Computer Vision and Image Understanding 115 (2011) 69–80

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Combined feature evaluation for adaptive visual object tracking

Zhenjun Han, Qixiang Ye, Jianbin Jiao *

Graduate University of Chinese Academy of Sciences, 100049 Beijing, PR China

ARTICLE INFO

Article history:
Received 14 October 2009
Accepted 19 September 2010
Available online 8 October 2010

Keywords:
Object tracking
Color histogram
Gradient orientation histogram
Kalman filter
Particle filter

ABSTRACT

Existing visual tracking methods are challenged by object and background appearance variations, which often occur in a long duration tracking. In this paper, we propose a combined feature evaluation approach in filter frameworks for adaptive object tracking. First, a feature set is constructed by combining color histogram (HC) and gradient orientation histogram (HOG), which gives a representation of both color and contour. Then, to adapt to the appearance changes of the object and its background, these features are assigned with different confidences adaptively to make the features with higher discriminative ability play more important roles in the instantaneous tracking. To keep the temporal consistency, the feature confidences are evaluated based on Kalman and Particle filters. Experiments and comparisons demonstrate that object tracking with evaluated features have good performance even when objects go across complex backgrounds.

1. Introduction

Object tracking is to automatically find the same object in adjacent frames from a video sequence after the object's location is initialized. It plays an important role in many video applications, such as automatic visual surveillance [1], human computer interaction systems [2] and robotics [3].

The difficulties of object tracking are caused by the motion state variation of the object, the appearance variation of the object and the background, occlusions, etc. Up to now, researchers have fallen into three different categories to deal with these difficulties, namely motion models, searching methods and object representations [4–10].

Motion models are employed to predict the object's location in a new frame within a video sequence based on its history motion characteristics. This can improve the tracking stabilization and make the tracking survive some occlusions [12–14]. On the other hand, searching methods are also indispensable to a successful tracking. Given a tracking object, searching methods use various matching strategies to find its location in a new video frame. In addition, when the object varies on size, it is needed to calculate the scale parameter. Mean Shift algorithm is the most representative searching method [15].

Although motion models and searching algorithms are crucial to object tracking, it is not true that a proper motion model together with a good searching algorithm will always lead to good tracking results. The most important issue in object tracking is whether the object representation is effective enough to discriminate the object with its background during all the tracking process. Therefore, in this paper, we cast the object tracking as finding an adaptive feature representation for the whole tracking process.

Color is one of the most widely used feature representations [16–19] for its effectiveness and efficiency. Discarding color spatial distribution, color histogram (HC) is robust to small object deformation, scale variation and some rotation, which ensures the HC feature representation succeed in many tracking conditions, especially when the appearances of both the object and background are stable. However, it is obvious that HC features cannot work well when the object has the similar color to its background or the object has appearance variation caused by illumination changes. In addition, other characteristics, such as texture and contour, are employed to represent the object [20,21]. Recently, histogram of oriented gradient (HOG) [22,23] was widely applied for object detection and tracking. HOG captures the edges or gradient structures, which are the characteristics of local contour and shape, and therefore it is insensitive to color variation. In [22], Dalal et al. justified that the representation ability of HOG is almost as capable as scale invariant feature transformation (SIFT) [25] descriptors given a fixed scale. However, similar to other contour and texture features, a disadvantage of HOG is that it cannot effectively represent objects or backgrounds with large smooth regions since the contours of them are indistinctive. Another disadvantage of HOG is that it is orientation sensitive, which means that when the object rotates the previous representation will be invalid.

The mutual complementarities of the two types of features inspired us to integrate them together. Then the central issue in object tracking will become which features are important and informative for tracking. Our insight is that the features which

* Corresponding author. Address: No. 19A, Yu Quan Road, Shi Jing Shan District, 100049 Beijing, PR China.
E-mail address: jiaojb@gucas.ac.cn (J. Jiao).

1077-3142/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.cviu.2010.09.004

can distinguish the object from the background better are more important and should be assigned larger confidences.

There have been enormous efforts on finding the "optimal" features. Collins et al. [19] developed an online feature selection method. They noted susceptibility of the variance ratio feature selection method to distraction by spatially correlated background clutter and developed an additional approach that seeks to minimize the likelihood of distraction. And they defined the discriminative ability of a feature according to the two-class variance ratio measures of the object and its background. Liang et al. [17] proposed a similar approach in which the discriminative ability of a feature is calculated based on Bayes Error Rate between the object and its background. Bayes Error Rate of a feature is calculated by the intersection of the likelihood function of the object and its background on the feature. Wang et al. [20] proposed a method to online evaluate a subset of Haar-like features by Adaboost learning. Chen et al. [18] used a hierarchical Monte Carlo algorithm to learn region confidences for object tracking. Despite the advantages of these approaches, a main drawback is that most of them select or evaluate the features separately in a video frame and seldom consider the temporal consistency of the process, which will decrease the tracking stabilization.

In literature [21], Wang et al. also used online feature evaluation of combined feature set for object tracking, while temporal consistency of the evaluation and features' rotation sensitivity are not considered. In this paper, these two problems are fully investigated. And the contributions are summarized as follows.

1.1. A combined feature set for adaptive object tracking

The combined feature set is the integration of colors, edge orientations, local contours, and SIFT descriptors. What is more, we propose an approach to reduce the orientation sensitivity of the HOG features by calculating the dominant orientation of the object, which improves the effectiveness of the proposed combined feature set.

1.2. A new feature evaluation approach considering temporal consistency based on tracking filters

Traditional filter algorithms are generally used to model object motion states in a visual tracking, while we apply them to assign feature confidences, which ensures that the evolution of feature confidence is temporal consistency by exploiting the feature discriminative abilities in the current frame and feature confidences in the previous frames. This is the main advantage of our approach compared with existing feature evaluation methods.

The rest of the paper is organized as follows. Overview of the method is presented in Section 2; details of feature evaluation using both Kalman and Particle filters are described in Section 3; experiments and comparisons of object tracking are given in Section 4, and conclusions in Section 5.

2. Overview of the proposed approach

In our approach, two kinds of features (HC and HOG) are firstly extracted to construct a combined feature set frame by frame. Then the feature evaluation is carried out to calculate the feature confidences, on which the tracking is completed finally.

2.1. Feature extraction and discriminative ability calculation

It is necessary to define object and background regions in which tracking features will be firstly extracted and then evaluated.

2.1.1. Object and background definition

An object is a rectangle area with $h \times w$ pixels and its background is defined as the surrounding ring area as shown in Fig. 1. The size of the background area is set as $\sqrt{2}h \times \sqrt{2}w - h \times w$ empirically. Object features $\{F^i(x,y)\}, i = 0, 1, \ldots, N$ are extracted from the object area whereas background features $\{B^i(x,y)\}, i = 0, 1, \ldots, N$ are from the pixels in background area, in which N is the feature dimension.

2.1.2. Discriminative ability calculation

We follow the idea of Collins et al. [19] to measure the discriminative ability of each feature in current frame by computing the log likelihood ratio between the object feature and its corresponding background feature as follows:

$$\widetilde{S}_t^i = \max\left(0, \min\left(1, \log\frac{\max(F_t^i(x,y),\delta)}{\max(B_t^i(x,y),\delta)}\right)\right), \quad i = 1 \ldots N \tag{1}$$

$$S_t^i = \frac{\widetilde{S}_t^i}{\sum_{i=1}^{N}\widetilde{S}_t^i} \quad i = 1 \ldots N \tag{2}$$

where $F_t^i(x,y)$ and $B_t^i(x,y)$ are the ith features of the object and the background at frame t respectively. δ is used to avoid dividing by zero or taking the log of zero, and is set as 0.005 empirically. Intuitively, $\log\frac{\max(F_t^i(x,y),\delta)}{\max(B_t^i(x,y),\delta)}$ takes positive values for features distinctive to the object, and negative for features associated with the background. The more distinctive to the object a feature is, the larger $\widetilde{S}_t^i$ is. Therefore, $\widetilde{S}_t^i$ represents a feature's ability of discriminating the object from its background. Function max() and min() are used to keep that $\widetilde{S}_t^i$ fells into (0.0, 1.0). Eq. (2) is used to normalize the discriminative ability.

