Robust Object Tracking Based on Principal Component Analysis and Local Sparse Representation

Haicang Liu, Shutao Li, *Senior Member, IEEE*, and Leyuan Fang, *Member, IEEE*

***Abstract*— Object tracking methods based on the principal component analysis (PCA) are effective against object change caused by illumination variation and motion blur. However, when the object is occluded, the tracking result of the PCA-based methods will drift away from the target. In this paper, we propose a new robust object tracking method based on the PCA and local sparse representation (LSR). First, candidates are reconstructed through the PCA subspace model in global manner. To handle occlusion, a patch-based similarity estimation strategy is proposed for the PCA subspace model. In the patch-based strategy, the PCA representation error map is divided into patches to estimate the similarity between target and candidate considering the occlusion. Second, the LSR is introduced to detect the occluded patches of the object and estimate the similarity through the residual error in the sparse coding. Finally, the two similarities of each candidate from the PCA subspace model and LSR model are fused to predict the tracking result. The experimental results demonstrate that the proposed tracking method favorably performs against several state-of-the-art methods on challenging image sequences.**

***Index Terms*— Object tracking, particle filter, principal component analysis (PCA), similarity estimation, sparse representation (SR).**

I. Introduction

OBJECT tracking plays an important role in computer vision applications [1]–[3]. In tracking task, object appearance often changes because of partial occlusion, motion blur, and illumination variation. Due to the difficulties, robust object tracking is still a challenging problem although various methods have been proposed [4]–[6].

The existing object tracking methods can be categorized into two classes: 1) discriminative methods and 2) generative methods [7]. The discriminative methods track the object in the image frames by modeling a conditional distribution to classify the object and background [8], [9]. In these methods, the tracking results will drift away from the target when enough object samples are not available for training the classifier. On the other hand, the generative methods track the object target by modeling a joint distribution and estimating the likelihoods of candidates regardless of the background [10], [11]. In this paper, we are focused on the generative methods.

In generative object tracking methods, the object appearance representation is an important problem that greatly affects the similarity estimation. Many representation methods have been proposed, such as subspace model based on the principal component analysis (PCA) [12], [13], object template based on pixel intensity [14], [15], and sparse representation (SR) model based on dictionary [16]–[18].

Subspace model based on the PCA is robust against the object appearance change [19]–[21]. Using the PCA, Ross *et al.* [22] proposed an incremental learning visual tracking (IVT) method to improve the robustness of the tracker. However, the IVT method is global, which is sensitive to the occlusion.

Considering the partial occlusion, Adam *et al.* [11] proposed a Frag-Track method based on object template. In the Frag-Track method, the template and candidates are represented in patches. The similarity between the template patch and candidate patch is measured by comparing the gray-level or color histograms. The similarities from all the patches are integrated and used to estimate the likelihood of the candidate. The occluded part is eliminated via a fixed threshold of the patch similarity. The Frag-Track method solves the partial occlusion problem, but it is not adaptive to the object appearance change due to the fixed template.

Recently, SR is applied in object tracking. Zhong *et al.* [23] and Yang *et al.* [24] have done much outstanding works for object tracking based on SR. There are mainly two manners to use the SR in object tracking, i.e., global SR (GSR) manner and local SR (LSR) manner. In the GSR, the holistic object is regarded as one atom in dictionary. To alleviate drift caused by object occlusion, a large number of trivial templates need to be found [5], [13]. The trivial templates increase the computing cost in sparse coding. In the LSR, object is sparse coded with dictionary in patches. The likelihoods of candidates are estimated through the patches and structure information of the object [25], [26].

Manuscript received February 23, 2015; revised March 30, 2015; accepted April 7, 2015. This work was supported in part by the National Natural Science Foundation of China under Grant 61172161, in part by the National Natural Science Foundation for Distinguished Young Scholars of China under Grant 61325007, in part by the Independent Research Funds through the State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, Hunan University, Changsha, China, under Grant 71165002, and in part by the Fundamental Research Funds for the Central Universities. The Associate Editor coordinating the review process was Dr. Shervin Shirmohammadi.

H. Liu is with the State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, College of Electrical and Information Engineering, Hunan University, Changsha 410082, China (e-mail: liuhaicang@hnu.edu.cn).

S. Li and L. Fang are with the College of Electrical and Information Engineering, Hunan University, Changsha 410082, China (e-mail: shutao_li@hnu.edu.cn; fangleyuan@gmail.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIM.2015.2437636

The PCA subspace model is robust against target change caused by illumination variations and motion blur in tracking moving object through representing object appearance in global manner. Meanwhile, the LSR model can accurately detect the object occlusion through representing object appearance in patches. In this paper, we propose a new robust object tracking method combining the PCA and LSR to further improve the tracking performance. The particle filter is used as the tracking frame in which candidates are generated in the state model by Gaussian distribution. For the observation model of particle filter, first candidates are represented through the PCA subspace and LSR, respectively. Second, considering the ability of handling the partial occlusions, a patch-based similarity estimation strategy for the PCA subspace model is proposed in which the occluded patches are measured through the residual error of the LSR model. Two similarities of each candidate are separately estimated through the patch-based representation error of the PCA subspace model and the residual error of the LSR model. Finally, the similarities from the two models are fused to predict the tracking result.

The related works are introduced in Section II. The tracking frame based on particle filter is presented in Section III. The patch-based PCA–LSR tracking method is proposed in Section IV. Performance comparisons and analyses of several state-of-the-art object tracking methods are provided in Section V. Finally, the concluding remark is given in Section VI.

II. Related Works

In the IVT method [22], the object appearance is represented through the mean and eigenbasis of the PCA. This online learning method is adaptive to the object appearance changes, especially to the changes caused by extrinsic illumination variations, object motion, and camera motion. Since this method represents the object as a holistic target, the IVT method cannot well handle the partial occlusions. The track result will drift away when the object is partial occluded.

Zhong *et al.* [23] proposed a method via sparse collaborative appearance model. In their sparse collaborative appearance model, a sparse discriminative classifier (SDC) and a sparse generative model (SGM) are developed and combined. Through holistic templates, the SDC separates the foreground object from the background. In the SGM, a histogram-based method considering the spatial information of each local patch of the object is used to measure the similarity between the candidate and the template. In the sparse collaborative appearance model, occlusion is discussed only in the SGM but omitted in the SDC.

The most relevant works to the proposed method are those in [13] and [26]. The online sparse prototypes tracking method (OSPT) in [13] is a global method in which eigenbases of the PCA are used as atoms in the dictionary of SR. In the OSPT method, a large number of trivial templates are needed to handle occlusion. The adaptive structural local sparse tracking method (ASLST) in [26] exploits both local information and global information of the target to compute the likelihoods of the candidates in the SR model. The ASLST method updates the object appearance as a holistic representation in the PCA subspace model. Different with the OSPT and ASLST methods, the proposed method estimates the similarities of the candidates for the PCA subspace model and the SR model both in local manner.

III. Particle Filter Tracking Frame

In image sequence, the object tracking can be seen as a Bayesian filtering process [5]. Let $\mathbf{x}$ denote the state variable describing the motion parameters of an object and $\mathbf{y}$ denote the observation. $\mathbf{x} = [l_x, l_y, \theta, s, \alpha, \phi]$, where $l_x, l_y, \theta, s, \alpha$, and ϕ denote x, y the translations, rotation angle, scale, aspect ratio, and skew, respectively. $\mathbf{x}_t$ and $\mathbf{y}_t$ denote the state and observation in the tth frame, and $\mathbf{y}_{1:t-1} = \{\mathbf{y}_1, \mathbf{y}_1, \ldots, \mathbf{y}_{t-1}\}$ is the observation set of the previous $t-1$ frames. Given the previous observation $\mathbf{y}_{1:t-1}$, the state of the object in the tth frame $\mathbf{x}_t$ can be predicted as

$$P(\mathbf{x}_t|\mathbf{y}_{1:t-1}) = \int P(\mathbf{x}_t|\mathbf{x}_{t-1})P(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})d\mathbf{x}_{t-1}. \quad (1)$$

Then, $\mathbf{x}_t$ can be decided through the maximum *a posteriori* estimation as

$$P(\mathbf{x}_t|\mathbf{y}_{1:t}) \propto P(\mathbf{y}_t|\mathbf{x}_t)P(\mathbf{x}_t|\mathbf{y}_{1:t-1}). \quad (2)$$

$P(\mathbf{x}_t|\mathbf{x}_{t-1})$ is the state model that represents the state transition of the object between the two consecutive frames. $P(\mathbf{y}_t|\mathbf{x}_t)$ is the observation model that denotes the likelihood of the candidate.

