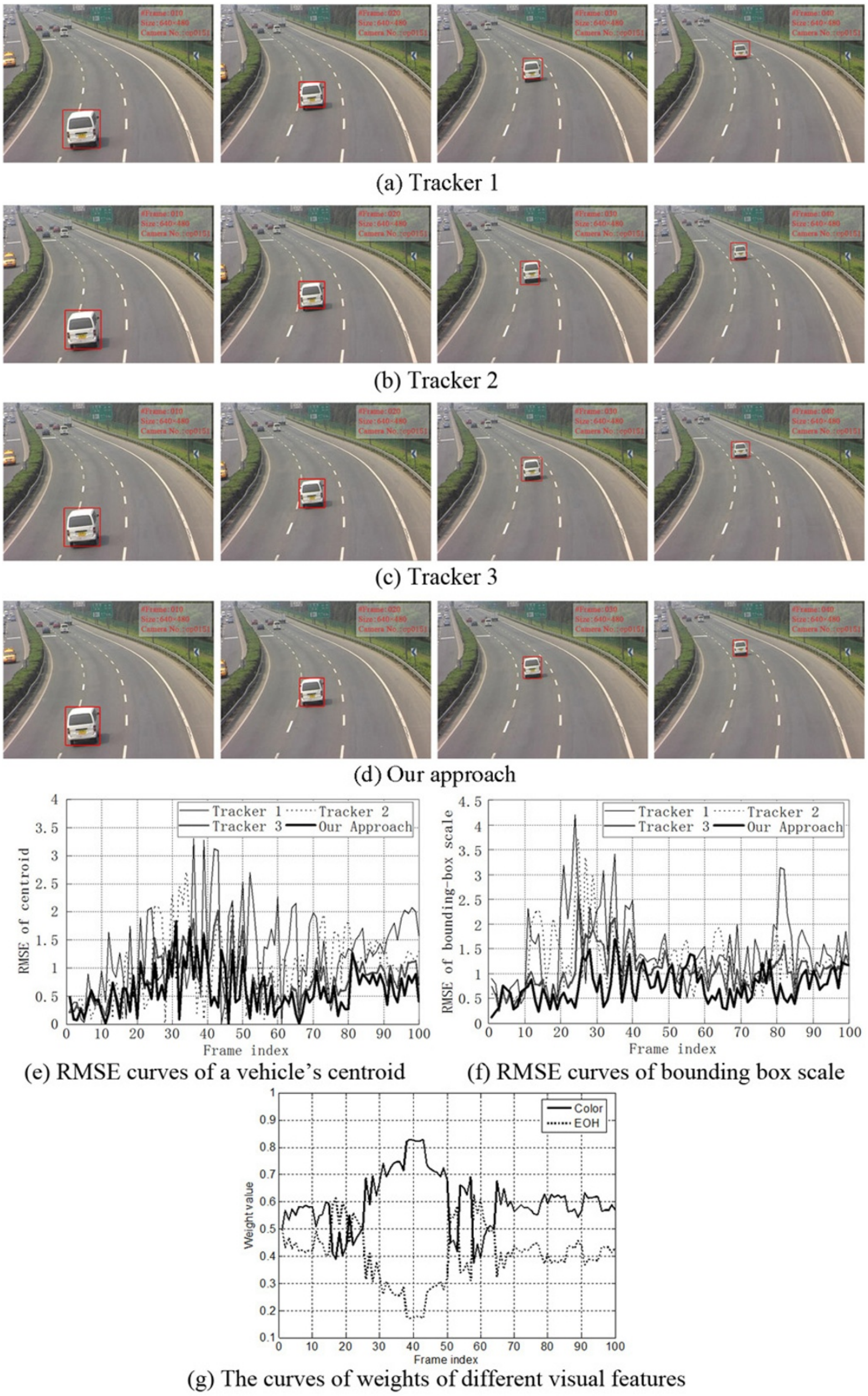
(g) The curves of weights of different visual features

Figure 7 Vehicle tracked when moving straight forward in the evening.

2.93 GHz PC. A real-life scenario, including partial occlusion, large-area occlusion in a short time and scale variation, is considered. We verify the performance of our approach via single and multiple vehicle target trackings. In the experiments, the length of each video sequence is 100 frames, and the number of particles is set to 50.

7.1. Quantitative evaluation

We evaluate our algorithm quantitatively in order to show the robustness for tracking. The evaluation compares the position and scale estimation of our approach with the ground-truth. Root mean squared error (RMSE) is used as the performance metric. The RMSE