2.2. Feature confidence calculation and object tracking

Using Eqs. (1) and (2), we can calculate feature discriminative ability in a single frame. However, it is not proper to directly harness it as feature confidence since it can be affected by noise

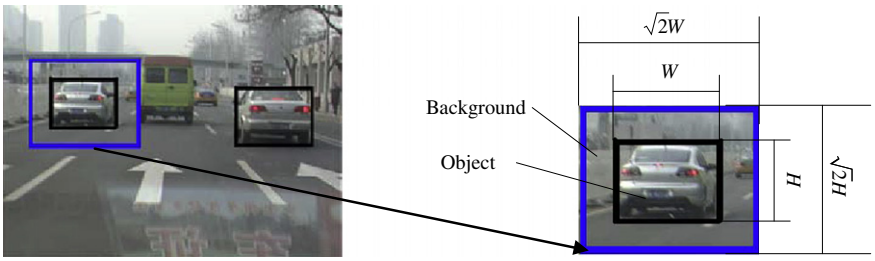

Fig. 1. Object and its background regions.

and the feature's instability in a long tracking process. Following the idea of traditional motion models, feature evaluation can be also formulated with filter frameworks.

2.2.1. Feature confidence calculation

We assume a first order Markov model for the feature evaluation, in which $w_t(i)$, the ith feature confidence in frame t, depends

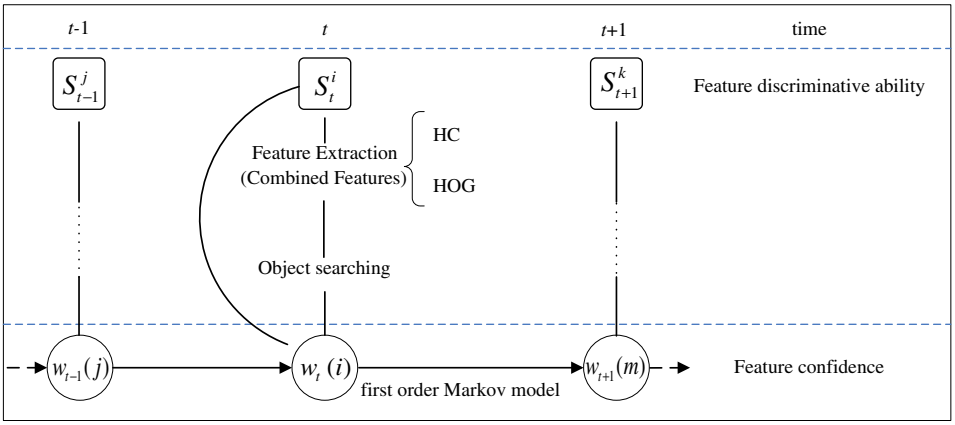

Fig. 2. Flow chart of the proposed approach.

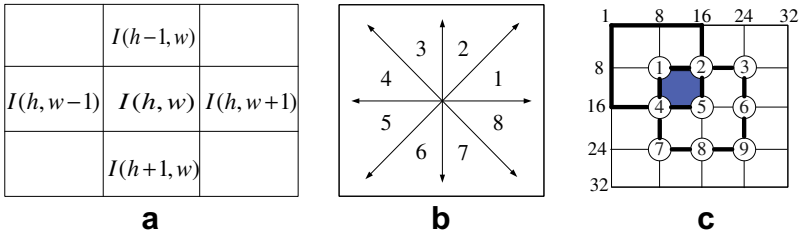

Fig. 3. HOG feature extraction. (a) Mask for pixel gradient calculation, (b) orientation bins, and (c) nine blocks for HOG feature extraction.

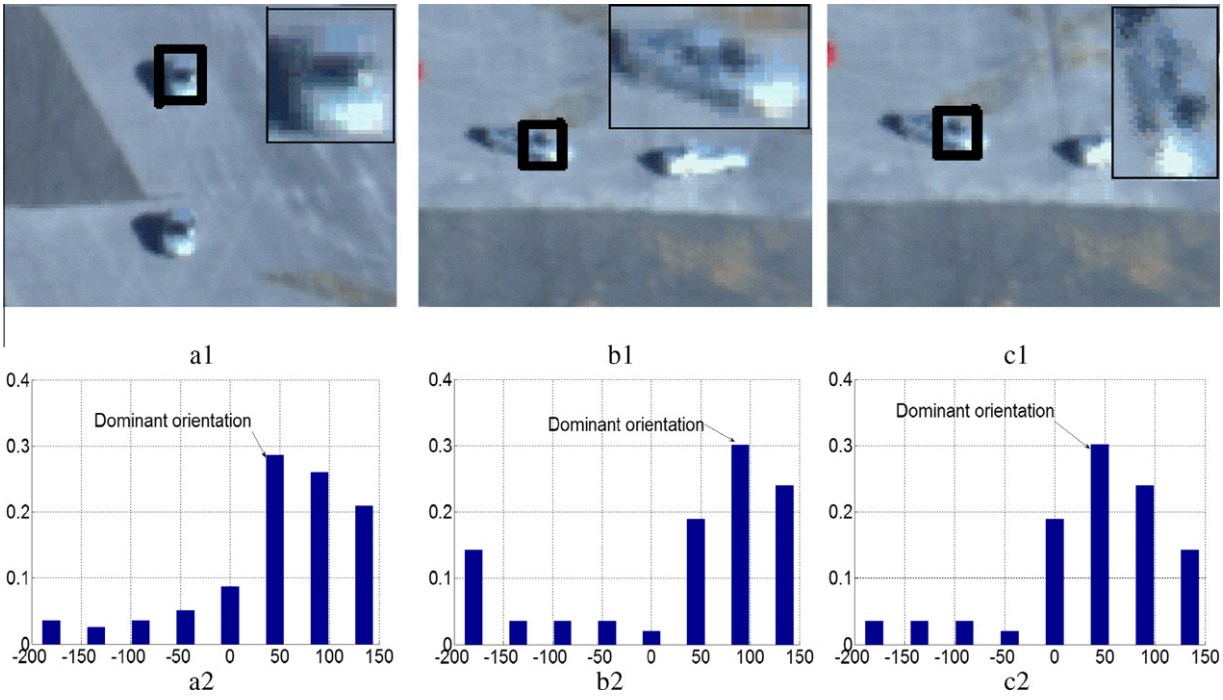

Fig. 4. Object rotation in HOG features extraction procedure. (a1) Object in the first video frame, (a2) the coarse orientation histogram of (a1) and the reference dominant orientation, (b1) candidate object in the tth video frame, (b2) the coarse orientation histogram of (b1) and its instantaneous dominant orientation, (c1) rotated candidate object in the tth video frame, (c2) the orientation histogram of (c1) and its normalized dominant orientation.

not only on the discriminative ability of the current frame S_t^i, but also the jth feature confidence $w_{t-1}(j)$ in frame $t-1$.

$$w_t(i) = f_{t,t-1}(w_{t-1}(j), S_t^i) + u_t \tag{3}$$

where $f_{t,t-1}$ represents a filter process, such as Kalman filter, in which case i is equal to j, or Particle filter, in which case i is always not equal to j, etc. u_t is the Gaussian noise function. The implementation detail of this equation will be presented in Section 3. In each confidence updating iteration, we use Eq. (4) to normalize the feature confidences.

$$w_t(i) = \frac{w_t(i)}{\sum_{i=1}^{N} w_t(i)}, \quad i = 1 \cdots N \tag{4}$$

With the definitions above, the flow chart of the proposed approach can be illustrated as Fig. 2.

Table 1
Feature evaluation using Kalman filter.

1. *Initialization* ($t = 0$). Initiate the confidence of each feature $(w_0(i),\ \Delta w_0(i))^T = (\frac{1}{N}, 0)^T$
2. *Prediction* ($t > 0$). For each feature in the feature set
   2.1 Use the Kalman filter to predict the prior confidence of each feature
   2.2 Use the HOGC features with prior confidences to guide the object searching (by Eqs. (5) and (6)) in next frame
3. *Correction* ($t + 1 > 0$). After obtaining the best match location of the object in the next frame
   3.1 Extract the instantaneous HOGC features
   3.2 Calculate the discriminative ability of each feature
   3.3 Calculate the posterior confidence of each feature
4. $t = t + 1$; Go to step 2 or end the tracking loop

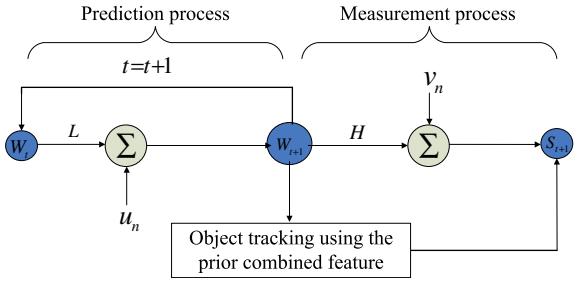

Fig. 5. Feature evaluation procedure using a Kalman filter.

