Pattern Recognition 48 (2015) 2885–2905

Contents lists available at ScienceDirect

Pattern Recognition

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

Real-time and robust object tracking in video via low-rank coherency analysis in feature space

Chenglizhao Chen [a], Shuai Li [a,*], Hong Qin [b], Aimin Hao [a]

[a] State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, China
[b] Stony Brook University SUNY, United States

ARTICLE INFO

Article history:
Received 30 July 2014
Received in revised form
7 November 2014
Accepted 27 January 2015
Available online 13 February 2015

Keywords:
Localized compressive sensing representation
Fast lowrank approximation
Lowrank coherency tracking
Visual tracking

ABSTRACT

Object tracking in video is vital for security surveillance, pattern and motion recognition, traffic control, augmented reality, human-computer interaction, etc. Despite the rapid growth of various techniques in recent years, certain technical challenges still exist in terms of efficiency, accuracy, and robustness. To ameliorate, this paper suggests a novel video object tracking approach by first collecting both local and global information from consecutive video observations (i.e., frames) and then exploring the low-rank coherency in the accompanying feature space of targeting objects, which enables real-time and robust object tracking in video while combating certain technical difficulties due to occlusion, deformation, transient illumination, rapid movement, and scale change. Our central idea is to integrate local space-distinctive candidate features and global time-continuous target coherency into a smart low-rank analysis model. For local candidate representation, we propose a simple yet efficient patch-level feature descriptor based on compressive sensing, which is directly derived from the frame color distribution available from video frames. Building upon this powerful local representation, we further organize all the candidates in the frame cache and the yet-to-be-processed new frame to form a space-time feature set, we then employ the low-rank decomposition to enable global coherency voting. Since the low-rank coherency implies the intrinsic co-occurring parts of different target observations, robust tracking can be achieved by employing this principle as the matching criterion even for objects with drastically varying appearance. Furthermore, we progressively incorporate the prior-frames' tracking results into the low-rank approximation in the current frame, which can greatly reduce the most time-consuming computation and guarantee real-time performance. We conduct extensive experiments on several well-known yet challenging benchmarks, and make comprehensive and quantitative evaluations with state-of-the-art methods. All the results demonstrate the superiority of our method in terms of accuracy, efficiency, robustness, and versatility.

1. Introduction and motivation

Visual tracking is still one of the most active research areas in computer vision and pattern recognition, which is extremely valuable in many applications, including surveillance [1], traffic control [2], motion recognition [3], etc. Although visual tracking research has gained great momentum and has achieved significant successes in recent years, it still remains challenging when the goal is to robustly track the target object with high-varying inter-frame appearance and occasional/frequent occlusion in real-time [4]. Generally speaking, current research methods of visual tracking can be roughly categorized into two groups: discriminative tracking [5–9] and generative tracking [10–14].

Discriminative tracking customarily employs binary classification to separate the target object from its background, wherein numerous training samples from the tracking results of previous frames are indispensable to the individual classifiers of current observation [6]. However, in order to achieve accurate and occlusion-invariant tracking, most of the discriminative methods [15,16] avoid updating their classifiers that are varying far away from the initial setting [17], as a result, drift occurs unavoidably when the target object undergoes heavy scale or rapid appearance change.

Unlike discriminative tracking methods, generative tracking methods usually resort to certain appearance models to depict the object-specific observation, and take the candidate having the best compatibility with the appearance model as the tracked object [14,18]. In theory, the generative tracking methods can accommodate any complex appearance variation at the expense of extra computational burden by continuing to enhance the description capacity for the partial appearance model. Hence, for appearance model with limited capacity, the target's representation together with its

* Corresponding author.
E-mail address: lishuai@buaa.edu.cn (S. Li).

http://dx.doi.org/10.1016/j.patcog.2015.01.025
0031-3203/

similarity-matching criteria are the vital issues of generative tracking. Inspired by this rationale, generative methods commonly adopt mid-level patch-based solution (e.g., PCA or histogram) to locally represent the target object, and treat the entire reconstruction error [11] or partial observation [18] of the appearance model as matching criteria, which in some sense can make effective tradeoff between versatility enhancement and drift suppression.