(a) Tracker 1

(b) Tracker 2

(c) Tracker 3

(d) Our approach

(e) RMSE curves of a vehicle's centroid

(f) RMSE curves of bounding box scale

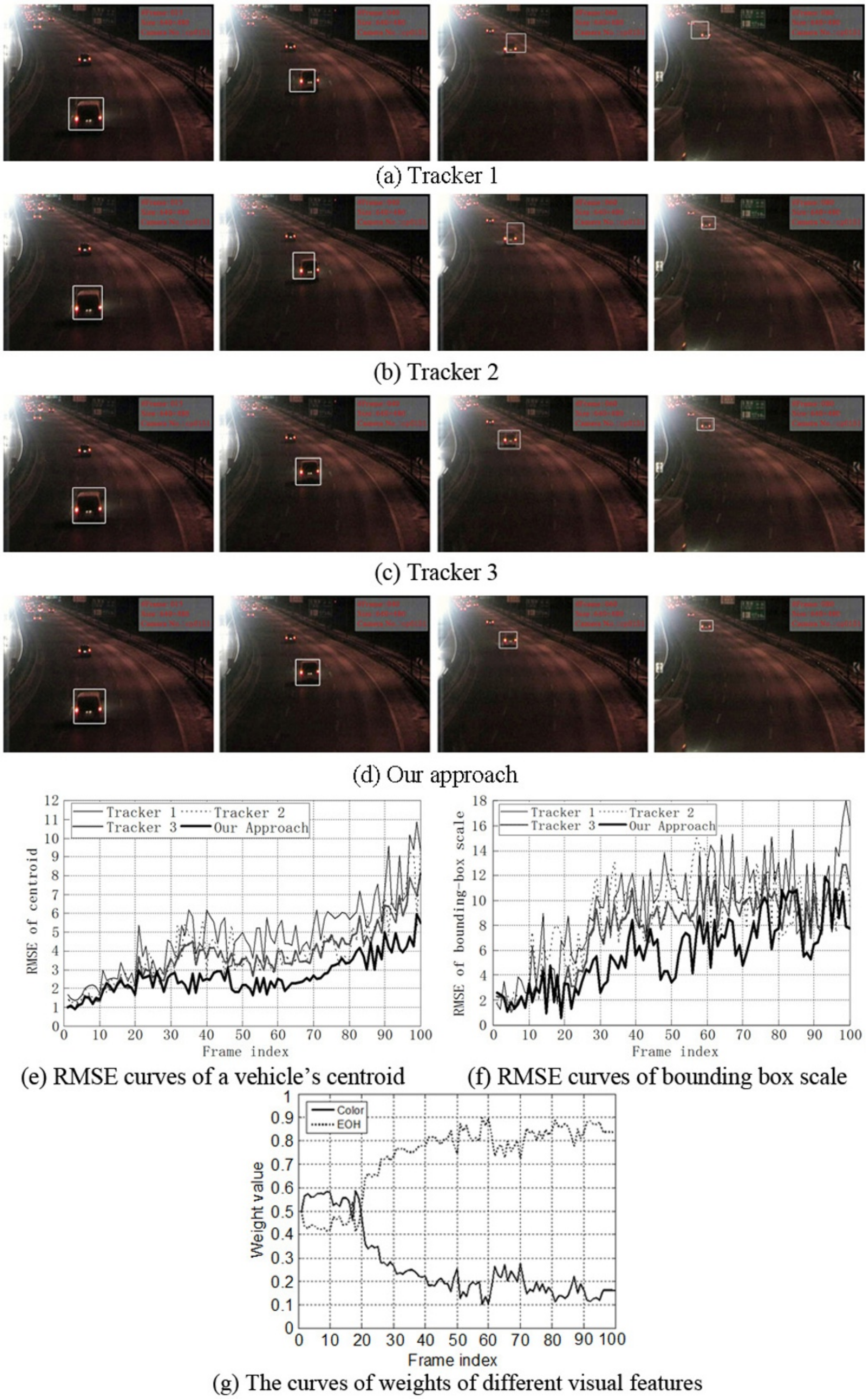
(g) The curves of weights of different visual features

Figure 8 Vehicle tracked during turning in daytime.

of a vehicle's centroid and bounding box scale are defined as follows:

$$\mathrm{RMSE}_{\mathrm{centroid}}(t) = \sqrt{\frac{1}{M}\sum_{m=1}^{M}\left[(x_{t,m} - x'_t)^2 + (y_{t,m} - y'_t)^2\right]} \quad (43)$$

$$\mathrm{RMSE}_{\mathrm{scale}}(t) = \sqrt{\frac{1}{M}\sum_{m=1}^{M}(s_{t,m} - s'_t)^2} \quad (44)$$