Regardless of the state distribution, the particle filter is an effective realization of Bayesian filtering [27], [28]. In particle filter, the particles $\mathbf{x}_t^n$ are predicted by a Gaussian function with mean $\mathbf{x}_{t-1}$ and variance σ^2 as

$$P\left(\mathbf{x}_t^n|\mathbf{x}_{t-1}\right) = G(\mathbf{x}_{t-1}, \sigma^2). \quad (3)$$

The optimal state is obtained as

$$\hat{\mathbf{x}}_t = \arg\max_{\mathbf{x}_t^n} P\left(\mathbf{y}_t^n|\mathbf{x}_t^n\right) P\left(\mathbf{x}_t^n|\mathbf{x}_{t-1}\right). \quad (4)$$

In this paper, the observation model $P(\mathbf{y}_t|\mathbf{x}_t)$ of the particle filter is estimated through computing the similarity between target and candidate based on the fusion of the PCA and LSR.

IV. Patch-Based PCA–LSR Tracking Method

To make full use of the advantages of the PCA subspace model and SR model, the two models are combined in estimating the similarity of the candidate. First the candidate is separately represented with the two models. Both similarities of each candidate from the PCA subspace model and LSR model are computed based on patches. The residual errors in the LSR model are used in detecting occluded patches to guide the PCA subspace model. Then, the two similarities from the two models are fused to make decision. The patch-based PCA–LSR observation model is shown in Fig. 1.

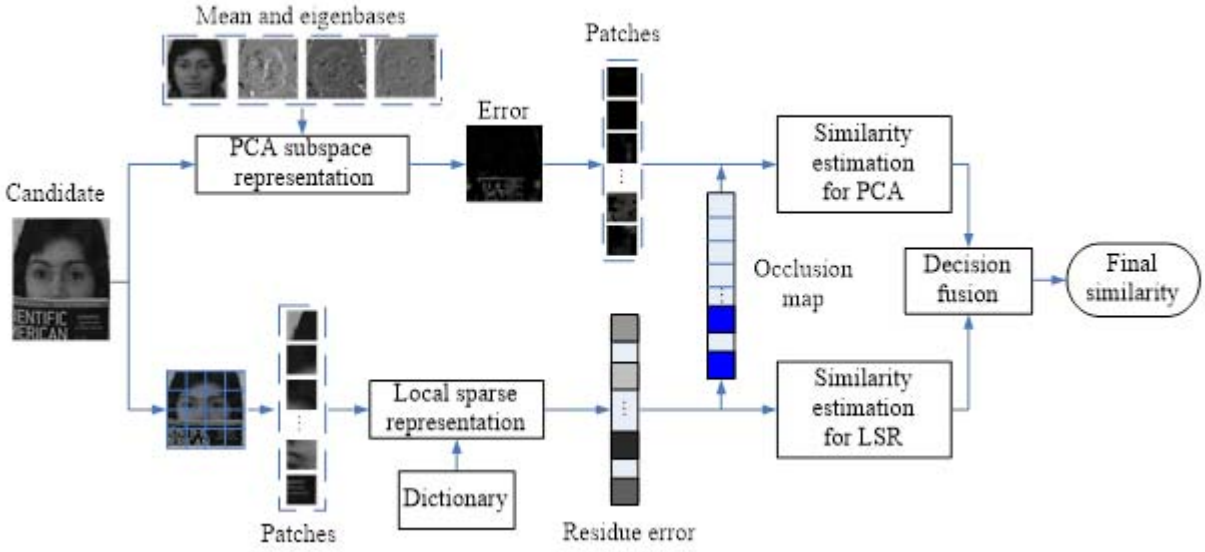


Fig. 1. Patch-based PCA–LSR observation model.

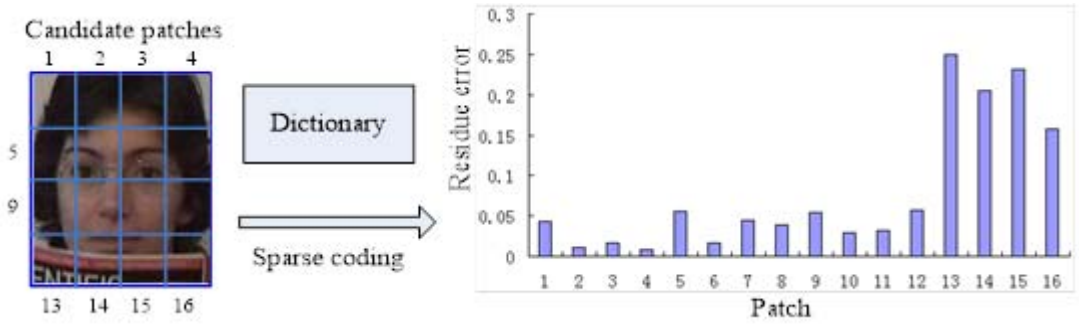


Fig. 2. Residual error comparison of object patches.

A. Object Appearance Representation

In the PCA subspace model, the object appearance is represented through the mean and eigenbasis. The eigenbasis is computed through the singular value decomposition (SVD). With the arrival of new object appearances, the mean and eigenbasis need to be updated to reduce the representation error. For online applications in the image sequences, the incremental learning method is used [22]. The mean $\bar{\mathbf{Z}}$ and eigenbasis matrix $\mathbf{U}$ are computed from the previous frames through the SVD. Then, the candidate $\mathbf{Z}_c$ in the tth frame can be represented with $\bar{\mathbf{Z}}$ and $\mathbf{U}$, and the representation error $\mathbf{E}_{\text{PCA}}$ is

$$\mathbf{E}_{\text{PCA}} = Z_c - (\bar{Z} + \mathbf{U}\mathbf{q}) \tag{5}$$

where $\mathbf{q}$ is the eigenbasis coefficient.

In the LSR model, the object appearance is represented through the dictionary and sparse coefficients [17], [23]. The dictionary $\mathbf{D}$ is composed of the patches that are obtained from the object in the first frame. The candidate $\mathbf{Z}_c$ in the tth frame is divided into patches, $\mathbf{Z}_c = [\mathbf{z}_c^1\ \mathbf{z}_c^2 \ldots \mathbf{z}_c^J]$, where J is the number of patches. The patches are sparse coded through the dictionary as

$$\min_{\beta_j} \left\|\mathbf{z}_c^j - \mathbf{D}\beta_j\right\|_2^2 + \lambda\|\beta_j\|_1 \quad \text{s.t. } \beta_j \geq 0 \quad (1 \leq j \leq J) \tag{6}$$

where β_j is the sparse coefficient and λ is a weight parameter. The residual error e_{LSR}^j of the patch $\mathbf{z}_c^j$ in the LSR model is

$$e_{\text{LSR}}^j = \left\|\mathbf{z}_c^j - \mathbf{D}\beta_j\right\|_2^2. \tag{7}$$

Then, the residual error of the candidate $\mathbf{Z}_c$ is $\mathbf{E}_{\text{LSR}} = [e_{\text{LSR}}^1,\ e_{\text{LSR}}^2, \ldots, e_{\text{LSR}}^J]$.

B. Similarity Estimation

The similarities of candidates are, respectively, estimated from the PCA subspace model and LSR model. Occlusion has a great influence on the similarity estimation. The object target will be lost in the tracking due to the accumulation of drift caused by occlusion. So, before estimating the similarity of candidates, occlusion patches need to be detected.

In the LSR model, the dictionary is composed of the object patches. So, in the sparse coding, the patches belonging to the object can be represented accurately, while the occlusion patches that do not belong to the object will have a large residual error. One example is shown in Fig. 2. It can be observed that the residue errors of the occlusion patches are obviously larger than those of the other patches.

Thus, the patches can be signed via the residual error in the occlusion map as

$$g_j = \begin{cases} 0 & e^j_{\mathrm{LSR}} > \eta \text{ (occlusion patch)} \\ 1 & \text{otherwise (object patch)} \end{cases} \tag{8}$$

where η is an adaptive threshold that is learned from the previous frames of the corresponding image sequences.