2.2.2. Object tracking

The tracking procedure is performed with an exhaustive search algorithm in a candidate area Ω_t predicted by Kalman filter motion model in the new frame. Our goal is to get the best object location $(x,y)_t$ in Ω_t, solving $p((x,y)_t|\Omega_t,(x,y)_{t-1})$. We apply the Bayesian inference to obtain the tracking result as follows:

$$\begin{aligned}
&\max_{(x,y)_t} p((x,y)_t|\Omega_t,(x,y)_{t-1}) \\
&= \max_{(x,y)_t} \int p(F_t((x,y)_t^c)|(x,y)_t^c, F_{t-1}) p((x,y)_t^c|\Omega_t) dc \\
&= \max_{(x,y)_t} \int \left(\sum_{i=1}^{N} (p(F_t^i) p(F_t^i((x,y)_t^c), F_{t-1}^i)) \right) p((x,y)_t^c|\Omega_t) dc
\end{aligned} \tag{5}$$

where $(x,y)_t^c$ is the cth position in Ω_t, $p(F_t^i((x,y)_t^c), F_{t-1}^i)$ is the degree of similarity between the two features, $p(F_t^i)$ represents the degree of belief of the ith feature at frame t, which can be approximated by its confidence $w_t(i)$, therefore Eq. (5) can be rewritten as follows:

$$\begin{aligned}
&\max_{(x,y)_t} p((x,y)_t|\Omega_t,(x,y)_{t-1}) \\
&= \max_{(x,y)_t} \int \left(\sum_{i=1}^{N} (w_t(i) p(F_t^i((x,y)_t^c), F_{t-1}^i)) \right) p((x,y)_t^c|\Omega_t) dc
\end{aligned} \tag{6}$$

3. Details of combined feature evaluation

In this section, we present the details of feature extraction, and then feature evaluation in Kalman and Particle filters respectively. Kalman and Particle filters are chosen since they are the most representative filter algorithms used in object tracking.

3.1. Combined feature extraction

We use HC in RGB color space and HOG on gray image data to construct the combined feature set and name it histogram of combined HOG and HC (HOGC). These features are chosen because: (1) they can be computed efficiently, since the calculation of HC and HOG are simple statistics of color and gradient orientation occurrence probability; (2) they can represent an object effectively because of their mutually complementary analyzed in Section 1.

3.1.1. Color histogram

To calculate the color histogram, RGB color space, generally robust to rotation and deformation [26], is chosen for its simplicity. We first convert the color information of each pixel into a

Table 2
Feature evaluation using Particle filter.

1. Initiate the confidence $\{w_0(i)|i = 1,2,\ldots,N\}$ of each feature (particle) with equal value $(i, w_0(i))^T = (i, \frac{1}{N})^T$, where N is the number of features of the combined feature set
2. In frame t, construct the new corresponding sample set for the Particle filter as follows
   2.1 *Calculate* ($t > 0$) the cumulative probability for each particle by
$$\begin{cases} c_t(0) = 0 \\ c_t(i) = c_t(i-1) + w_t(i) \end{cases} \quad i = 1,2,\ldots,N$$
This induces $Set_t = \{(i, w_t(i), c_t(i)), \quad i = 1,2,\ldots,N\}$
   2.2 *Select* ($t > 0$) M particles (can be repeated) from Set_t (resampling); each particle i' is selected as follows
      (a) Generate a random number $r \in [0, 1.0]$, uniformly distributed
      (b) Find the smallest i for which $c_t(i) \geqslant r$
      (c) Set $i' = i$
   2.3 *Predict* ($t > 0$) the re-sampled particles $\{i'|i' = 1,2,\ldots,M\}$ by sampling from $p(k|i')$; in our approach, the transition probability $p(k|i')$ is defined as a normal distribution of feature as $p(k|i') \sim N(0,\delta)$, and δ is set as $2N$ in our experiments
   2.4 *Weight* ($t + 1 > 0$) $w_{t+1}(k)$ of each new predicted particle k by the measurement $p(S_{t+1}^k|k)$; in this paper, we use the feature discriminative ability (shown in Eq. (2)) as the measurement $p(S_{t+1}^k|k)$
   2.5 *Normalize* ($t + 1 > 0$) each feature by Eq. (4) and then obtain a new particle set $\{(k, w_{t+1}(k))\}, \quad k = 1,2,\ldots,N$
3. Object spatial location searching in the $t + 1$th frame by Eqs. (5) and (6)
4. $t = t + 1$; Go to step 2 or end the tracking loop

quantized value, and then the quantized value is mapped to an index of a corresponding histogram bin. The number of pixels assigned to each bin is accumulated over the whole image patch. In this paper, each color component (R, G and B) is linearly quantized into 16 levels and then a histogram of 16 dimensions is extracted on each component. We obtain a color histogram (HC) of 48 dimensions totally.

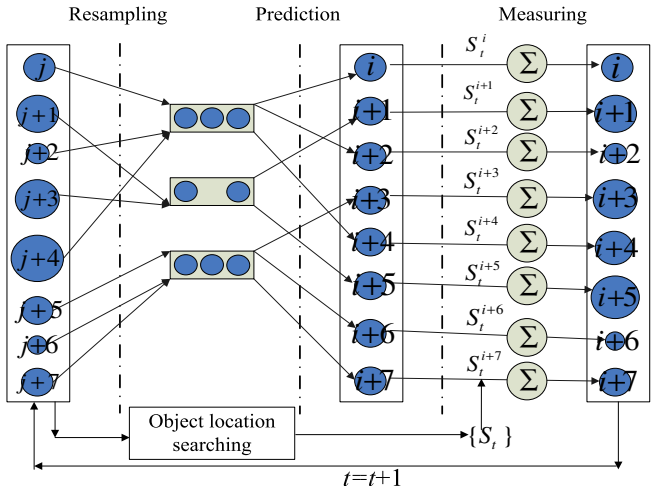

Fig. 6. Feature evaluation procedure of eight features using a Particle filter. Each blue cycle represents a particle. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3.1.2. Gradient orientation histogram

Motivated by the work in [22,23], a histogram of 72 dimensions is extracted to descript the gradient orientation of an object, called HOG. Details of HOG feature extraction are described as follows.

HOG is calculated in grayscale space. We first resize the rectangle object region into a normalized window of fixed size, say 32×32 pixels. Then, we divide the window into small spatial cells with the size of 8×8 and 4 (2×2) such adjacent cells are then integrated into a block, therefore we can obtain nine blocks numbered from ① to ⑨, which overlaps each other (shown in Fig. 3c). Different from the method in [22], each block in this approach constructs an 8-bin HOG without local normalization (shown in Fig. 3b). Each pixel in the block calculates its gradient orientation $ori(h, w)$ based on Eq. (7). The mask for the calculation of $ori(h, w)$ is shown in Fig. 3a. Then we combine the HOG of each block to obtain a 72-dimension feature for the whole window.

$$\begin{aligned} I &= G(\sigma, 0)^{*} I_0 \\ dy &= I(h+1, w) - I(h-1, w) \\ dx &= I(h, w+1) - I(h, w-1) \\ ori(h, w) &= a\tan 2(dy, dx) \quad \text{or} \quad i \in [-\pi, \pi] \end{aligned} \tag{7}$$

where $G(\sigma, 0)$ is a Gaussian smooth function and σ is the scale parameter determined empirically.