(a) Tracker 1

(b) Tracker 2

(c) Tracker 3

(d) Our approach

(e) RMSE curves of a vehicle's centroid

(f) RMSE curves of bounding box scale

(g) The curves of weights of different visual features

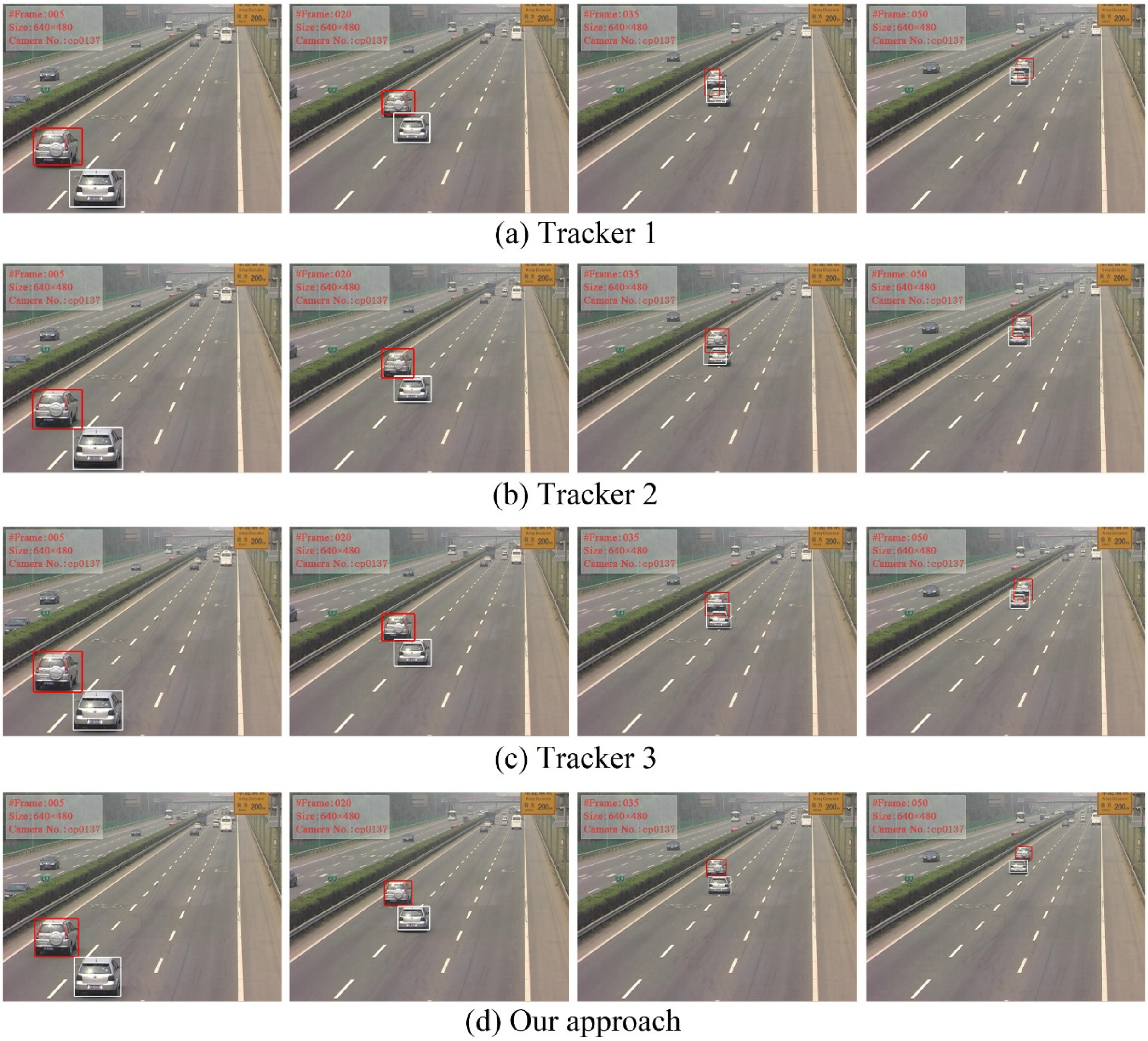
Figure 9 Vehicle tracked during turning at night.

where (x'_t, y'_t) and s'_t are the ground-truth centroid and scale of a vehicle at time t, respectively. M is the measurement time.

7.2. Tracking results and discussion

For comparison, we conducted our experiments with four different types of trackers: a color-based PF tracker (Tracker 1), an EOH-based PF tracker (Tracker 2), a PF tracker based on fixed-weight multiple visual features (Tracker 3), and our approach. The former three trackers had no adaptive updating during tracking, and the weights of color histogram and EOH features are 0.5 in Tracker 3, respectively.

In Figure 7, the vehicle traveled on a straight driveway. The tracking results of the four trackers are shown in Figure 7a-d. From Figure 7e, f, we can see that the RMSEs of a vehicle's centroid and bounding box scale of the four trackers are all maintained at a lower level, and the four trackers provide nearly similar tracking results. Figure 7g gives the curves of weights of different visual features. It can be seen that the features are in a relatively stable state with no dramatic change throughout the video sequence because of no obvious change of illumination or translation, rotation, etc. So, the tracking results of Track 3 and our approach are more similar to each other.