The candidate should have a small summation of all the e^j_{LSR} if it is the correct object. Considering the occlusion map, we define the similarity between the target and the candidate in the LSR model as

$$P_{\mathrm{LSR}}(\mathbf{y}^n|\mathbf{x}^n, \mathbf{g}^n) = \prod_{j=1}^{J} \exp\left(-e^j_{\mathrm{LSR}-O}\right) \tag{9}$$

where

$$e^j_{\mathrm{LSR}-O} = \begin{cases} \eta & g_j = 0 \\ e^j_{\mathrm{LSR}} & g_j = 1, \end{cases} \quad \mathbf{g}^n = \left[g^n_1, g^n_2, \ldots, g^n_J\right].$$

In the PCA subspace model, the similarity can be estimated by the representation error as

$$P(\mathbf{y}^n|\mathbf{x}^n) = \exp\left(-\|\mathbf{E}_{\mathrm{PCA}}\|^2_2\right). \tag{10}$$

Equation (10) is a global estimation method. Occlusion will bring a large error in the global method. In this paper, a patch-based similarity estimation strategy for the PCA subspace model is proposed. First the representation error is divided into patches with the same size in the LSR model, $\mathbf{E}_{\mathrm{PCA}} = [\mathbf{e}^1_{\mathrm{PCA}}\, \mathbf{e}^2_{\mathrm{PCA}} \cdots \mathbf{e}^J_{\mathrm{PCA}}]$. Then, considering the occluded patches signed by the LSR model, the patch-based similarity for the PCA subspace model can be formulated as

$$P_{\mathrm{PCA}}(\mathbf{y}^n|\mathbf{x}^n, \mathbf{g}^n) = \prod_{j=1}^{J} \exp\left(-\left\|\mathbf{e}^j_{\mathrm{PCA}-O}\right\|^2_2\right) \tag{11}$$

where

$$\mathbf{e}^j_{\mathrm{PCA}-O} = \begin{cases} \mathbf{e}_\eta & g_j = 0 \\ \mathbf{e}^j_{\mathrm{PCA}} & g_j = 1 \end{cases}$$

and $\mathbf{e}_\eta$ is an adaptive threshold that is learned from the previous frames of the corresponding image sequences.

Two similarities are separately obtained from the LSR model and the PCA subspace model, and they are fused to create the final similarity for each candidate via a decision fusion strategy [29]. To make that the two similarities have the same weight in the fusion, the two similarities are separately normalized

$$P_{\mathrm{LSR}-N}(\mathbf{y}^n|\mathbf{x}^n, \mathbf{g}^n) = \mathrm{norm}(P_{\mathrm{LSR}}(\mathbf{y}^n|\mathbf{x}^n, \mathbf{g}^n)) \tag{12}$$

$$P_{\mathrm{PCA}-N}(\mathbf{y}^n|\mathbf{x}^n, \mathbf{g}^n) = \mathrm{norm}(P_{\mathrm{PCA}}(\mathbf{y}^n|\mathbf{x}^n, \mathbf{g}^n)). \tag{13}$$

In this paper, we select arithmetic mean as the fusion rule. The mean of the two similarities is computed as the final similarity of the candidate

$$P_f(\mathbf{y}^n|\mathbf{x}^n, \mathbf{g}^n) = \frac{1}{2}(P_{\mathrm{LSR}-N}(\mathbf{y}^n|\mathbf{x}^n, \mathbf{g}^n) + P_{\mathrm{PCA}-N}(\mathbf{y}^n|\mathbf{x}^n, \mathbf{g}^n)). \tag{14}$$

The candidate with the maximal final similarity is selected as the tracking result.

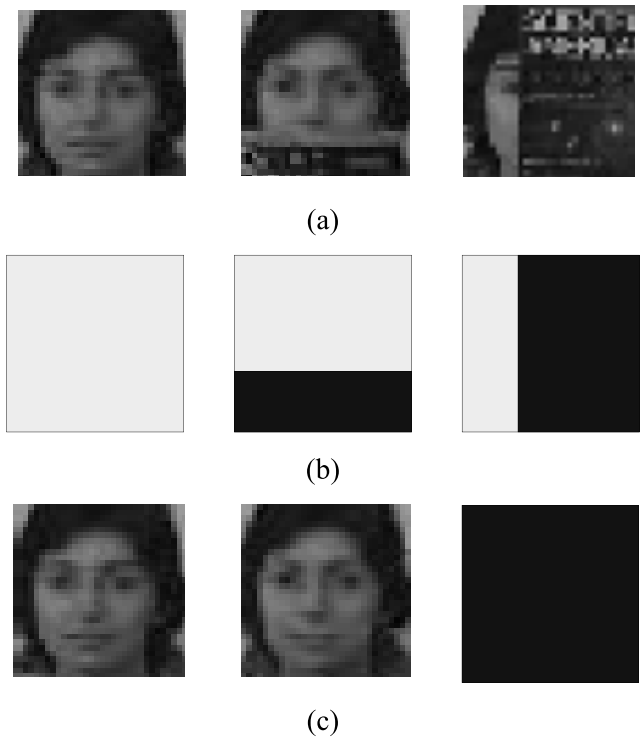


Fig. 3. Some tracking results with corresponding occlusion maps for observation model update. (a) Tracking results. (b) Occlusion maps. (c) New appearances used in update.

C. Observation Model Update

In the sparse model, the dictionary is not updated. The change of the object appearance can be reflected through the sparse coefficients. The object that is manually cropped in the first frame is the most accurate in all the frames. Objects in the later frame may be occluded, and it will bring error if the dictionary is updated with the occluded object. In addition, updating the dictionary is time consuming.

In the PCA subspace model, the mean and eigenbasis are updated with the new object appearances through the IVT method [22]. The object is first measured via the threshold to generate the occlusion map. If the object is occluded, the occluded parts will be replaced with the corresponding parts in the previous frame. Then, the new appearances are used to compute the mean and eigenbasis of the PCA subspace model. If the object is occluded with a large ratio, the observation does not update in the frame. Reconstructing the new appearance breaks the correlation of the patches a little when the object is occluded. Because in the PCA subspace model, the object is represented with the mean and eigenbasis, not the object appearance directly, the break does not affect the updating much. We update the model every five frames, and thus the mean and eigenbasis are computed based on at least five new object appearances. Some tracking samples are shown in Fig. 3 to illuminate the observation model update. Fig. 3(a) shows the tracking result. Fig. 3(b) shows the corresponding occlusion map, and Fig. 3(c) shows the corresponding new appearances used to update the mean and eigenbasis in the PCA subspace model. Without occlusion,

the tracking result is directly used as the new appearance for the first image in Fig. 3(c). The second image in Fig. 3(c) is reconstructed with the tracking result and the previous appearance that is not occluded. Because the object is occluded too much, the third image in Fig. 3(c) does not take part in the update.

D. Patch-Based PCA–LSR Tracking in Particle Filter

The patch-based PCA–LSR method tracks the object through the particle filter. First, candidates in the frame are generated in the particle filter. Then, each candidate is represented through the PCA subspace model and the LSR model separately. After estimating the final similarities of all the candidates via fusing the two models, the candidate with the maximal final similarity is selected as the tracking result. The complete patch-based PCA–LSR tracking method is detailed in Algorithm 1.