In sharp contrast, we explore the compressive sensing and low-rank analysis theory to facilitate the video object tracking in feature space. It should be noted that [8] suggests to represent the tracking object with Haar-like feature formulation based on compressive sensing. However, the Haar-like feature formulation imposes too much emphasis on the global characteristics distribution to handle the occlusion, which further limits its application scope for generative tracking. To respect object's local characteristics, we propose to employ mid-level patch-based integral histogram to represent the candidate target, which can be seen as a relaxed local version of [8] at the expense of possibly compromising discriminative power. Moreover, based on this novel representation, we have found that the principal characteristics distribution of the target object will not change much within the consecutive frames, even though the appearance may have changed drastically. Therefore, it provides enough rationale for us to leverage such coherency for robust object tracking by resorting to low-rank analysis, and moreover, the low-rank coherency extracted from the tracked instance of our appearance model in previous frames will serve as the matching criteria of current-frame candidate targets. Meanwhile, since the principal low-rank information are redundant, it can be synchronously used to govern the dynamic update of the appearance model with limited capacity (e.g., 100 in our experiments). Benefiting from the elegant integration of space-distinctive candidate features and time-continuous coherency, our method can accommodate high-varying appearance and occasional/frequent occlusion. In particular, the salient contributions of our work documented in this paper can be summarized as follows:

- We propose a versatile, real-time, and robust video object tracking method, which can neatly accommodate the object's varying appearances caused by local occlusion, large deformation, transient illumination, rapid movement, drastic scale change, etc.
- We define an efficient yet discriminative patch-based appearance model based on compressive sensing, which can compactly represent the intrinsic characteristics distribution information of the local object parts in a very low dimensional feature space.
- We propose a novel low-rank decomposition based cross-frame coherency analysis model to robustly capture video object that may undergo large appearance variation, and such method can also be used to govern the dynamic update of our appearance model.
- We formulate a series of sparsity-measuring based criteria to accelerate tracking performance, assist appearance model update, and handle occlusion effectively.

2. Related work

Based upon the feature representation and the matching criteria, we further classify the large variety of discriminative and generative tracking methods into global-representation based tracking methods, local-representation based tracking methods, and hybrid tracking methods. Now we briefly review them as follows.

Global-representation based tracking methods: Most of the global-representation based tracking methods usually resort to certain types of color or intensity based histogram for feature representation. Since the histogram implies discriminative color distribution to distinguish the target object from its surroundings, the tracking problem may be converted to a binary classification problem by globally making a decision for boundary [19–21]. Meanwhile, global tracking criteria derived from all previous frames are oftentimes used to facilitate current-frame tracking [22,23]. Thus, such methods give rise to high tracking accuracy and low computational cost [5,7]. Despite some special advantages of the global-representation based tracking methods, several common problems remain to be solved. First, because global representation is sensitive to occlusion, such methods tend to mistakenly consider occlusion as reasonable appearance variation when updating their basic classifiers, which may easily result in tracking drift [6,16]. Second, because the global feature representation is discontinuous in nature, learning based global tracking solutions (or matching criteria) are usually hard to accommodate fast appearance change [24].

Local-representation based tracking methods: Local-representation based tracking methods commonly decompose the target object into many discriminative patches/regions, and employ the patch-to-patch or region-to-region matching strategy to conduct object tracking. For example, Adam et al. [13] represented the target with multiple regular image fragments, which locally describe its different components. Wang et al. [10] represented the target object with the irregular super-pixels based SLIC method [25], and employed clustering based matching criteria together with the special voting or integrating strategy [26] to improve the robustness of object tracking. Although local-representation based tracking methods can well solve the occlusion problem [27], the absence of global spatial-distribution information may weaken the distinguishability of object representation. Therefore, it may at times lead to tracking drift, and the patch-wise matching operation may further deteriorate the tracking result. To combat such limitations, Erdem et al. [12] proposed to represent the target object with higher-level object regions, wherein they adopted the grid-based region representation and searched the target object in a region-to-region manner. Most recently, He et al. [28] used a locally sensitive histogram based region representation to combat the illumination variation, and obtained rather amazing results. Meanwhile, Yao et al. [9] leveraged latent variables based online learning to facilitate the region-weighted representation of target objects, which achieves more robust tracking results. Although local-representation based tracking methods demonstrate their special advantages in handling occlusion and fast partial appearance, however, these methods are still hard to accommodate drastic appearance variation, and the inevitable computational complexity involved in such methods heavily limits their real-time tracking capability (it may be noted that in such cases, FPS $\ll$ 15).