(a) Tracker 1

(b) Tracker 2

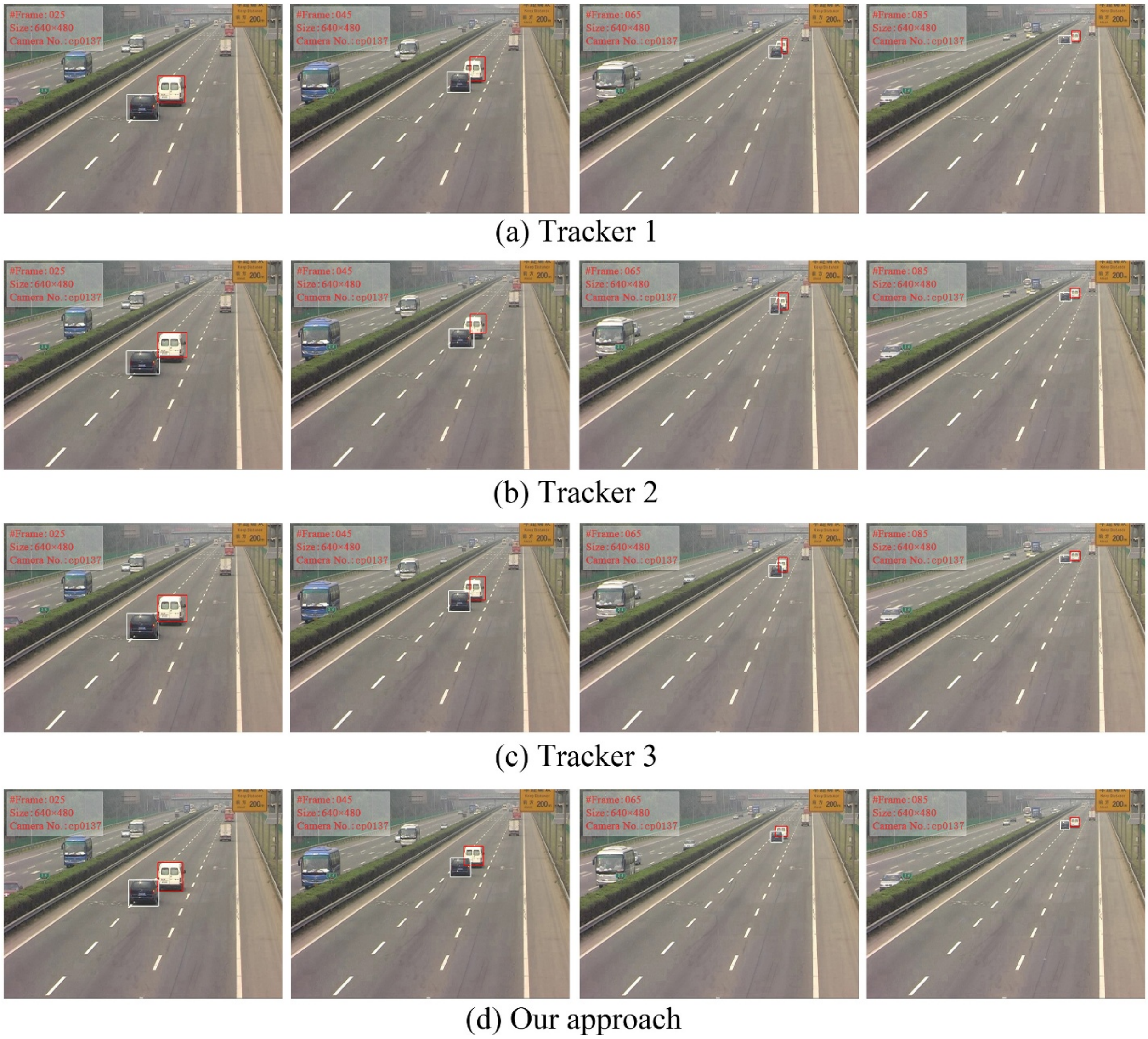
(c) Tracker 3

(d) Our approach

Figure 10 Two similarly colored vehicles when slight occlusion occurs in daytime.

Figure 8 shows the vehicle turning. Figure 8a-d presents the tracking results of the four trackers and we can see from Figure 8e, f that the RMSE of our approach is lower than that of the other three trackers, especially obvious when the vehicle turns between frames 30 and 40. In Figure 8g, due to translation and rotation, the EOH of the vehicle has a greater change. The decline of the EOH weight makes the color histogram more reliable. From the comparison of the RMSEs of Tracker 3 and our approach in Figure 8e, f, we can see that the RMSEs of a vehicle's centroid are provided more similar tracking results than the bounding box scale because of the advantage of multiple visual features integration for vehicles.

The third sequence is captured at night, as shown in Figure 9. The color of the vehicle and the background are very similar, but the edge feature is obvious due to the streetlight and the headlight of the rear vehicle. Figure 9a-d is still the tracking results of four trackers. From Figure 9e, f, we can see that the RMSEs of all trackers increase significantly, but our approach is still more accurate than the other methods. Furthermore, the accuracy of Tracker 3 decreases much faster than our approach, caused by the fixed weights of multiple