Algorithm 1 Patch-Based PCA–LSR Tracking Method

1. Initializing: crop the object manually in the first frame; for the previous 10 frames, track object through matching template; in all the patches of the previous 10 frames, select the max e^j_{LSR} as η, and select the max $\mathbf{e}^j_{\text{PCA}}$ as $\mathbf{e}_\eta$.
2. For the tth frame, generate candidates $\mathbf{x}^n$ $(1 \le n \le N)$ in the particle filter.
3. Represent each candidate via the PCA subspace model and the LSR model separately.
4. Compute the occlusion map $\mathbf{g}^n$ for each candidate $\mathbf{x}^n$
$$g_j = \begin{cases} 0 & e^j_{LSR} > \eta \quad (occlusion\ patch) \\ 1 & \text{other}wise \quad (object\ patch); \end{cases}$$
5. Estimate the similarity of each candidate from the two models
$$P_{LSR}\left(\mathbf{y}^n \left| \mathbf{x}^n, \mathbf{g}^n \right.\right) = \prod_{j=1}^{J} \exp\left(-e^j_{LSR-O}\right)$$
$$P_{PCA}\left(\mathbf{y}^n \left| \mathbf{x}^n, \mathbf{g}^n \right.\right) = \prod_{j=1}^{J} \exp\left(-\left\|\mathbf{e}^j_{PCA-O}\right\|_2^2\right).$$
6. Normalize the two similarities
$$P_{LSR-N}\left(\mathbf{y}^n \left| \mathbf{x}^n, \mathbf{g}^n \right.\right) = P_{LSR}\left(\mathbf{y}^n \left| \mathbf{x}^n, \mathbf{g}^n \right.\right) \Big/ \left\| P_{LSR}\left(\mathbf{y} \left| \mathbf{x}, \mathbf{g} \right.\right)\right\|_2$$
$$P_{PCA-N}\left(\mathbf{y}^n \left| \mathbf{x}^n, \mathbf{g}^n \right.\right) = P_{PCA}\left(\mathbf{y}^n \left| \mathbf{x}^n, \mathbf{g}^n \right.\right) \Big/ \left\| P_{PCA}\left(\mathbf{y} \left| \mathbf{x}, \mathbf{g} \right.\right)\right\|_2;$$
7. Fuse the two similarities
$$P_f\left(\mathbf{y}^n \left| \mathbf{x}^n, \mathbf{g}^n \right.\right) = \frac{1}{2}\left(P_{LSR-N}\left(\mathbf{y}^n \left| \mathbf{x}^n, \mathbf{g}^n \right.\right) + P_{PCA-N}\left(\mathbf{y}^n \left| \mathbf{x}^n, \mathbf{g}^n \right.\right)\right)$$
and select the candidate with the maximal final similarity as the tracking result
$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}^n} P_f\left(\mathbf{y}^n \left| \mathbf{x}^n, \mathbf{g}^n \right.\right);$$
8. Update the observation model through computing the mean and eigenbasis based on the new object appearances.
9. Go to Step 2 and track object in the next frame.

V. Experiments

In this section, the proposed method is first tested with different patch sizes to achieve the best tracking performance in terms of center location error based on the ground truth. Then, the proposed method is compared with six state-of-the-art methods on eight challenging image sequences.

A. Tracking Performance Test

In the proposed method, each object observation is normalized to 32×32 pixels. In each frame, 600 particles are generated as the candidates. For the PCA subspace model, 16 eigenvectors are used. For the LSR model, λ in (6) enforces the sparsity of the solution, and bigger λ will make the solution be sparser. We directly fix the weight parameter λ to be 0.01 as in [23], which is approved by the experiments. In the previous 10 frames of every image sequence, there is no occlusion on the target. The objects are divided into patches. The residual error e^j_{LSR} of the patch $\mathbf{z}^j_c$ in the LSR model and the representation error $\mathbf{e}^j_{\text{PCA}}$ in the PCA subspace model are computed. In all the patches of the previous 10 frames, the max e^j_{LSR} is selected as η, and the max $\mathbf{e}^j_{\text{PCA}}$ is selected as $\mathbf{e}_\eta$ in the corresponding image sequence. In the PCA subspace model, the mean and eigenbasis are updated with new object appearances every five frames.

Since the proposed method is operated on local patches, the patch size greatly affects the tracking performance. The tracking performance of the proposed method with different patch sizes is tested, and the related qualitative and quantitative results are shown in Fig. 4. The test is implemented on the image sequence of face *Occlusion 1* [11] with occlusion occurring in most of the frames. As can be observed in the qualitative results in Fig. 4(a), the method with the patch in 16×16 pixels began to drift in the 233 frames, and then lost the object due to the occlusion. The method with a patch size of 8×8 tracked the object more accurately than those with other patch sizes, especially when the face is occluded.

The quantitative evaluated performances in terms of center location error and overlap ratio based on the ground truth [30] are shown in Fig. 4(b) and (c), respectively. The center location error d is computed as $d = ((l_{x-T} - l_{x-G})^2 + (l_{y-T} - l_{y-G})^2)^{1/2}$, where (l_{x-T}, l_{y-T}) and (l_{x-G}, l_{y-G}) are the object center locations of the tracking result and ground truth. The smaller center location error denotes the higher tracking performance. The overlap ratio r is computed by the intersection over union based on the tracking result R_T and the ground truth R_G as $r = (R_T \cap R_G / R_T \cup R_G)$. The higher overlap ratio stands for the higher tracking performance. It can be observed that the method with the patch in 8×8 pixels performs better than the methods with other patch sizes in both terms of center location error and overlap rate.

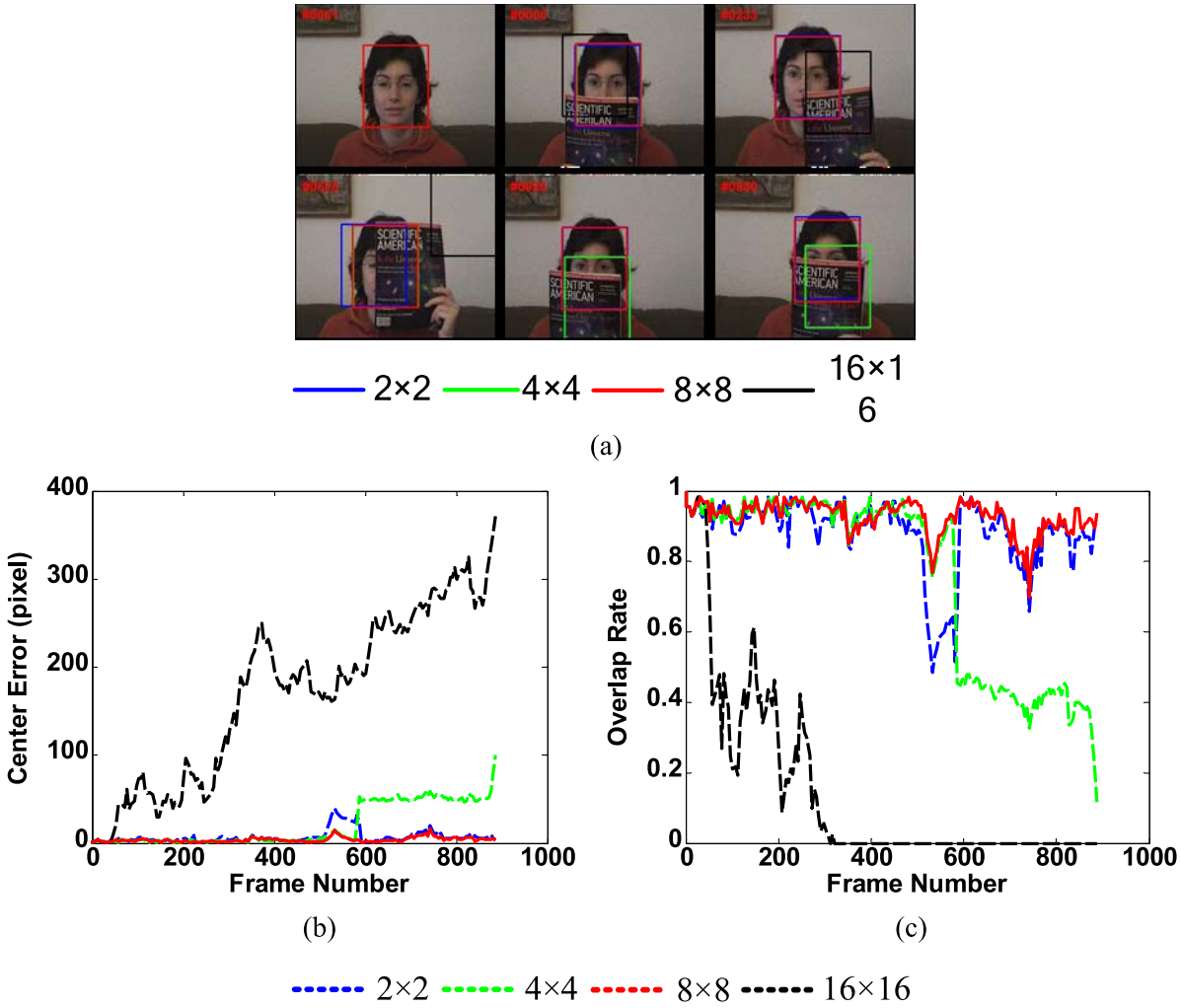


Fig. 4. Tracking performance of the proposed method with different patch sizes. (a) Qualitative evaluated performance. (b) Quantitative evaluated performance in terms of center location error. (c) Quantitative evaluated performance in terms of overlap rate.

The proposed method is implemented in MATLAB 7.10 on a 2.1-GHz i3-2310M Core computer with 4-GB memory. The proposed method with patch in 8×8 pixels runs at 1.8 frames/s.