Sparse-representation based tracking methods: Based on the sparse representation (SR) theory, Liu et al. [11] employed the image patches based local representation and global reconstruction error based matching criteria to locate the target object (i.e., local representation with global tracking), wherein the basis functions used for reconstruction are learned from previous-frames' tracking results. Similarly, Zhong et al. [29] introduced a sparsity-based generative model by alternatively formulating sparsity based local feature, and further integrated the global spatial information of each individual patch into an occlusion handing scheme. Xu at el. [14] proposed a sparse coding pool based hybrid representation and taken into account multiple templates during target matching, which achieves improved tracking results. And then, they [18] further proposed an occlusion-handling method by coupling additional noise templates (which is local) with a batch of PCA based individual phototypes (which is global) [15]. Zhang et al. [30] also obtained comparable tracking performance by introducing the low-rank constraint into the formulation of SR basis functions. Recently, Zhang et al. [31] concentrated on localized tracking solution, wherein their SR basis functions are learned within multiple observation constraints. Meanwhile, following the diametrically opposed rationality (global representation with local tracking) of sparse representation, Zhang et al. proposed compressive sensing based global tracking methods [8] via globally representing the target object and

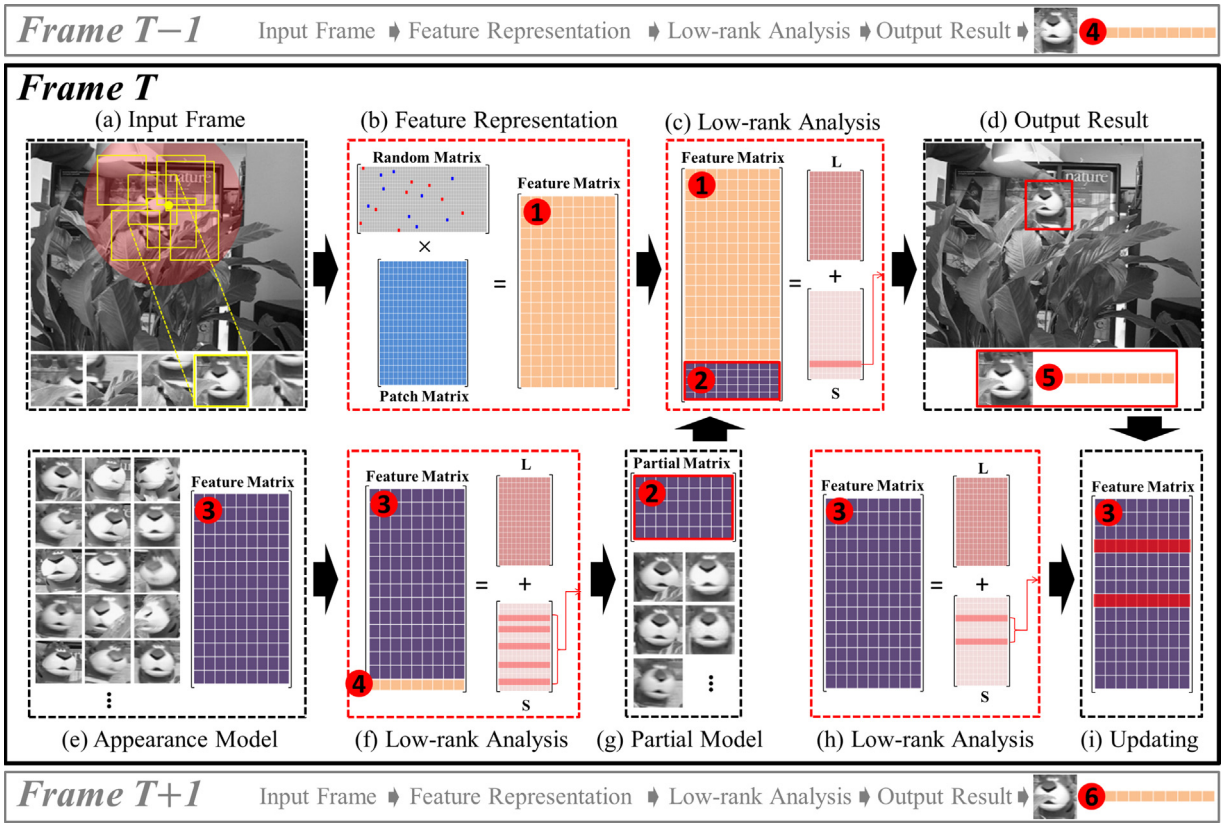

Fig. 1. Architecture of our tracking framework.

dynamically formulating a subgroup of weak classifiers from a classifier pool, which enables discriminative representation, low computational cost, and fast appearance variation. However, in spite of the limited success and popularity of hybrid tracking methods, they are easy to become unstable and cause drift when occlusion and drastic appearance variation occur simultaneously, because such methods are hard to distinguish the occlusion from the appearance variation in sparse feature space. And we will conduct more comprehensive evaluation and discussion on this subject in Section 5.3.

Brief summary: In general, according to the recent comprehensive evaluations performed by Wu et al. [32] and our own experiences, local-representation based tracking methods are more efficient and flexible than the global-representation based ones when handling occlusion. Nevertheless, local-representation based methods are far more likely to cause tracking drift, because the feature space spanned by local representations has less discriminative power than that of global-representation based ones. Strongly inspired by the aforementioned methods, we propose to discriminatively formulate the partial appearance model by collecting both local and global information to form a localized compressive sensing representation, and track the target object by globally exploring the low-rank coherency of the partial appearance model. And the overview of our method is described in the following section.