To deal with object rotation, we adopt the dominant orientation method of SIFT [25]. First, a coarse orientation histogram (8 bins instead of 72 bins) is calculated on the object when the tracking process is initialized, then the orientation bin with the largest value is detected and set as the dominant (reference) orientation of

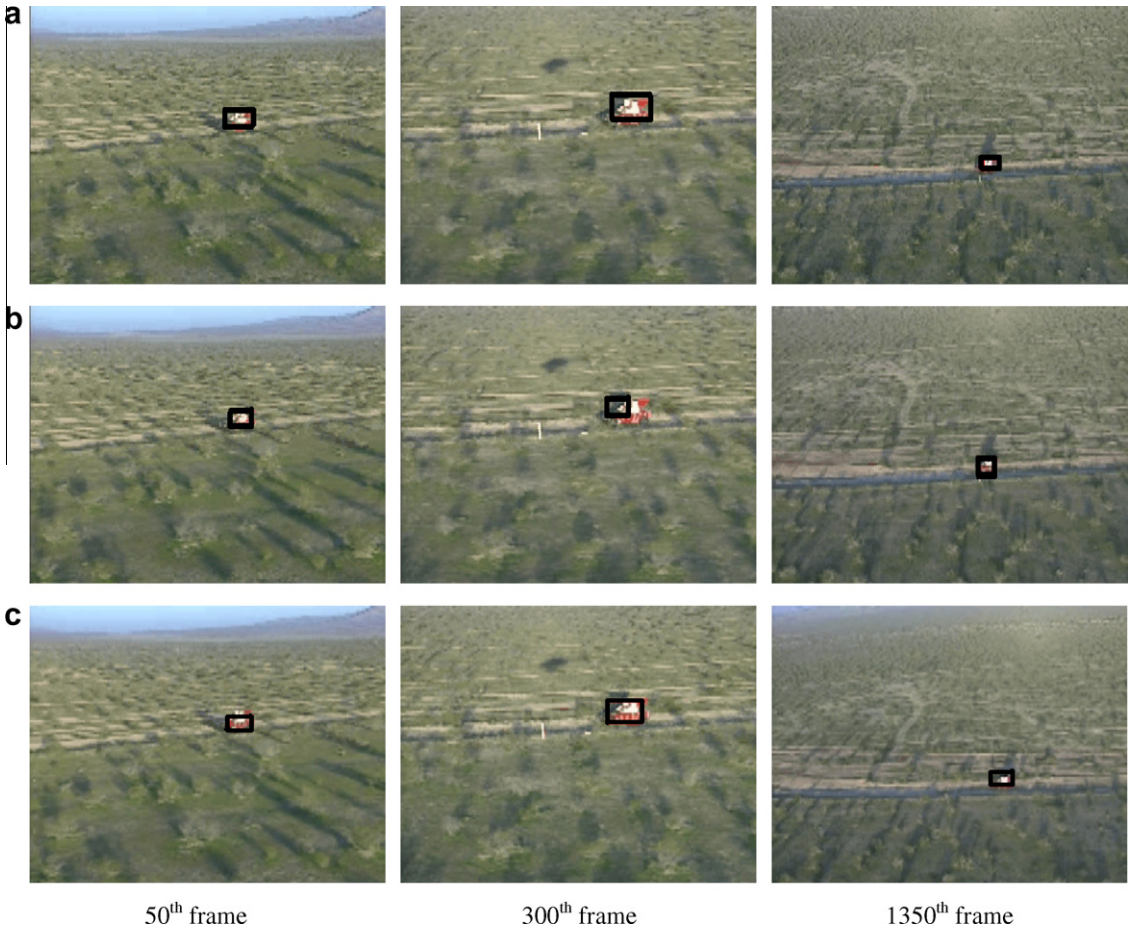

Fig. 7. Tracking results in relatively simple backgrounds. (a) Results of color-based tracking. (b) Results of SIFT-based tracking method. (c) Results of HOGC-based tracking method.

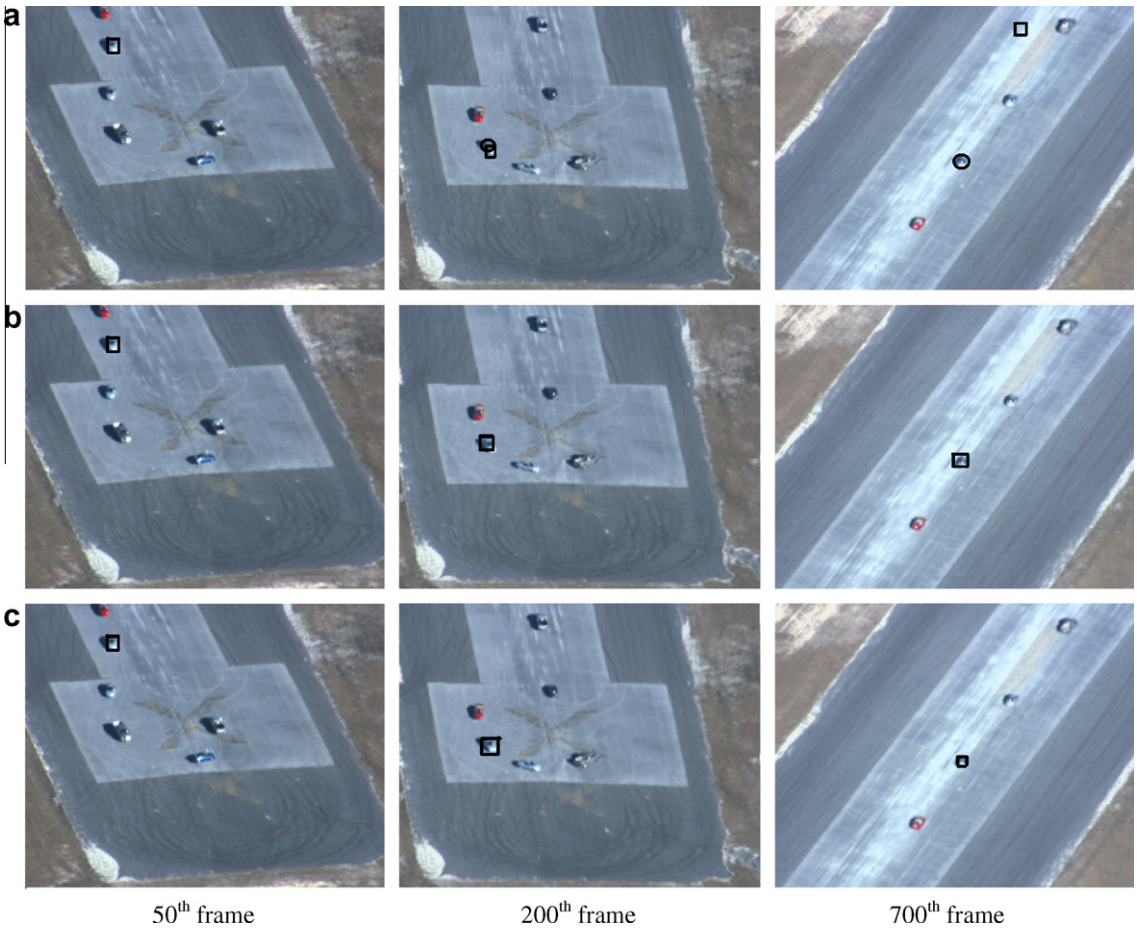

Fig. 8. Tracking results in background that has the similar color with the tracked object. The tracking result is marked with black rectangle and the ground-truth is with black ellipse, once there are tracking errors. (a) Results of color-based tracking. (b) Results of SIFT-based tracking. (c) Results of HOGC-based tracking.

the object, which is shown in Fig. 4a1 and a2. During the tracking process, we can calculate the dominant orientation of a candidate object and then rotate it to ensure that the instantaneous dominant orientation is consistent with the reference one, shown in Fig. 4b and c. Then we can calculate the 72-dimension HOG of the rotated candidate object for tracking. In a word, we normalize the dominant orientation of each candidate object before extracting its HOG, therefore the extracted HOG is insensitive to rotation to some extent.

3.2. Feature evaluation using Kalman filter

Kalman Filter provides a recursive solution to the linear optimal filtering problem and applies to stationary as well as non-stationary environment [11]. Feature evaluation in Kalman filter during a tracking process [27] is under the following constrains: (1) The confidence and discriminative ability (defined in Section 2, part A) of a feature is with Gaussian distribution, reflected by a float value fallen into [0.0, 1.0]. (2) Features of higher discriminative ability should have larger confidences, and vice versa.