(a) Tracker 1

(b) Tracker 2

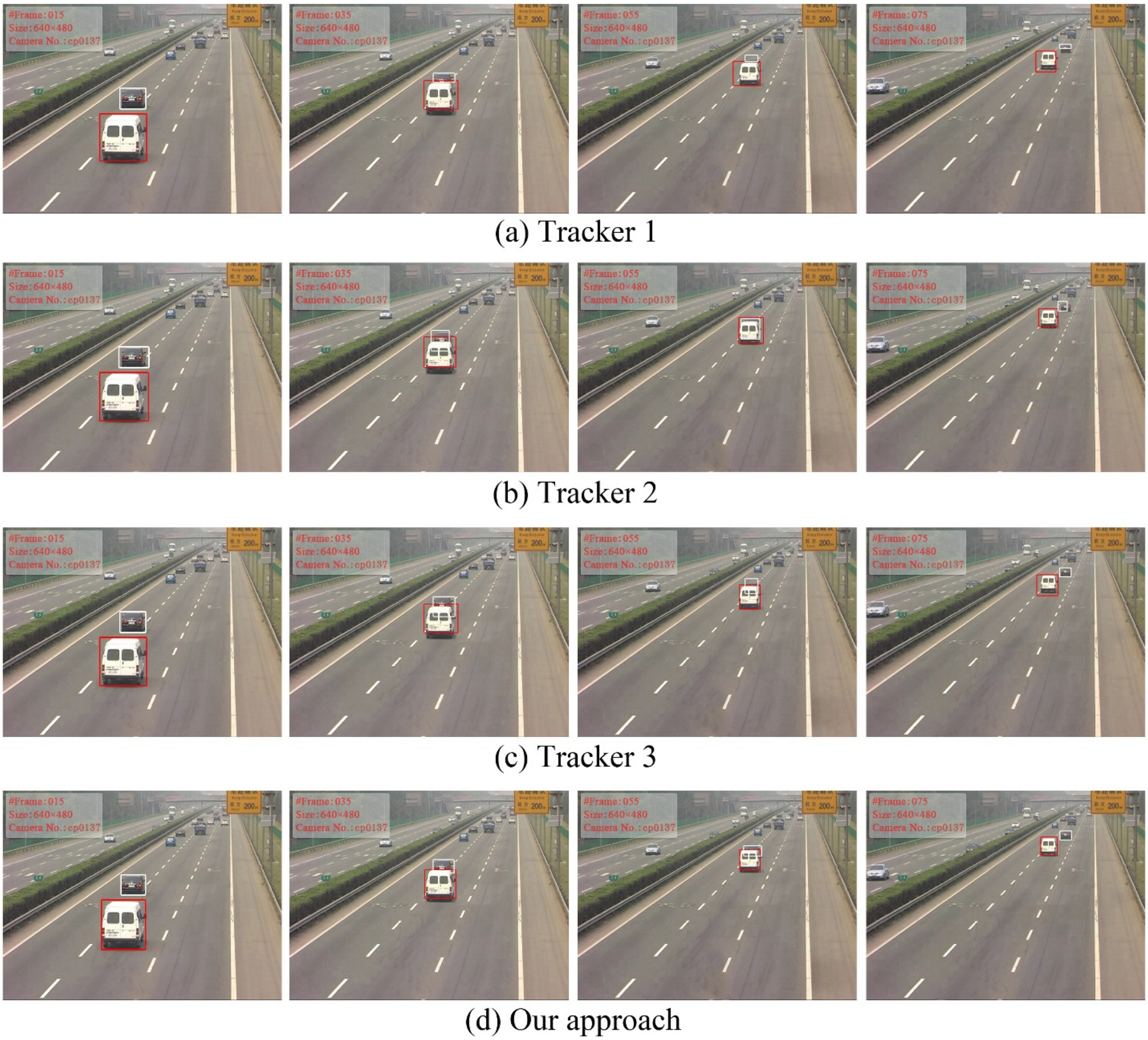
(c) Tracker 3

(d) Our approach

Figure 11 Vehicles tracked when partial occlusion occurs in daytime.

visual features. In Figure 9g, the changes of weight show that the color histogram is unreliable with low weight because the color of the vehicle shows weak discrimination from the background. Instead, the EOH feature plays a dominant role at this moment.

When vehicles with very similar colors appear close to each other, tracking algorithms using color, EOH, or fixed-weight multiple visual features fail. As shown in Figure 10, the precision of tracking with fixed-weight multiple visual features is slightly better than color-based tracking, but the deviation caused by color similarity is just prolonged and cannot be prevented completely. And the adaptive integration of multiple features makes a contribution to distinguish the vehicles to get a robust tracking result. Partial occlusion between vehicles, even large area occlusion, is the key issue to influence robustness and tracking accuracy. Figures 11, 12, 13, and 14 show the cases of occurrence of occlusion. As illustrated in Figures 11 and 12, partial and large area occlusion appear and last for about 50 frames, respectively. While the proposed approach incorporates spatial information, i.e., fragmentation, the proposed approach can track vehicles with non-occluded fragments. Since some fragments were occluded, the observation model of these regions is unreliable and the forgetting factor is determined to