3. Method overview

As shown in Fig. 1, our method is mainly comprised of two components: compressive sensing based appearance model (Fig. 1(b)), and low-rank decomposition based coherency analysis (Fig. 1(c), (f) and (h)).

Comparing with traditional representation methods, our appearance model has several specific advantages: (1) it enables fast visual tracking in an intrinsic low-dimensional feature space; (2) benefiting from the distance-preserving property, it provides sufficient discriminative power for accurate tracking; (3) it concentrates on the object's local structure representation, which enables robust visual tracking; and (4) it enforces constraints that all the candidates should have strong linear correlation, which naturally gives rise to the subsequent low-rank analysis during frame-wise object tracking and appearance model updating. Fig. 1(a) shows the patch-based candidate targets randomly sampled from the t-th video frame, which are represented by our proposed appearance model and will be further organized in the format of row vector to form a Patch Matrix. Then, we employ compressive sensing based feature representation to transfer the Patch Matrix into low-dimensional Feature Matrix (Fig. 1(b)). Meanwhile, we conduct low-rank analysis (Fig. 1(c)) over the appearance model to obtain the partial appearance model (Fig. 1(f) and (g), and see the detailed discussion in Section 5.4), which will serve as a controller to govern the object tracking (Fig. 1(d)).

Similar to the generic matching problem, the tracking procedure aims to seek the target object with minimal feature distance to the appearance model. Different from the traditional l_1-norm optimization based affinity matrix [33], which measures the differences by element-independently performing unweighed peer-to-peer comparison, our low-rank analysis regards the matching procedure as a procedure of globally seeking the common occurrences [34,35] via weighted region-to-region clustering. The low-rank analysis in Fig. 1(f) aims to globally select a subgroup of feature vectors that are mostly correlated to the previous-frame tracking results. With the obtained partial appearance model (Fig. 1(g)), we further conduct common-parts-biased low-rank analysis to locate the target object (Fig. 1(c), and see the detailed discussion in Section 5.1), and finally we conduct unbiased low-rank analysis to globally determine which elements in the appearance model should be dynamically updated (Fig. 1(h) (i), and see the detailed discussion in Section 5.5).