B. Comparison With Other Methods

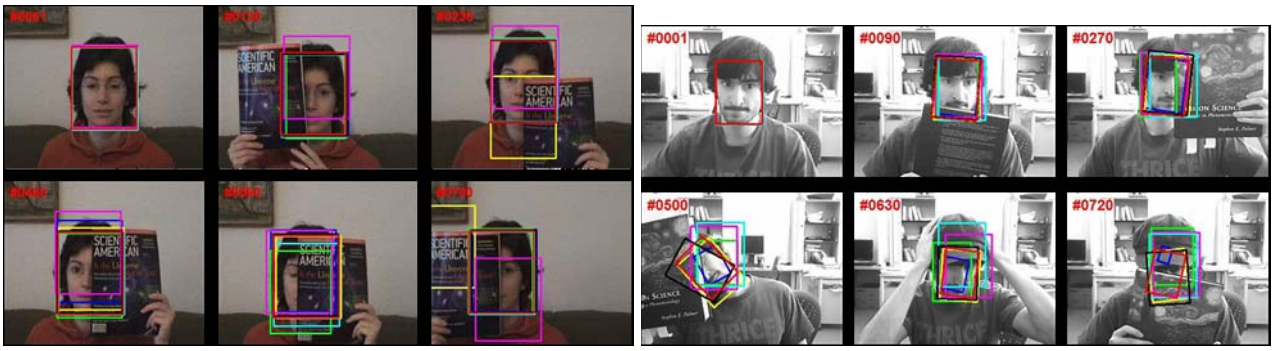
Eight image sequences with different challenging situations are selected to compare the tracking performance of the methods. The situations include heavy occlusion, illumination variation, motion blur, in-plane and out-of-plane rotations, scale change, and background clutter. The image sequences and challenging factors are listed in Table I. The eight image sequences are all selected from [13] and [23], and they are all downloadable.

The proposed method is compared with six state-of-the-art methods including the IVT [22], l_1 [5], Frag [11], multiple instance learning (MIL) [14], OSPT [13], and ASLST [26]. The methods are implemented using the source codes and parameters provided by the authors for fair comparisons. The object target is initialized in the first frame according to the ground truth. The comparison results of qualitative evaluated tracking performance are shown in Figs. 5–8.

TABLE I
TEST IMAGE SEQUENCES

Sequence	# Frames	Challenging Factors
Occlusion 1	898	partial occlusion
Occlusion 2	819	partial occlusion, in-plane rotation
David Indoor	462	illumination variations, scale changes, out-plane rotation
Car 11	393	illumination variations, scale changes
Deer	71	motion blur
Jumping	313	motion blur, background clutter
Lemming	1336	background clutter, scale change, partial occlusion, out-plane rotation
Stone	593	background clutter, partial occlusion

Fig. 5 shows the tracking performance of the methods against object partial occlusion. As can be observed

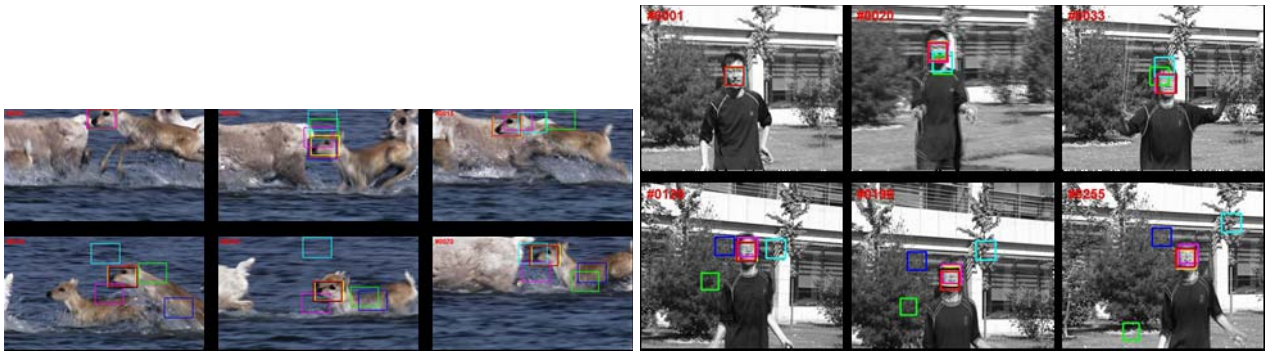


(a) (b)

Fig. 5. Tracking performance comparison on qualitative evaluation against partial occlusion. (a) *Occlusion 1*. (b) *Occlusion 2*. —IVT, —l_1, —Frag, —MIL, —OSPT, —ASLST, and —our method.

(a) (b)

Fig. 6. Tracking performance comparison on qualitative evaluation against illumination variation. (a) *David Indoor*. (b) *Car11*. —IVT, —l_1, —Frag, —MIL, —OSPT, —ASLST, and —our method.



(a) (b)

Fig. 7. Tracking performance comparison on qualitative evaluation against motion blur. (a) *Deer*. (b) *Jumping*. —IVT, —l_1, —Frag, —MIL, —OSPT, —ASLST, and —our method.

in Fig. 5(a), the ASLST method has absolutely lost the object after the object is heavy occluded for a long time (e.g., #0700). The IVT method is so sensitive to the occlusion that the tracker drifts much in some frames (e.g., #0500, #0630, and #0720). Through the patch-based similarity estimation strategy, the proposed method can still track the object well when occlusion occurs.

Fig. 6 shows the tracking performance of the methods against illumination variations. The *David Indoor* sequence nearly includes all challenging factors, such as illumination variation, scale change, partial occlusion, motion blur, in-plane rotation, and out-plane rotation. Because of the illumination variation, the methods of l_1, Frag, and MIL often lose the object target (e.g., #0200 and #0400 in the *David Indoor* sequence and #0250 and #0370 in the *Car 11* sequence). Integrating the advantage of the PCA subspace model and LSR model, the proposed method is more robust than other methods in handling almost all the challenging factors, especially in Fig. 6(a).

(a) (b)

Fig. 8. Tracking performance comparison on qualitative evaluation against background clutter. (a) *Lemming*. (b) *Stone*. —IVT, —l_1, —Frag, —MIL, —OSPT, —ASLST, and —our method.

TABLE II
AVERAGE CENTER ERROR OF TRACKING METHODS (IN PIXELS)

	IVT [22]	l_1 [5]	Frag [11]	MIL [14]	OSPT [13]	ASLST [26]	Our method
Occlusion 1	9.1	6.5	5.6	32.2	**4.7**	41.5	**3.8**
Occlusion 2	10.2	11.1	15.4	14.0	**4.0**	**4.0**	**3.4**
David Indoor	3.6	7.6	76.7	16.1	**3.7**	3.8	**3.0**
Car 11	2.1	33.2	63.9	43.5	2.2	**2.0**	**1.7**
Deer	127.5	171.5	92.1	66.5	8.5	**5.2**	**8.4**
Jumping	36.8	92.4	58.4	9.9	**5.0**	**5.2**	5.9
Lemming	93.4	184.8	149.1	25.6	**9.1**	76.9	**11.6**
Stone	2.2	19.2	65.8	32.3	**1.6**	2.0	**1.8**

Fig. 7 shows the tracking performance of the methods against motion blur. It can be observed that due to object and camera motion, the objects in image sequence are very blur. In this situation, the l_1 method performs worst because it cannot represent the object in blur well. The IVT method handles the motion blur better than the SR model, but it also fails without the constraint of the SR model (e.g., #0129, #0196, and #0255 in the *Jumping* sequence). The OSPT, ASLST, and the proposed method combine the PCA subspace model and SR model, and they track the object more accurately than other methods.

Fig. 8 shows the tracking performance of the methods against background clutter. The sequences in Fig. 8 also include the challenging factors of scale change, partial occlusion, motion blur, and out-plane rotation. In complex background, the Frag-Track method first loses the object in Fig. 8(a) (e.g., #0100 in the *Lemming* sequence). With the increasing drift, the ASLST method also fails in the tracking (e.g., #0749 in the *Lemming* sequence). In Fig. 8(b), a piece of stone is tracked as the target, which is initialized in the first frame. Because of the similarity between the target and background, the methods of l_1, Frag, and MIL all bring large errors. In the *Lemming* sequence in which the object target is often rotated out-plane, the OSPT method achieves the best tracking performance, and the proposed method gets the second one. This is because the global method of OSPT is more robust to the out-plane rotation, compared with the local method. Therefore, incorporating the global strategy into our framework is part of our ongoing work.