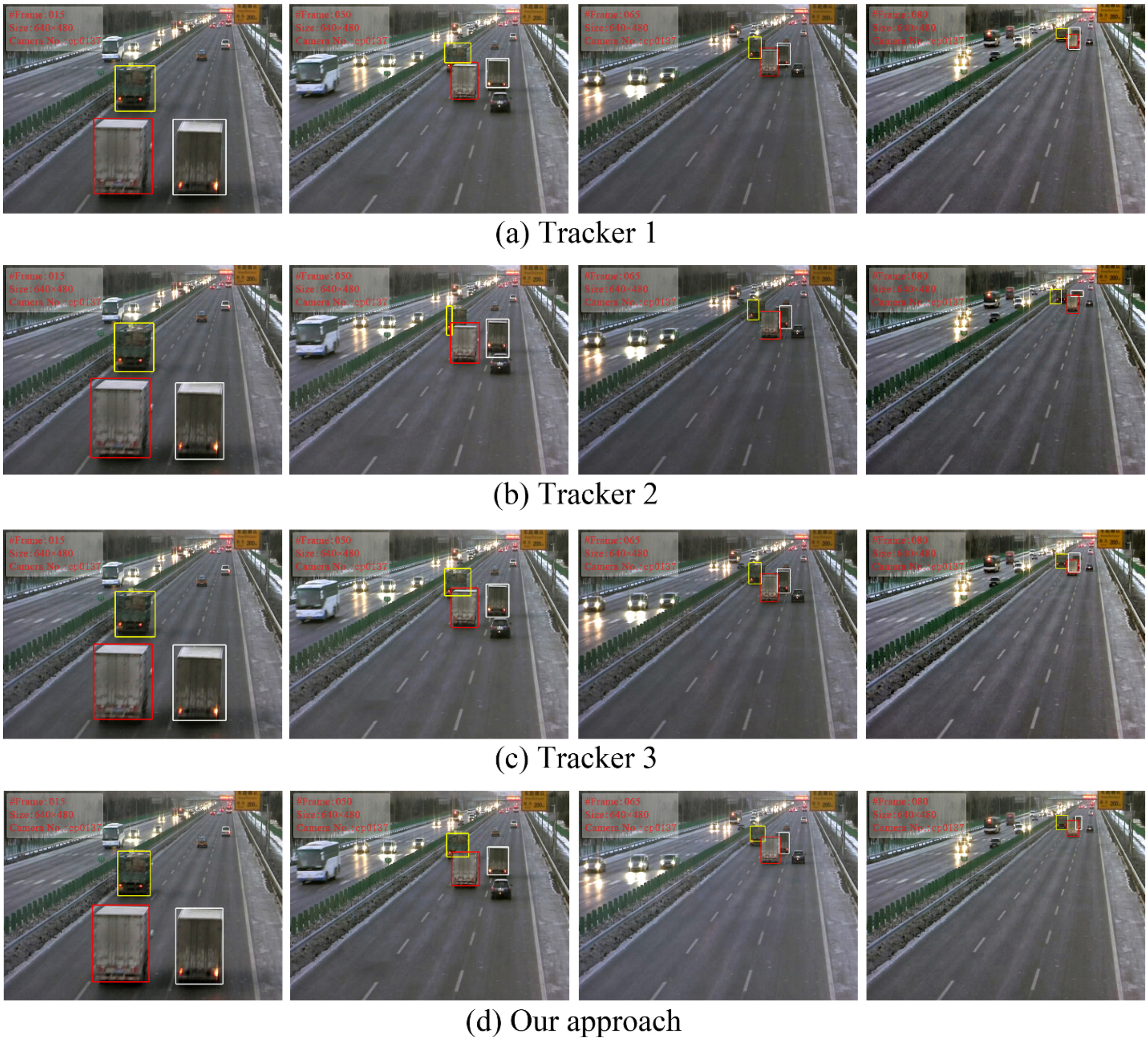
(a) Tracker 1

(b) Tracker 2

(c) Tracker 3

(d) Our approach

Figure 12 Vehicles tracked when large-area occlusion occurs in daytime.