We first define the state of the Kalman filter for feature evaluation as the combination of the confidence $W_t = \{w_t(1), w_t(2), \ldots, w_t(N)\}$ and its variation "velocity" $\Delta W_t = W_t - W_{t-1}$ of each feature, where $w_t(i)$ is the confidence of the ith feature. Then we define the measurements of the filter as $S_t = \{S_t^1, S_t^2, \ldots, S_t^N\}$, which is the discriminative ability vector at frame t. S_t^i is the discriminative ability of the ith feature and it can be calculated by Eqs. (1) and (2). Then we can obtain the prediction equation and the measurement equation of a Kalman filter as follows:

$$\begin{cases} \begin{pmatrix} W_{t+1} \\ \Delta W_{t+1} \end{pmatrix} = \begin{pmatrix} I_{N\times N} & I_{N\times N} \\ 0 & I_{N\times N} \end{pmatrix} \begin{pmatrix} W_t \\ \Delta W_t \end{pmatrix} + u_t \\ (S_t) = (I_{N\times N} 0) \begin{pmatrix} W_t \\ \Delta W_t \end{pmatrix} + v_t \end{cases} \tag{8}$$

where $I_{N\times N}$ is an identity matrix in our experiment. u_t and v_t are both Gaussian noises functions. Based on Eq. (8), the feature evaluation procedure using Kalman filter is presented in Table 1.

We illustrate the procedure of feature evaluation using Kalman filter in Fig. 5. The iterations of Kalman filter can ensure that the confidence of the feature is temporal consistency.where $L = (I_{N\times N}\ I_{N\times N}\ 0I_{N\times N})$ is the transition matrix of the prediction

Table 3
Video file list for average DER calculation.

Video test set	Video name
VIVID tracking video set	redteam.avi egtest01.avi egtest02.avi egtest04.avi
SDL tracking video set	xiangshan_ 0032.avi xiangshan_ 0043.avi
CARVIA tracking video set	Browse1.avi Fight_Chase.avi OneStopMoveEnter1cor.avi EnterExitCrossingPaths2front.avi

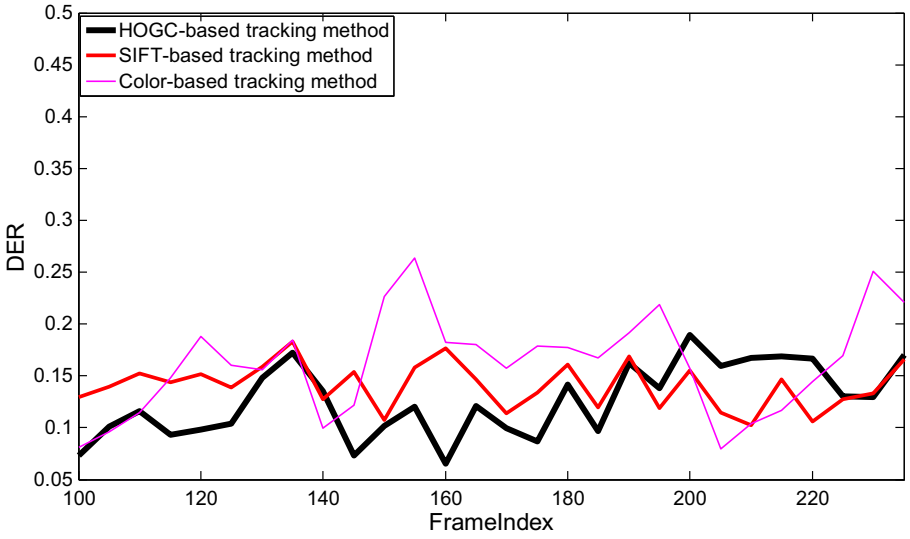

Fig. 9. Average DER of the three feature-based tracking methods.

equation, $H = (I_{N\times N}0)$ is the measurement matrix of the measurement equation.

3.3. Feature evaluation using Particle filter

Particle filter is an estimation algorithm for implementing a recursive temporal Bayesian filter by Monte Carlo simulations. It represents the posterior state of a moving object by a set of random samples with associated confidences [13]. The feature evaluation using Particle filter can deal with confidence or discriminative ability that is with non-Gaussian and non-linear distribution [28].

The core idea of this procedure is to define a Particle filter for the combined feature set, and each feature in the set is seen as a particle. In other words, a feature together with its confidence is a weighted sample, which is like that used for representing object moving state [13]. Supposing that there are a set of particles at frame t, $\{(i, w_t(i))\}$, $i = 1, 2, \ldots, N$, where i denotes the ith feature (particle) of feature set, $w_t(i)$ is the confidence of ith feature in frame t. Given these definitions, we can present the feature evaluation procedure as shown in Table 2:

We show the feature confidence assignment procedure of eight particles from frame t to $t + 1$ in Fig. 6. The iteration of this process can ensure that the evolution of feature confidence is temporal consistency even the variation of feature confidence is non-linear and non-Gaussian.

4. Experiments

In this section, experiments with comparisons are carried out to validate the proposed combined feature set and the feature evaluation approach.

The experimental videos are from VIVID [29], CAVIAR [30] and our SDL data set [31]. The test videos consist of a variety of cases, including occlusions between tracking objects, lighting changes, scale variations, object rotations and complex backgrounds. Some of the videos are captured on moving platforms and the target objects include moving pedestrians and vehicles. In the experiments, no image pre-processing module is employed.

4.1. Validation of the combined feature set HOGC

We compare our proposed combined feature set HOGC with other two representative ones, including the color [16] and SIFT [24] features. The test video in Fig. 7 is from the VIVID set, and the target object is a small jeep with a relative simple background and uniform object movement. All of the three methods obtained satisfied tracking results. The result of SIFT-based method has some instability caused by noisy SIFT keypoints (generated from scrubs around the object) in the 300th frame shown in Fig. 7b. Due to the obvious appearance difference between the object and its background (the object is white and red, while its background looks green), HOGC and color-based tracking methods obtain better tracking stabilization reflected by the stable outline box on the object.

The second video is also from VIVID set in which there are small cars moving on the dynamic background. In this video, the car being tracked first loops around on a runway, then goes straightly and speeds up. The appearance of the car changes remarkably in both size and orientation during the tracking process. The color between the car and its background is also similar. All these make the tracking very challenging. As shown in Fig. 8a, color-based tracking has some tracking instability in the 200th frame and loses the tracked object in the 700th frame because of the similar colors between the car and its background. Since the SIFT feature can represent the car well, SIFT-based method obtains good results shown in (Fig. 8b). Our combined feature set, for representation of both color and contour, also obtains satisfied tracking results (Fig. 8c).

The main drawback of SIFT is its low computational efficiency which make it not appropriate to a real-time tracking application. In the experiments, we find that the proposed HOGC-based method can work almost real time (about 20 frames per second averagely) on a computer with Pentium IV CPU (2.4G), which is almost as fast as the color-based method, and is much faster than the SIFT-based one.

To quantitatively evaluate the proposed combined feature set, we define a criterion entitled the relative displacement error rates (DER).

$$\mathrm{DER} = \frac{\text{Displacement error between tracked object position and ground-truth}}{\text{Size of the object}}$$

Fig. 10. Tracking results of some occlusions and scale variation. (a) Results of color-based tracking method. (b) Results of variance ratio feature shift method. (c) Results of Peak difference feature shift method. (d) Results of our proposed tracking approaches based on HOGC feature evaluation in Kalman filter and (e) in Particle filter.

In experiments, we use the average DER of 10 video clips (listed in Table 3) to reflect the tracking performance. The lower the average DER is, the better the tracking performance. The tracking results of the three features are shown in Fig. 9. It can be seen that the average DER of HOGC-based tracking, which is close to that of the SIFT-based tracking (about 0.05–0.2), is smaller than that of the color-based tracking (about 0.10–0.25) in almost the whole tracking process.

4.2. Validation of feature evaluation in filter framework

We compare our approaches (object tracking based on feature evaluation in Kalman filter and Particle filter) with other three methods, including color-based tracking method (no feature evaluation) [16], Variance ratio feature shift [19], and Peak difference feature shift tracking methods [19]. The last two are representative feature selection/evaluation methods for adaptive visual tracking.