The performance of tracking methods is also evaluated on quantity in terms of center location error and overlap ratio based on the ground truth. The center location errors of the seven methods on eight challenging image sequences are shown in Fig. 9, and the overlap ratios are shown in Fig. 10.

The average center location errors and average overlapping ratios of the seven methods on eight challenging image sequences are listed in Tables II and III. The numbers indicating high tracking performance are labeled with bold. In Table II, the proposed method obtains the best performance in four image sequences and the second best performance in other three image sequences. The proposed method can handle occlusion better than the other tracking methods. In Table III, it can be observed that the OSPT method tracks the object target more accurately than other methods in terms of overlap rate.

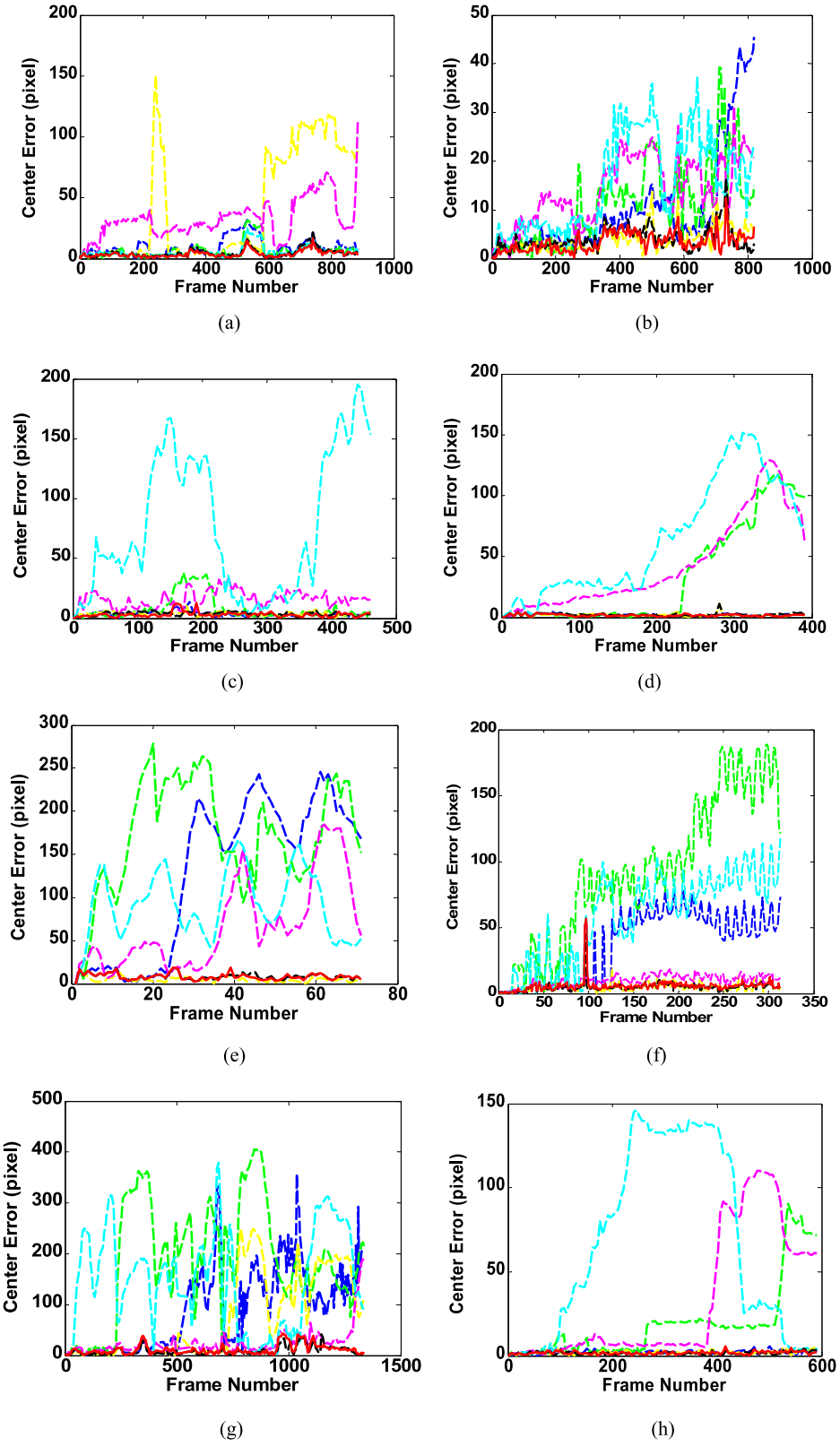


Fig. 9. Quantitative evaluation in terms of center location error (in pixels). (a) *Occlusion 1*. (b) *Occlusion 2*. (c) *David Indoor*. (d) *Car11*. (e) *Deer*. (f) *Jumping*. (g) *Lemming*. (h) *Stone*. ----IVT, ----l_1, ----Frag, ----MIL, ----OSPT, ----ASLST, and ——our method.

The time of the proposed method is also evaluated and compared with that of other methods. The results are listed in Table IV. Because the source codes provided by the authors are implemented through C++ in the methods of Frag [11] and MIL [14], the two methods do not take part in the comparison for fairness. The IVT method is the fastest, and

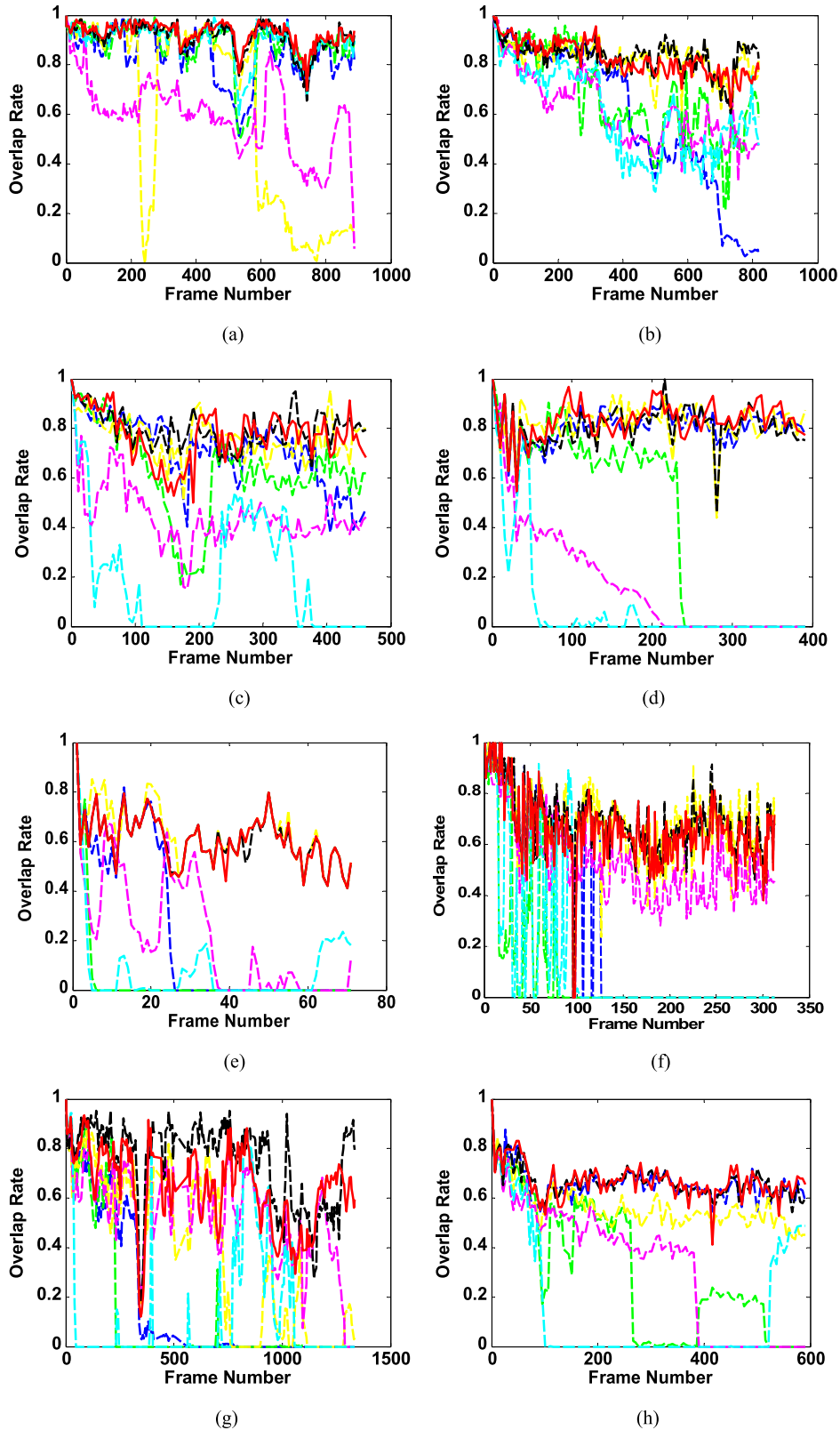


Fig. 10. Quantitative evaluation in terms of overlap ratio. (a) *Occlusion 1*. (b) *Occlusion 2*. (c) *David Indoor*. (d) *Car11*. (e) *Deer*. (f) *Jumping*. (g) *Lemming*. (h) *Stone*. ----IVT, ----l_1, ----Frag, ----MIL, ----OSPT, ----ASLST, and ----our method.