(a) Tracker 1

(b) Tracker 2

(c) Tracker 3

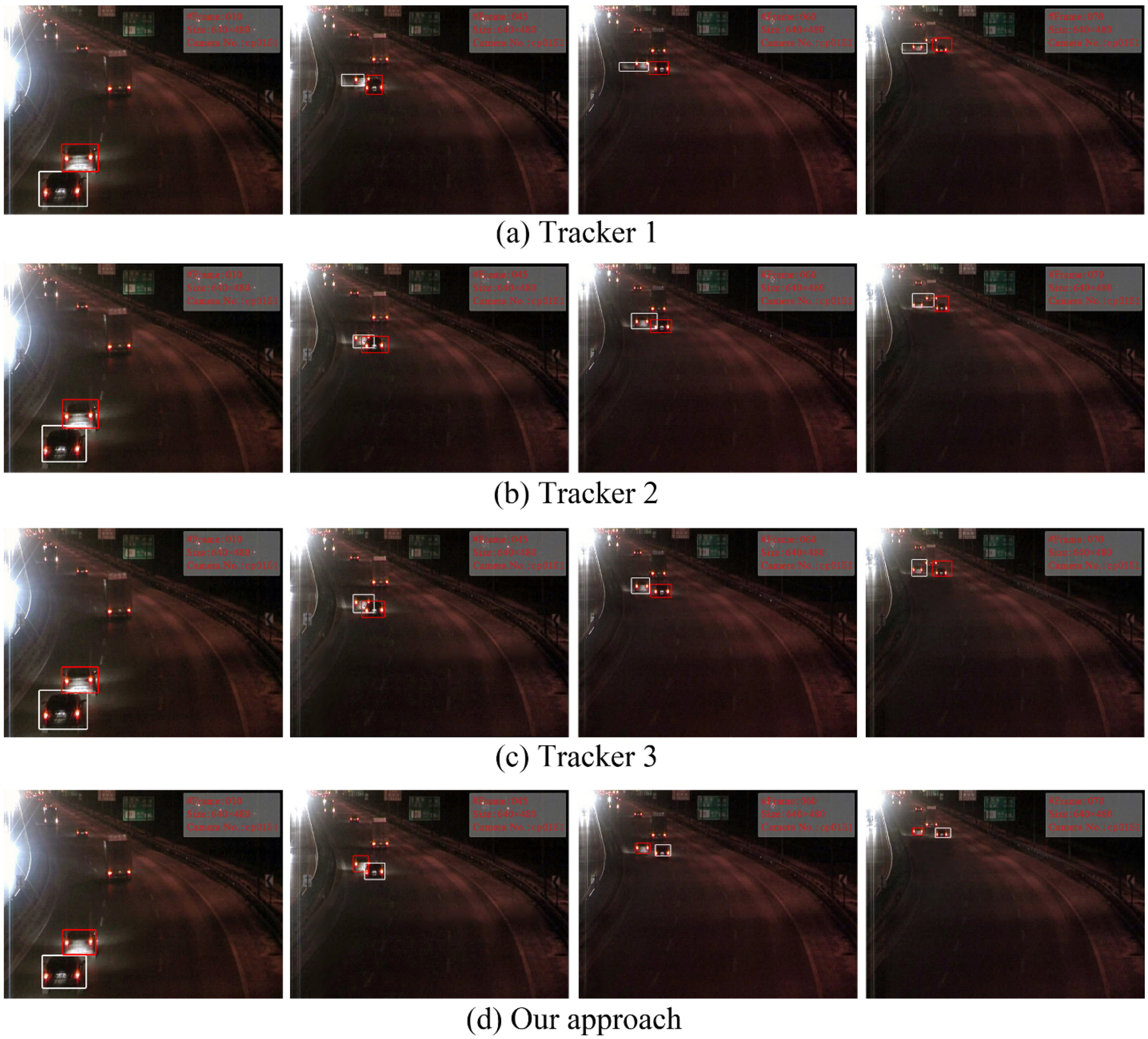
(d) Our approach

Figure 13 More vehicles tracked under occlusion in the evening.

be equal to 0. Therefore, the observation model is stopped from updating. When occlusion is finished, the proposed tracker can still give continuous tracking. Figure 14 shows vehicles traveling under occlusion at night. As Figure 14a-c demonstrates, the former three trackers may lead to inaccurate results due to the ambiguities inherent in the processing of the video sequence when considering single modalities. There are objects in the background which have a similar appearance to the vehicle. Therefore, soon after the initialization, the color-based tracking framework starts on the vehicle and gradually deviates from the ground-truth.

8. Conclusions

This article presents a robust tracking approach for multiple vehicles using adaptive integration of multiple visual features. Color histograms and EOHs are selected as visual features to model the observation of vehicles and integrated by a democratic integration strategy, and the observation model is embedded in a PF tracking framework. The spatial information is incorporated into the observation model to improve the robustness of object representation by dividing the object to be tracked into a number of fragments. Further, in order to avoid errors caused by model drift, the updating process should only be implemented in a reliable manner, and


(a) Tracker 1

(b) Tracker 2

(c) Tracker 3

(d) Our approach

Figure 14 Vehicles tracked by our approach at night.

the rate of updating can be controlled according to this reliability. The posterior probability density function of distribution of state vector and similarity between the candidate and reference observation of an object are used to define the valid measurement of reliability to model updating during tracking. Experimental results in real traffic surveillance video sequences show that our approach outperforms others in vehicle tracking under complex conditions.

Acknowledgements
This study was supported by the optional research topic from the National Natural Science Foundation of China (No. 61103094). Furthermore, it was also supported by the National High Technology Research and Development Program (863) with the research topic ID 2011AA010502.

Author details
[1]School of Instrument Science and Opto-electro Engineering, Beihang University, Beijing 100191, People's Republic of China [2]Research Institute of Beihang University in Shenzhen, Shenzhen 518057, People's Republic of China [3]School of Computer Science, Beihang University, Beijing 100191, People's Republic of China

Competing interests
The authors declare that they have no competing interests.

Received: 19 August 2011 Accepted: 24 February 2012
Published: 24 February 2012

References

1. Ruiter H, Benhabib B: **Tracking of rigid-bodies for autonomous surveillance.** In *Proceedings of IEEE International Conference on Mechatronics and Automation. Volume 2.* Niagara Falls, Canada; 2005:928-933.