Table 1
List of the key mathematical symbols used in this paper.

m, n	The dimension of original data, and the dimension of feature space
W, H	The width and height of candidate target rectangle, $m = W \times H$
k	The number of candidate targets (also known as particle numbers)
K	The update strength
q	The power iteration times
M, N	The capacity of appearance model and partial appearance model
$\mathbf{v}$	Observation vector
$\mathbf{p}, \mathbf{P}$	The matching score of our tracking procedure, and the score matrix
$\mathbf{A}, \tilde{\mathbf{A}}$	Appearance model, partial appearance model
$\mathbf{W}, \mathbf{D}$	Voting prior
$\mathbf{O}$	Occlusion mask
$\mathbf{S}$	Sparse matrix obtained by the low-rank decomposition
$\mathbf{L}, \tilde{\mathbf{L}}$	Low-rank matrix, and the low-rank matrix of partial appearance model
$\mathbf{F}$	Candidate pool
$\mathbf{R}, \mathbf{X}, \mathbf{V}$	Measurement matrix, original input matrix, and feature matrix
$\mathbf{A}_*$	Random projection matrix
$\mathbf{Y}_*, \mathbf{Q}_*, \mathbf{R}_*$	Projection matrix, and its QR decomposition: $\mathbf{Y}_* = \mathbf{Q}_*\mathbf{R}_*$

Towards the goal of better assisting the readers to fully understand our mathematical formulations in the following sections, Table (1) summarizes key symbols used in the following mathematical derivations, wherein normal-case letters denote scalars, bold lower-case letters denote finite dimensional vectors, and bold upper-case letters denote matrices.

4. Appearance model and low-rank approximation

4.1. Appearance model based on localized compressive sensing

For real-time visual tracking, one of its typical bottlenecks is the high computational cost of target representation. Despite the simplicity of 2D intensity distribution based object representation, it usually adds great burden on the subsequent tracking procedure due to the involved high dimensional feature space (10^3–10^5). However, as shown in Fig. 2(a), from the perspective of compressive sensing, we can project the high dimensional feature to a stable low dimensional space via Eq. (1), wherein the embedded sparse representation can well preserve the distance distribution of the original feature space:

$$\mathbf{V} = \mathbf{R}\mathbf{X}, \tag{1}$$

where $\mathbf{X} \in \mathbb{R}^m$, $\mathbf{V} \in \mathbb{R}^n$, and $n \ll m$. Meanwhile, the original high-dimensional $\mathbf{X}$ can also be recovered from the sparse signal $\mathbf{V}$ via solving the optimization in Eq. (2) as long as the measurement matrix $\mathbf{R}$ satisfies the Johnson–Lindenstrauss lemma [36]:

$$\min_{\mathbf{X}} \|\mathbf{X}\|_1 \quad \text{s.t. } \|\mathbf{V} - \mathbf{R}\mathbf{X}\|_2 \le \varepsilon, \tag{2}$$

where ε is the predefined error threshold. A typical choice of $\mathbf{R}$ is the random Gaussian matrix ($\mathbf{R} \in \mathbb{R}^{d \times m}, R_{i,j} \sim \mathcal{N}(0,1)$) [37,16,8].

To deal with the scale problem properly, each non-zero entity in $\mathbf{R}$ should be convolved with a rectangle filter (Eq. (3)), and the obtained representation of $\mathbf{X}$ naturally has Haar-like formulation.

$$h_{i,j}(x,y) = \begin{cases} 1, & 1 \le x \le i,\ 1 \le y \le j \\ 0 & \text{otherwise} \end{cases}. \tag{3}$$

Although the principal information distribution of $\mathbf{X}$ can be well preserved in $\mathbf{V}$, the large variation of $\mathbf{V}$ caused by occlusion will inevitably reduce the tracking accuracy. For example, Fig. 2(a) shows that almost 30% elements in $\mathbf{V}$ will be influenced even if only 11% (which is about 1/9) elements of $\mathbf{X}$ are occluded (denoted by yellow color). As shown in Fig. 2(b), increasing non-zero entities (d) in $\mathbf{R}$ indeed contributes to the information preservation of $\mathbf{X}$, however, the variation percentage of the elements in $\mathbf{V}$ will also increase sharply at the same time. In addition, the non-zero entities in $\mathbf{R}$ may also perturb the cross correlations of individual samples due to the convolution in Eq. (3), which will further deteriorate the situation.