it only need 0.09 s to deal with one frame. Our method is a little slower than other methods due to the patch-based operation.

C. Fusion Rule Comparison

Fusion rules include arithmetic mean, geometric mean, maximum, and so on. The fusion rules are compared in

TABLE III
AVERAGE OVERLAP RATE OF TRACKING METHODS

	IVT [22]	l_1 [5]	Frag [11]	MIL [14]	OSPT [13]	ASLST [26]	Our method
Occlusion 1	0.84	0.87	0.89	0.59	**0.91**	0.60	**0.92**
Occlusion 2	0.58	0.67	0.60	0.61	**0.84**	0.82	**0.83**
David Indoor	0.71	0.63	0.19	0.45	**0.80**	0.76	**0.77**
Car 11	0.81	0.44	0.09	0.17	0.81	**0.82**	**0.84**
Deer	0.22	0.04	0.08	0.21	**0.61**	**0.63**	**0.61**
Jumping	0.28	0.09	0.14	0.53	**0.69**	**0.68**	0.66
Lemming	0.18	0.13	0.13	0.53	**0.75**	0.37	**0.64**
Stone	0.65	0.29	0.15	0.32	**0.66**	0.56	**0.66**

TABLE IV
AVERAGE TRACKING TIME OF DIFFERENT METHODS

Methods	IVT [22]	l_1 [5]	OSPT [13]	ASLST [26]	Our method
Time (seconds /frame)	0.09	0.53	0.48	0.43	0.55

TABLE V
AVERAGE CENTER ERROR OF TRACKING METHODS WITH DIFFERENT FUSION RULES (IN PIXELS)

	LPCA	LSR	Arithmetic mean	Geometric mean	Maximum
Occlusion 1	3.69	3.86	3.84	3.69	3.69
Occlusion 2	3.44	3.69	3.43	3.15	3.44
David Indoor	3.03	56.43	3.03	4.21	3.03
Car 11	1.67	1.95	1.67	1.61	1.67
Deer	8.35	81.64	8.38	8.23	8.35
Jumping	5.89	68.93	5.88	5.53	5.88
Lemming	12.10	125.52	11.59	68.42	12.10
Stone	1.78	30.19	1.76	28.50	1.78

this section. The fusion rule of arithmetic mean can be formulated as (14). The rules of geometric mean and maximum can be separately formulated as follows:

$$P_{f\_g}(\mathbf{y}^n|\mathbf{x}^n,\mathbf{g}^n) = \sqrt{(P_{\mathrm{LSR}-N}(\mathbf{y}^n|\mathbf{x}^n,\mathbf{g}^n) \times P_{\mathrm{PCA}-N}(\mathbf{y}^n|\mathbf{x}^n,\mathbf{g}^n))} \quad (15)$$

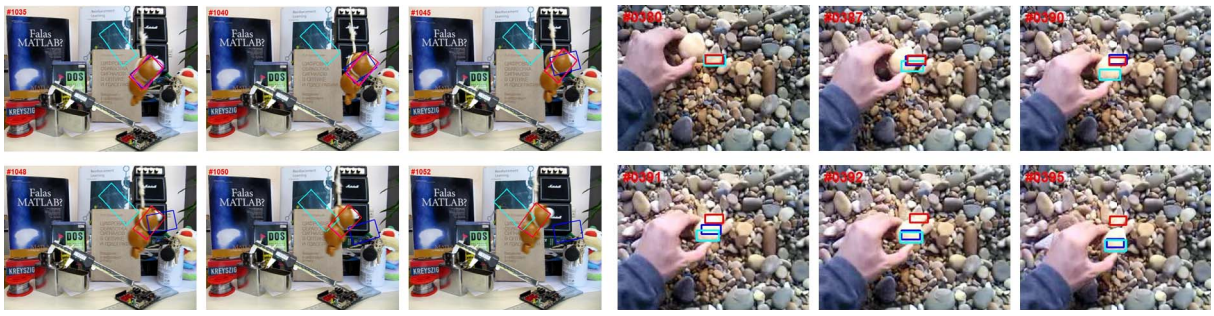
$$P_{f\_m}(\mathbf{y}^n|\mathbf{x}^n,\mathbf{g}^n) = \max(P_{\mathrm{LSR}-N}(\mathbf{y}^n|\mathbf{x}^n,\mathbf{g}^n),\, P_{\mathrm{PCA}-N}(\mathbf{y}^n|\mathbf{x}^n,\mathbf{g}^n)). \quad (16)$$

The fusion rules are compared in the object tracking with different challenging situations. The two single methods taking part in the fusion are separately named local principal component analysis (LPCA) and LSR. Both the LPCA method and LSR method track the object through the representation error in patches. The tracking results via the different fusion rules are listed in Tables V and VI.

It can be observed that through measuring the representation error in patches, the tracking performance of the LPCA method is much higher than that of the IVT method in [22]. However, the LSR method can not track the object well only through the representation error, and it lost the target in the image sequences of *David Indoor*, *Deer*, *Jumping*, *Lemming*, and *Stone*. The two single methods are fused through the rules of arithmetic mean, geometric mean,

TABLE VI
AVERAGE OVERLAP RATE OF TRACKING METHODS WITH DIFFERENT FUSION RULES

	LPCA	LSR	Arithmetic mean	Geometric mean	Maximum
Occlusion 1	0.92	0.92	0.92	0.92	0.92
Occlusion 2	0.82	0.81	0.82	0.82	0.82
David Indoor	0.77	0.22	0.77	0.75	0.77
Car 11	0.83	0.73	0.83	0.83	0.83
Deer	0.60	0.30	0.60	0.60	0.60
Jumping	0.65	0.10	0.65	0.66	0.65
Lemming	0.64	0.20	0.64	0.50	0.64
Stone	0.66	0.27	0.66	0.44	0.66



(a) (b)

Fig. 11. Tracking results of methods with different fusion rules. (a) *Lemming*. (b) *Stone*. —LSR, —LPCA, —arithmetic mean, —geometric mean, and —maximum.

and maximum, respectively. The tracking method with the arithmetic mean fusion rule obtains the best performance, and it is the most robust against all the challenging situations. The tracking method with the geometric mean fusion rule obtains smaller average center errors than the method with the arithmetic mean fusion rule in the image sequences of *Occlusion 1*, *Occlusion 2*, *Car 11*, *Deer*, and *Jumping*, but it lost the targets in the image sequences of *Lemming* and *Stone*. The performance of the tracking method with the maximum fusion rule mainly depends on the LPCA method. In Table V, it can be found that the fusion rule of geometric mean helps to improve the tracking accuracy, and the fusion rule of arithmetic mean benefits improving the robustness of the tracing method. The fusion methods improve the tracking performance than the single methods only a little in pixels, but the improvement is very important. In the tracking, the drift will be accumulated, and this can be observed in Fig. 11.

Fig. 11 shows some tracking results of methods with different fusion rules. The tracking results of the fusion method with arithmetic mean (red box) are always correct in the two image sequences, and they are more accurate than those of the fusion method with maximum. However, the fusion method with geometric mean (blue box) lost the targets in the two image sequences because of the drift accumulation. In the *Lemming* image sequence, the target rotated out-plane, and in the *Stone* image sequence, the target was occluded by other stone with a large ratio. In these two situations, the LSR method cannot track the target correctly, and after multiplication operation, the fusion method with geometric mean lost the target.