2. Chen YQ, Rui Y, Huang TS: **JPDAF based HMM for real-time contour tracking.** In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Volume 1.* Hawaii, USA; 2001:543-550.
3. Li PH, Zhang TW, Arthur ECP: **Visual contour tracking based on particle filters.** *Image Vis Comput* 2003, **21**:111-123.
4. Czyz J, Ristic B, Macq B: **A particle filter for joint detection and tracking of color objects.** *J Image Vis Comput* 2007, **25**:1271-1281.
5. Zhai Y, Yeary MB, Cheng S, Kehtarnavaz N: **An object-tracking algorithm based on multiple-model particle filtering with state partitioning.** *IEEE Trans Instrum Meas* 2009, **58**:1797-1809.
6. Kazuhiro H: **Adaptive weighting of local classifiers by particle filters for robust tracking.** *Pattern Recogn* 2009, **42**:619-628.
7. Cui P, Sun L, Yang S: **Adaptive mixture observation models for multiple object tracking.** *Sci China Ser F: Inf Sci* 2009, **52**:226-235.
8. Comaniciu D, Ramesh V, Meer P: **Kernel-based object tracking.** *IEEE Trans Pattern Anal Mach Intell* 2003, **25**:564-577.
9. Jonathan D, Ian R: **Articulated body motion capture by stochastic search.** *Int J Comput Vis* 2005, **61**:185-205.
10. Tu Q, Xu YP, Zhou ML: **Robust vehicle tracking based on scale in-variant feature transform.** *Proceedings of IEEE International Conference on Information and Automation* Changsha, China; 2008, 86-90.
11. Wei Q, Xiong Z, Li C: **Color spatial feature based approach for multiple-vehicle tracking.** *Appl Opt* 2010, **49(31)**:6034-6047.
12. Jahangheer SS, Khan MI: **Detection and tracking of rotated and scaled targets by use of hilbert-wavelet transform.** *Appl Opt* 2003, **42(23)**:4718-4735.
13. Birchfield S: **Elliptical head tracking using intensity gradients and color histograms.** *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition* Santa Barnara, CA, USA; 1998, 232-237.
14. Tao X, Christian D: **Monte Carlo visual tracking using color histograms and a spatially weighted oriented Hausdorff measure.** In *Proceedings of the Conference on Analysis of Images and Patterns. Volume 2756.* Groningen, Netherlands; 2003:190-197.
15. Kwolek B: **Stereovision-based head tracking using color and ellipse fitting in a particle filter.** In *Proceedings of the 8th European Conference on Computer Vision. Volume 3023.* Prague, Czech Republic; 2004:192-204.
16. Spengler M, Schiele B: **Towards robust multi-cue integration for visual tracking.** *Mach Vis Appl* 2003, **14**:50-58.
17. Jepson AD, Fleet DJ, El-Maraghi TF: **Robust online appearance models for visual tracking.** *IEEE Trans Pattern Anal Mach Intell* 2003, **25(10)**:415-522.
18. Toyama K, Blake A: **Probabilistic tracking with exemplars in a metric space.** *Int J Comput Vis* 2002, **48(1)**:9-19.
19. Yang M, Wu Y: **Tracking non-stationary appearances and dynamic feature selection.** In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition. Volume 2.* San Diego, CA, USA; 2005:1059-1066.
20. Avidan S: **Ensemble tracking.** *IEEE Trans Pattern Anal Mach Intell* 2007, **29(2)**:261-271.
21. Triesch J, Malsburg C: **Self-organized integration of adaptive visual cues for face tracking.** *Proceedings of IEEE International Conference on Automatic Face Gesture Recognition* Grenoble, France; 2000, 102-107.
22. Sheng H, Xiong Z, Weng JN, Wei Q: **An approach to detecting abnormal vehicle events in complex factors over highway surveillance video.** *Sci China Ser E: Technol Sci* 2008, **51**:199-208.
23. Sheng H, Li C, Wei Q, Xiong Z: **Real-time detection of abnormal vehicle events with multi-feature over Highway Surveillance Video.** *Proceedings of IEEE International Conference on Intelligent Transportation System* Beijing, China; 2008, 550-556.
24. Duan Z, Cai Z, Yu J: **Adaptive particle filter for unknown fault detection of wheeled mobile robots.** *Proceedings of IEEE International Conference on Intelligent Robots and Systems* 2006, 1312-1315.
25. Levi K, Weiss Y: **Learning object detection from a small number of examples: the importance of good features.** *Comput Vis Pattern Recogn* 2004, **2**:53-60.
26. Adam A, Rivlin E, Shimshoni I: **Robust fragments-based tracking using the integral histogram.** In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition. Volume 1.* New York, USA; 2006:798-805.
27. Maggio E, Cavallaro A: **Multi-part target representation for color tracking.** In *Proceedings of IEEE International Conference on Image Processing. Volume 1.* Genoa, Italy; 2005:729-732.
28. Nummiaro K, Koller-Meier E, Gool LJV: **An adaptive color-based particle filter.** *Image Vis Comput* 2003, **21**:99-110.
29. Choeychuen K, Kumhoma P, Chamnongthaia K: **Robust ambiguous target handling for visual object tracking.** *AEU Int J Electron Commun* 2010, **64(10)**:960-970.
30. Moreno-Noguer F, Sanfeliu A: **A framework to integrate particle filters for robust tracking in non-stationary environments.** *Pattern Recogn Image Anal* 2005, **3522**:93-101.
31. Liu H, Yu Z, Zha HB, Zou YX, Zhang L: **Robust human tracking based on multi-cue integration and mean-shift.** *Pattern Recogn Lett* 2009, **30(9)**:827-837.
32. McKenna S, Jabri S, Doric Z, Wechsler H, Rosenfeld A: **Tracking groups of people.** *Comput Vis Image Understand* 2000, **80**:42-56.

doi:10.1186/1687-5281-2012-2
Cite this article as: Sheng *et al.*: Robust multiple-vehicle tracking via adaptive integration of multiple visual features. *EURASIP Journal on Image and Video Processing* 2012 **2012**:2.