To alleviate the above-documented problem, we propose to relax Eq. (1) by introducing two constraints to localize the representation at the expense of possibly losing little information. First, we impose restrictions on the measurement matrix with $\|R_i\|_0 = 1, i \in \{1, 2, \ldots, n\}$, where $\|\cdot\|_0$ indicates the l_0-norm. Then, we redefine the rectangular filter with the constant convolution domain as

$$h_{i,j}(x,y) = \begin{cases} 1, & i \le x \le (i+\gamma W),\ j \le y \le (j+\gamma H) \\ 0 & \text{otherwise} \end{cases}. \tag{4}$$

Here W and H denote the rectangle width and height respectively, and γ is set to be 1/7, which will be further discussed in Section 6.1 in detail. Thus, the feature correlation is more neighbor-dependent while the Haar-like representation remains globally random. From the perspective of random projection, the aggressive measurement number should be larger than $\log(m)$ [38] (e.g., there should be at least 8 non-zero entities (1 or -1) in each row of $\mathbf{R}$ for a 50×50 candidate image window). Our method satisfies this condition, because our $\mathbf{R}$ has $W \times H$ non-zero entities (either all equal to 1 or all equal to -1) after the convolution (Eq. (4)). And the final feature representation $\mathbf{V}$ can be defined as

$$\mathbf{V} = [v_1, v_2, \ldots, v_n]^T = \frac{1}{Z}(\mathbf{R} \odot h)\mathbf{X}. \tag{5}$$

Here $\odot$ denotes the column-wise OR operation, Z is the normalization factor, $\mathbf{V} \in \mathbb{R}^{1 \times n}$, $\mathbf{R} \in \mathbb{R}^{n \times m}$, $\mathbf{X} \in \mathbb{R}^{1 \times m}$, $m = W \times H$, and $n \ll m$.

As a metaphor of visual illustration and imagination, we could summarize our ideas vividly as follows. Traditional compressive sensing can be visualized as "*recognizing a target object globally by wearing many different glasses* (n)", while our method concentrates on "*wearing only 2 glasses to observe different parts of the target object locally*". Besides, compared with other traditional representations, except for high efficiency, our appearance model has another two salient advantages: (1) tiny local appearance variation and affine illumination change can be automatically handled via *MINMAX* normalization on Z, which forces the subsequent tracking procedure to concentrate on global-level matching. (2) Benefiting from the uniform formulation (Fig. 3(e)), the low-rank decomposition (see the detailed discussion in Section 4.2) can be easily approximated within a few iterations.

4.2. Rapid low-rank approximation

Given a $n \times m$ matrix $\mathbf{X}$, the fast rank-r approximation $\mathbf{L}$ can be obtained through bilateral random projections (BRP) [39],

$$\mathbf{L} = \mathbf{Y}_1(\mathbf{A}_2^T\mathbf{Y}_1)^{-1}\mathbf{Y}_2^T. \tag{6}$$