VI. CONCLUSION

In this paper, a new robust object tracking method is proposed based on PCA and LSR. The advantages of the PCA subspace model and LSR model are integrated to handle the partial occlusion, illumination variation, motion blur, and background clutter in the tracking. In the PCA subspace model, to alleviate sensitivity to occlusion, the representation error of PCA is divided into patches. The similarity for the PCA subspace model is computed in patches based on the occlusion map that is obtained from the residual error of the LSR model. The two similarities from the PCA subspace model and the LSR model are fused to make decision. The experiments demonstrate that the proposed method performs better than several state-of-the-art methods on challenging image sequences.

ACKNOWLEDGMENT

The authors would like to thank the editors and all of the anonymous reviewers for their constructive suggestions that greatly improved this paper.

REFERENCES

[1] Y. Yuan, J. Fang, and Q. Wang, "Robust superpixel tracking via depth fusion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 1, pp. 15–26, Jan. 2014.

[2] A. Bal and M. S. Alam, "Automatic target tracking in FLIR image sequences using intensity variation function and template modeling," *IEEE Trans. Instrum. Meas.*, vol. 54, no. 5, pp. 1846–1852, Oct. 2005.

[3] L. Xu, Y. Yan, S. Cornwell, and G. Riley, "Online fuel tracking by combining principal component analysis and neural network techniques," *IEEE Trans. Instrum. Meas.*, vol. 54, no. 4, pp. 1640–1645, Aug. 2005.

[4] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Comput. Surv.*, vol. 38, no. 4, Dec. 2006, Art. ID 13.

[5] X. Mei and H. Ling, "Robust visual tracking and vehicle classification via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 11, pp. 2259–2272, Nov. 2011.

[6] G. Paravati, A. Sanna, B. Pralio, and F. Lamberti, "A genetic algorithm for target tracking in FLIR video sequences using intensity variation function," *IEEE Trans. Instrum. Meas.*, vol. 58, no. 10, pp. 3457–3467, Oct. 2009.

[7] A. Y. Ng and M. I. Jordan, "On discriminative vs. generative classifiers: A comparison of logistic regression and naive Bayes," in *Proc. NIPS*, vol. 14. 2002, pp. 841–848.

[8] K. Zhang, L. Zhang, and M.-H. Yang, "Real-time compressive tracking," in *Proc. 12th Eur. Conf. Comput. Vis.*, 2012, pp. 864–877.

[9] S. Avidan, "Ensemble tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 2, pp. 261–271, Feb. 2007.

[10] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 5, pp. 564–575, May 2003.

[11] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2006, pp. 798–805.

[12] F. Chen, Q. Wang, S. Wang, W. Zhang, and W. Xu, "Object tracking via appearance modeling and sparse representation," *Image Vis. Comput.*, vol. 29, no. 11, pp. 787–796, Oct. 2011.

[13] D. Wang, H. Lu, and M.-H. Yang, "Online object tracking with sparse prototypes," *IEEE Trans. Image Process.*, vol. 22, no. 1, pp. 314–325, Jan. 2008.

[14] B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 983–990.

[15] F. Lamberti, A. Sanna, and G. Paravati, "Improving robustness of infrared target tracking algorithms based on template matching," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 47, no. 2, pp. 1467–1480, Apr. 2011.

[16] B. Liu, L. Yang, J. Huang, P. Meer, L. Gong, and C. Kulikowski, "Robust and fast collaborative tracking with two stage sparse optimization," in *Proc. 11th Eur. Conf. Comput. Vis.*, 2010, pp. 624–637.

[17] B. Liu, J. Huang, C. Kulikowski, and L. Yang, "Robust visual tracking using local sparse appearance model and K-selection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 12, pp. 2968–2981, Dec. 2013.

[18] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja, "Robust visual tracking via multi-task sparse learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2042–2049.

[19] X. Wang, N. D. Georganas, and E. M. Petriu, "Fabric texture analysis using computer vision techniques," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 1, pp. 44–56, Jan. 2011.

[20] L. Sun and G. Liu, "Visual object tracking based on combination of local description and global representation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 4, pp. 408–420, Apr. 2011.

[21] J. Krabicka, G. Lu, and Y. Yan, "Profiling and characterization of flame radicals by combining spectroscopic imaging and neural network techniques," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 5, pp. 1854–1860, May 2011.

[22] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *Int. J. Comput. Vis.*, vol. 77, nos. 1–3, pp. 125–141, May 2008.

[23] W. Zhong, H. Lu, and M.-H. Yang, "Robust object tracking via sparse collaborative appearance model," *IEEE Trans. Image Process.*, vol. 23, no. 5, pp. 2356–2368, May 2014.

[24] F. Yang, H. Lu, and M.-H. Yang, "Learning structured visual dictionary for object tracking," *Image Vis. Comput.*, vol. 31, no. 12, pp. 992–999, Dec. 2013.

[25] B. Zhuang, H. Lu, Z. Xiao, and D. Wang, "Visual tracking via discriminative sparse similarity map," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1872–1881, Apr. 2011.

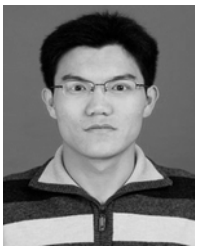
[26] X. Jia, H. Lu, and M.-H. Yang, "Visual tracking via adaptive structural local sparse appearance model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1822–1829.

[27] Y. Li, H. Ai, T. Yamashita, S. Lao, and M. Kawade, "Tracking in low frame rate video: A cascade particle filter with discriminative observers of different life spans," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 10, pp. 1728–1740, Oct. 2008.

[28] Y. Zhai, M. B. Yeary, S. Cheng, and N. Kehtarnavaz, "An object-tracking algorithm based on multiple-model particle filtering with state partitioning," *IEEE Trans. Instrum. Meas.*, vol. 58, no. 5, pp. 1797–1809, May 2009.

[29] R. Huan and Y. Pan, "Decision fusion strategies for SAR image target recognition," *IET Radar, Sonar, Navigat.*, vol. 5, no. 7, pp. 747–755, Aug. 2011.

[30] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jul. 2010.



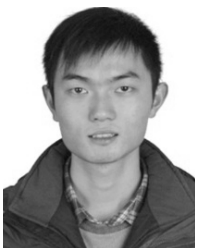
Haicang Liu received the B.S. and M.S. degrees from the College of Electrical and Information Engineering, Hunan University, Changsha, China, in 2006 and 2011, respectively, where he is currently pursuing the Ph.D. degree in image processing.

His current research interests include information fusion, sparse representation, pattern recognition, target detection, and tracking.

Shutao Li (M'07–SM'15) received the B.S., M.S., and Ph.D. degrees in electrical engineering from Hunan University, Changsha, China, in 1995, 1997, and 2001, respectively.

He joined the College of Electrical and Information Engineering, Hunan University, in 2001. He was a Research Associate with the Department of Computer Science, Hong Kong University of Science and Technology, Hong Kong, in 2001. From 2002 to 2003, he was a Post-Doctoral Fellow with Royal Holloway College, University of London, London, U.K., with Prof. J. Shawe-Taylor. In 2005, he visited the Department of Computer Science, Hong Kong University of Science and Technology, as a Visiting Professor. He is currently a Full Professor with the College of Electrical and Information Engineering, Hunan University. He has authored or co-authored over 160 refereed papers. His current research interests include compressive sensing, sparse representation, image processing, and pattern recognition.

Dr. Li is an Associate Editor of the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, and a member of the Editorial Board of the *Information Fusion* journal and *Sensing and Imaging*. He was a recipient of two Second-Grade National Awards at the Science and Technology Progress of China in 2004 and 2006.



Leyuan Fang (S'10–M'14) received the B.S. degree in electrical engineering from the Hunan University of Science and Technology, Xiangtan, China, in 2008.

He joined the College of Electrical and Information Engineering, Hunan University, Changsha, China, in 2008, for the Ph.D. degree program. Since 2011, he has been a Visiting Ph.D. Student with the Department of Ophthalmology, Duke University, Durham, NC, USA, supported by the China Scholarship Council. His current research interests include sparse representation and multiresolution analysis in remote sensing and medical image processing.

Mr. Fang was a recipient of the Scholarship Award for Excellent Doctoral Student by the Chinese Ministry of Education in 2011.