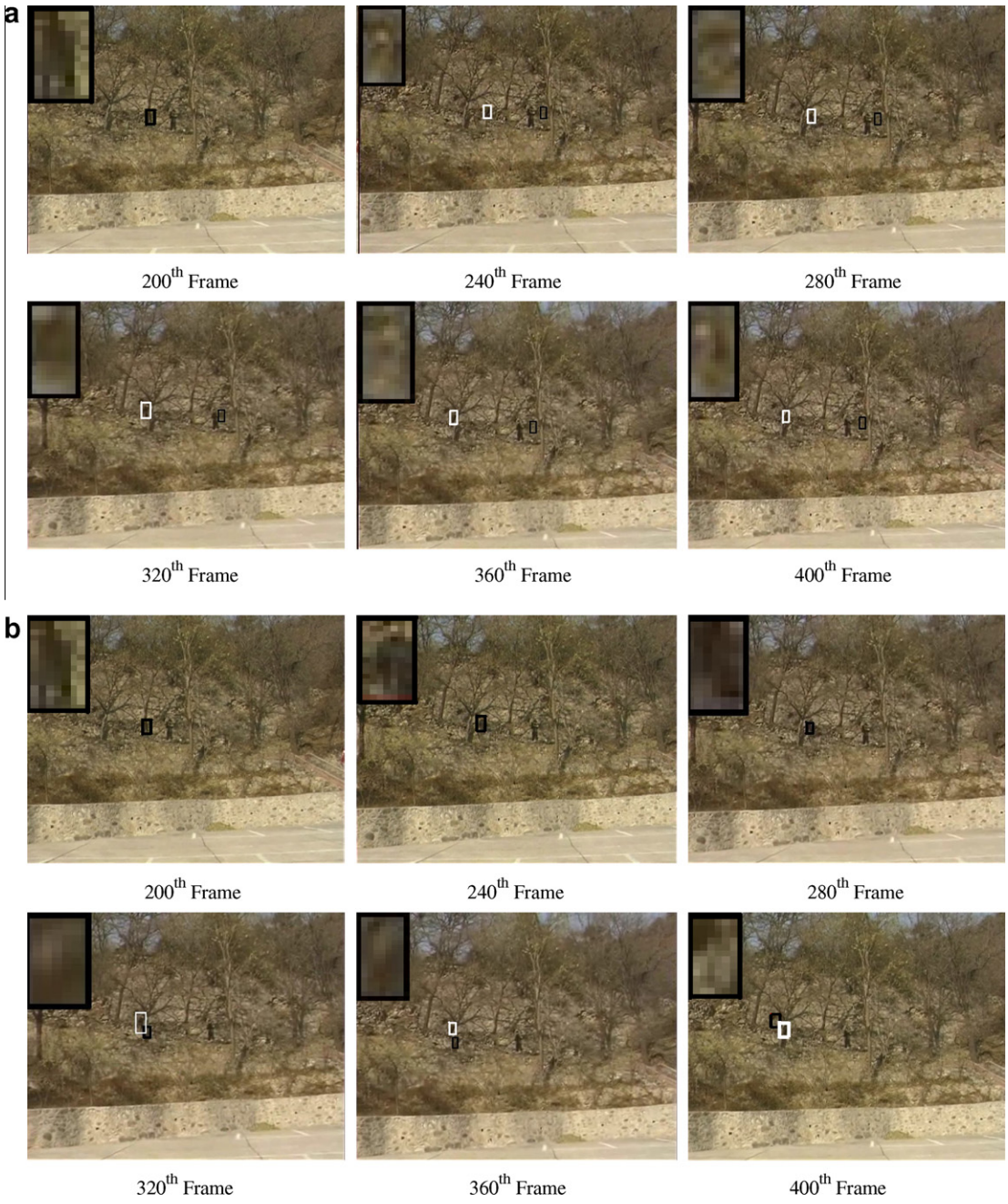

Fig. 11. Tracking results in complex backgrounds. The tracking results are marked with black rectangle and the ground-truth is with white rectangle, once there are tracking errors. (a) Results of color-based tracking method. (b) Results of variance ratio feature shift method. (c) Results of Peak difference feature shift method. (d) Results of Our proposed tracking methods based on HOGC feature evaluation in Kalman filter and (e) in Particle filter.

In the experiment shown in Fig. 10, the main challenges of tracking in this video sequence arise from partial occlusions of the object by other pedestrians and scale variation. Color-based (Fig. 10a) tracking has some instability shown in the 665th frame and the 829th frame, where there are illumination changes and the color between the object and the mimic object are quite indistinctive. Although the variance ratio and Peak difference feature shift methods (Fig. 10b and c) can distinguish the object from its background to some extent, there are tracking errors in 829th frame. It can be seen that our proposed approaches (Fig. 10d and e) obtain the best tracking results. The target pedestrian, marked with white box, is tracked steadily all the duration.

In Fig. 11, we illustrate the challenging test video in which the background has the similar color to the tracking object and at the same time, there are some small trees which are quite similar to the object in contour. All five methods mentioned above are tested on this video for the comparison purpose, and the top-left rectangle gives the tracking results in a large view. It can be seen that our proposed methods are able to track the object robustly (shown in Fig. 11d and e) even in such a complex circumstance. The color-based tracking method (Fig. 11a) lost the tracked pedestrian in the 240th, 280th, 320th, 360th and 400th frames. Variance ratio and Peak difference feature shift tracking methods with online feature selection can track the object in some frames, while it lost the

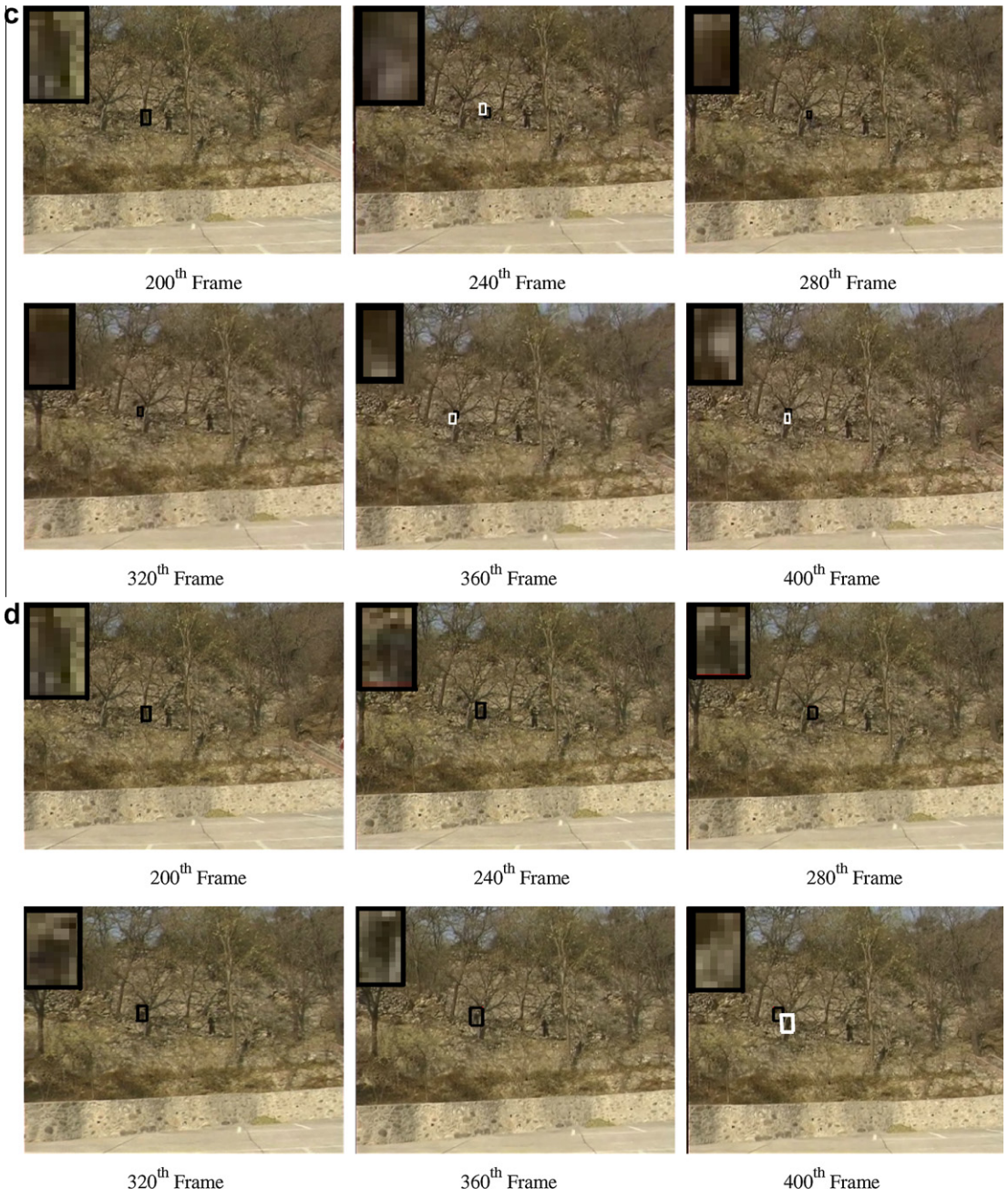

Fig. 11 (*continued*)

object in the flowing frames (such as the 320th, 360th and 400th frames in Fig. 11b and the 240th, 360th and 400th frames in Fig. 11c). The reason may be that these methods do not consider the temporal consistency of feature evolution, which will make them suffer from unstable feature confidences caused by mimic objects in the background.

In the experiments, we use the average DER (10 video clips listed in Table 3) to show the performance of each method. There are various factors that make the tracking challenging: different viewpoints (most of these sequences are captured by moving cameras), illumination changes, variations of the objects and partial occlusions. The results of the five methods are shown in Fig. 12. It can be seen that the average DER of our proposed approaches (about 0.1–0.15) is much lower than those of the other methods, which validate that the object tracking with the proposed feature evaluation approach has a better adaptation.

5. Conclusions

Online feature evaluation is very important to improve the adaptability of visual object tracking. In this paper, we propose a novel feature evaluation approach and then give implementations of the approach in Kalman and Particle filters respectively. Experimental results with comparisons are provided, which validates the effectiveness of both the combined feature set and the feature evaluation approach. The results also indicated that object tracking with the proposed combined feature evaluation outperforms the existing methods. The proposed approach also shows its adaptation when the background is complex or there are object/background appearance variations.

The new concepts and techniques introduced in this paper include HOGC, which is the combined feature set of HC and HOG, processing of object rotation, and feature evaluation using filter

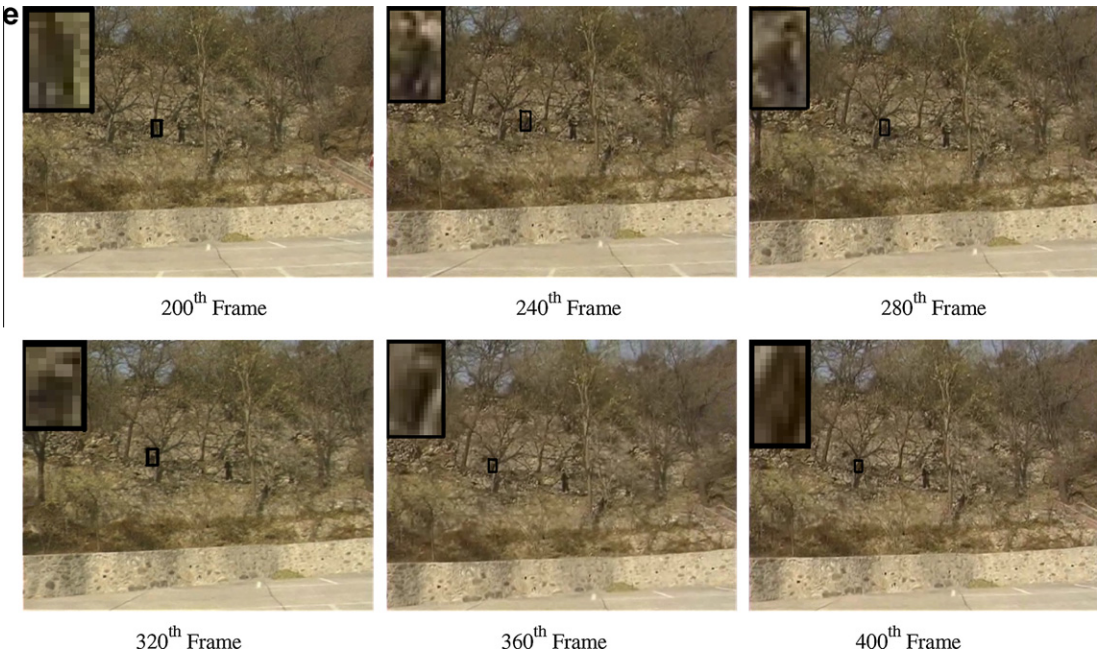

Fig. 11 (*continued*)

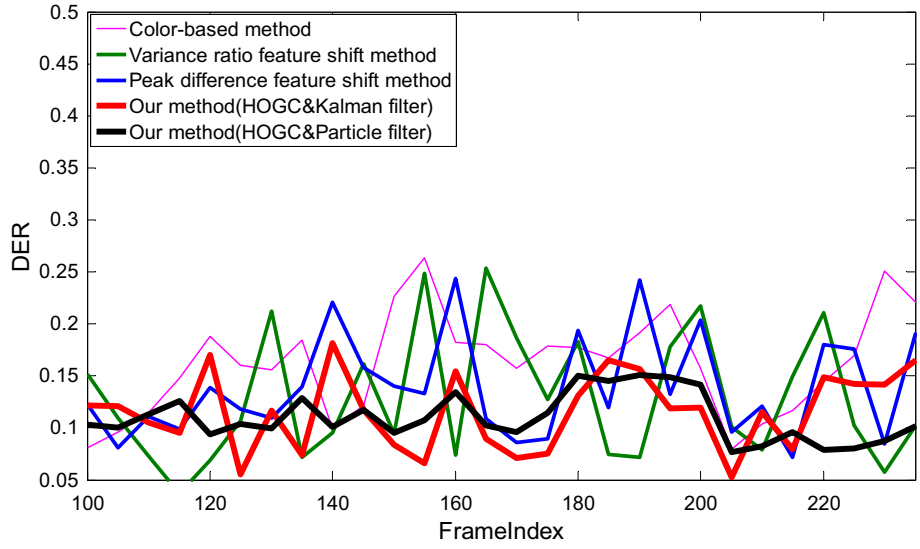

Fig. 12. Average DER of five tracking methods.

frameworks. In particular, the combined feature set can be easily extended by integrating other features, such as texture, feature points, and so on. In the evaluation process, we extend the function of tracking filters, which is novel in visual object tracking research.

A known disadvantage of the proposed feature set is that it can not cope with large scale changes of the target object, for the tracking template updating issue [32] is not considered, and the proposed tracking approach cannot handle multiple objects tracking at present either. These issues should be considered in the future work.

Acknowledgments

The authors would like to thank Dr. Baochang Zhang, Guangyu Zhu, Xiangyang Ji, Ran Xu and the anonymous reviewers for their constructive comments. This work is supported in part by National Basic Research Program of China (973 Program) with Nos. 2011CB706900 and 2010CB731800, and National Science Foundation of China with Nos. 61039003 and 60872143.

Appendix A

When we use Kalman filters for both movement modeling and feature evaluation, we can select a lump filter model or separated filter models. In our previous work [27], we have used a lumped Kalman filter model for the feature evaluation and the movement (location and velocity) of the tracked object as shown in Eq. (9).

$$\begin{cases} \begin{pmatrix} W_{t+1}\Delta W_{t+} \\ Pos_{t+1} \\ \Delta Pos_{t+1} \end{pmatrix} = \begin{pmatrix} I_{N\times N} & I_{N\times N} & 0 & 0 \\ 0 & I_{N\times N} & 0 & 0 \\ 0 & 0 & I_{M\times M} & I_{M\times M} \\ 0 & 0 & 0 & I_{M\times M} \end{pmatrix} \begin{pmatrix} W_t\Delta W_t \\ Pos_t\Delta Pos_t \end{pmatrix} + u_t \\ \begin{pmatrix} S_t \\ mPos_t \end{pmatrix} = \begin{pmatrix} I_{N\times N} & 0 & 0 & 0 \\ 0 & 0 & I_{M\times M} & 0 \end{pmatrix} \begin{pmatrix} W_t\Delta W_t \\ Pos_t\Delta Pos_t \end{pmatrix} + v_t \end{cases} \quad (9)$$

where $Pos_t = (x, y)_t$ is the predicted location of the tracked object and $\Delta Pos_t = Pos_t - Pos_{t-1}$. $mPos_t = (mx, my)_t$ is the location obtained during the matching procedure. And M is 2, representing a two dimensional position. Other symbols have same meaning as Eq. (8).

Different from [27], here we use two separate Kalman filters for object tracking. However, according the definitions of the transition matrix and measurement matrix in Eq. (9), with a matrix partitioning operation, the lumped Kalman filter can be equally separated as the following two Kalman filters:

$$\begin{cases} \begin{pmatrix} W_{t+1} \\ \Delta W_{t+1} \end{pmatrix} = \begin{pmatrix} I_{N\times N} & I_{N\times N} \\ 0 & I_{N\times N} \end{pmatrix} \begin{pmatrix} W_t \\ \Delta W_t \end{pmatrix} + u_t \\ (S_t) = (I_{N\times N} \quad 0) \begin{pmatrix} W_t \\ \Delta W_t \end{pmatrix} + v_t \end{cases} \text{ and}$$

$$\begin{cases} \begin{pmatrix} Pos_{t+1} \\ \Delta Pos_{t+1} \end{pmatrix} = \begin{pmatrix} I_{M\times M} & I_{M\times M} \\ 0 & I_{M\times M} \end{pmatrix} \begin{pmatrix} Pos_t \\ \Delta Pos_t \end{pmatrix} + u_t \\ (mPos_t) = (I_{M\times M} \quad 0) \begin{pmatrix} Pos_t \\ \Delta Pos_t \end{pmatrix} + v_t \end{cases} \quad (10)$$

While the left one Kalman filter is equal to the Eq. (8), which is used to evaluate the feature confidence and the right one Kalman filter is a traditional one, which is often used to model the movement of the tracked object [12]. Therefore, object tracking can be performed either using a lumped Kalman filter (shown in Eq. (9)) or using two separate Kalman filters (one is for the movement of the candidate area of tracked object, the other one is for the feature evaluation during the tracking process), which is proved equal in theory, while we select a separated one in this paper. In our previous work [27], we have tested the lump model and similar tracking results are reported.

References

[1] C. Stauffer, W. Grimson, Adaptive background mixture models for real-time tracking, in: Proceedings of IEEE conference on CVPR, 1999, pp. 246–252.
[2] G. Bradski, Real time face and object tracking as a component of a perceptual user interface, in: Proceedings of IEEE workshop on applications of computer vision, 1998, pp. 214–219.
[3] N. Papanikolopoulos, P. Khosla, T. Kanade, Visual tracking of a moving target by a camera mounted on a robot: a combination of control and vision, IEEE Transactions on Robotics and Automation 9 (1993) 14–35.
[4] A. Yilmaz, X. Li, B. Shah, Contour based object tracking with occlusion handling in video acquired using mobile cameras, IEEE Transactions on PAMI26 (2004) 1531–1536.
[5] J.Y. Pan, B. Hu, J.Q. Zhang, Robust and accurate object tracking under various types of occlusions, IEEE Transactions on CSVT 18 (2008) 223–236.
[6] K. Hariharakrishnan, D. Schonfeld, Fast object tracking using adaptive block matching, IEEE Transactions on Multimedia 7 (2005) 853–859.
[7] J. Shi, C. Tomasi, Good features to track, in: Proceedings of IEEE Conference on CVPR, 1994, pp. 593–600.
[8] S. Birchfield, Elliptical head tracking using intensity gradients and color histograms, in: Proceedings of IEEE Conference on CVPR, 1998, pp. 232–237.
[9] D. Comaniciu, V. Ramesh, P. Meer, Kernel-based object tracking, IEEE Transactions on PAMI 25 (2003) 564–577.
[10] G. Hager, P. Belhumeur, Efficient region tracking with parametric models of geometry and illumination, IEEE Transactions on PAMI 20 (1998) 1025–1039.
[11] R.G. Brown, P.Y.C. Hwang, Introduction to random signals and applied Kalman Filtering [M], John Wiley & Sons, Inc., New York, 1992.
[12] E. Cuevas, D. Zaldivar, R. Rojas, Kalman filter for vision tracking, Technical Report B, Fachbereich Mathematikund Informatik, Freie Universität Berlin, 2005.
[13] E. Cuevas, D. Zaldivar, R. Rojas, Particle filter for vision tracking, Technical Report B, Fachbereich Mathematikund Informatik, Freie Universität Berlin, 2005.
[14] M. Isard, A. Blake, CONDENSATION – conditional density propagation for visual tracking, International Journal on Computer Vision 29 (1998) 5–28.
[15] D. Comaniciu, V. Ramesh, P. Meer, Real-time tracking of non-rigid objects using mean shift, in: Proceedings of IEEE Conference on CVPR, 2000, pp. 142–149.
[16] T.D. Grove, K.D. Baker, T.N. Tan, Colour based object tracking, in: 14th International Conference on Pattern Recognition, vol.2, 1998.
[17] D.W. Liang, Q.M. Huang, W. Gao, H.X. Yao, Online selection of discriminative features using Bayes error rate for visual tracking, in: 7th Pacific-Rim Conference on Multimedia, 2006, pp. 547–555.
[18] D. Chen, J. Yang, Robust object tracking via online spatial bias appearance model learning, IEEE Transactions on PAMI 29 (2007) 2157–2169.
[19] R. Collins, Y. Liu, M. Leordeanu, Online selection of discriminative tracking features, IEEE Transactions on PAMI 27 (2005).
[20] J. Wang, X. Chen, W. Gao, Online selecting discriminative tracking features using particle filter, in: Proceedings of IEEE Conference on CVPR, 2005, pp. 1037–1042.
[21] J.Q. Wang, Y.S. Yagi, Integrating color and shape-texture features for adaptive real-time object tracking, IEEE Transactions on Image Processing 17 (2008) 235–240.
[22] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Proceedings of IEEE Conference on CVPR, 2005, pp. 1063–6919.
[23] Q. Zhu, S. Avidan, M.C. Yeh, K.T. Cheng, Fast human detection using a cascade of histograms of oriented gradients, in: Proceedings of IEEE Conference on CVPR, 2006.
[24] Y.J. Li, J.F. Yang, R.B. Wu, F.X. Gong, efficient object tracking based on local invariant features, in: IEEE International Symposium on Communications and Information Technologies, 2006, pp. 697–700.
[25] D.G. Lowe, Distinctive image features from scale-invariant keypoints, International Journal on Computer Vision 17 (2004) 91–110.
[26] W.C. Huang, T.C. Hwann, A square-root sampling approach to fast histogram based search, in: Proceedings of IEEE Conference on CVPR, 2010, pp. 3043–3049.
[27] Z.J. Han, Q.X. Ye, J.B. Jiao, Online feature evaluation for object tracking using kalman Filter, in: 19th International Conference on Pattern Recognition, 2008.
[28] Z.J. Han, Q.X. Ye, J.B. Jiao, Feature evaluation by particle filter for adaptive object tracking, in: Proceedings of SPIE Visual Communication and Image Processing, 2009.
[29] VIVID Tracking Evaluation. <http://www.vividevaluation.ri.cmu.edu/datasets/datasets.html>.
[30] CAVIAR Test Case Scenarios. <http://homepages.inf.ed.ac.uk/rbf/CAVIAR>.
[31] SDL Data Set. <http://coe.gucas.ac.cn/SDL-HomePage/resource.asp>.
[32] I. Matthews, T. Ishikawa, S. Baker, The template update problem, IEEE Transactions on PAMI 26 (2004) 810–815.

Zhenjun Han received his B.S. degree in software engineering from Tianjin University (TJU), Tianjin, in 2006. Since 2009, he has been a Ph.D candidate of the Graduate University of Chinese Academy of Sciences, Beijing, China. His research interests include image processing and intelligent surveillance etc.

Qixiang Ye received his B.S. and M.S. degrees in mechanical & electronic engineering from Harbin Institute of Technology of China (HIT), Harbin, in 1999 and in 2001 respectively. He received his Ph.D degree from the Institute of Computing Technology, Chinese Academy of Sciences in 2006. Since 2009, he has been an associate professor at the Graduate University of the Chinese Academy of Sciences, Beijing. His research interests include image processing, pattern recognition, and statistic learning etc.

Jianbin Jiao received the B.S., M.S. and Ph.D degrees in mechanical and electronic engineering from Harbin Institute of Technology of China (HIT), Harbin, in 1989, 1992, and 1995 respectively. From 1997 to 2005, he was an associate professor of HIT. Since 2006, he has been a professor of the Graduate University of Chinese Academy of Sciences, Beijing. His research interests include image processing, pattern recognition, and intelligent surveillance, etc.