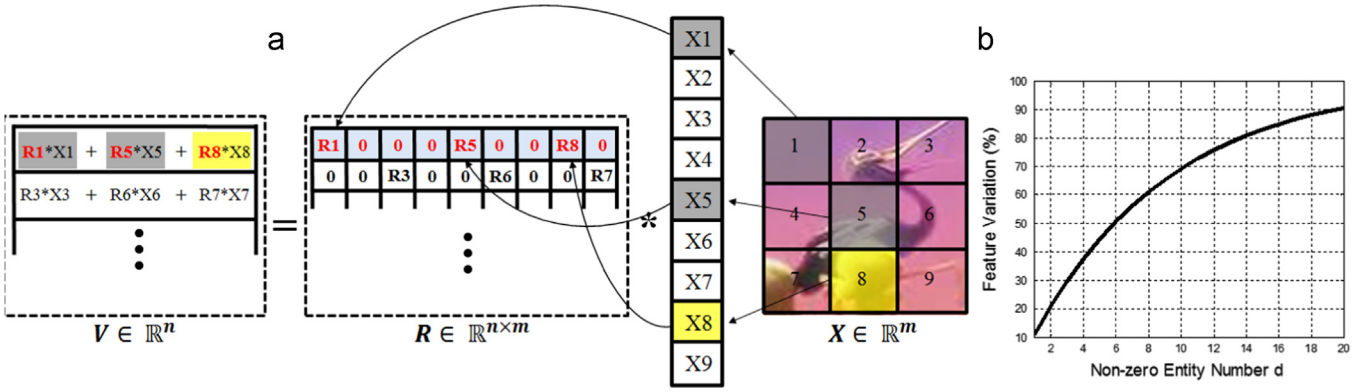

Fig. 2. Incapability analysis of global compressive sensing based representation when handling occlusion. The compressive sensing based feature representation [16,8] is illustrated in (a), where $m=9$, $n \ll m$, and the yellow element in $\mathbf{X}$ indicates the occluded region. The feature variation level caused by partial occlusion is demonstrated in (b). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

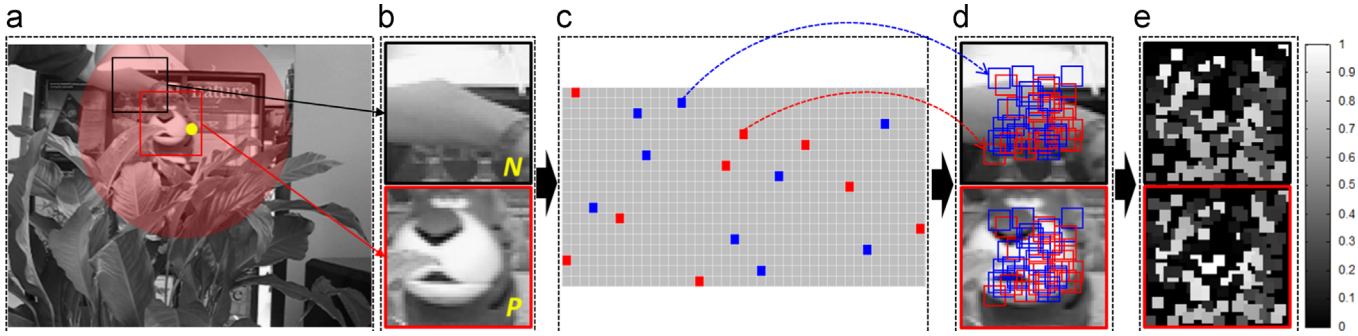

Fig. 3. Illustration of our appearance model. The disk region highlighted in (a) denotes the search area centered at the previous-frame tracking result (see the yellow dot). The red and black rectangles respectively denote two candidate targets, and (b) shows the corresponding zoom-in images, (c) is a toy demonstration of the constrained random matrix, wherein the blue, grey, and red squares represent -1, 0 and 1 respectively, (d) demonstrates the convolution results based on the rectangle filter and (e) shows the feature representations of the candidate targets. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

Here $\mathbf{A}_1$, $\mathbf{A}_2$ are independent Gaussian/SRFT random matrices [40], $\mathbf{A}_1 \in \mathbb{R}^{n\times r}$ and $\mathbf{A}_2 \in \mathbb{R}^{m\times r}$, $\mathbf{Y}_1 = \mathbf{X}\mathbf{A}_1$ and $\mathbf{Y}_2 = \mathbf{X}^T\mathbf{A}_2$. Since this low-rank approximation method tends to become inaccurate when the eigenvalues of $\mathbf{X}$ decay slowly, the power iteration modification (also known as the power scheme [41]) $\tilde{\mathbf{X}} = (\mathbf{X}\mathbf{X}^T)^q\mathbf{X}$ is introduced to accelerate the low-rank approximation and improve the accuracy [42], wherein q is the power iteration strength and $\mathbf{Y}_1 = \tilde{\mathbf{X}}\mathbf{A}_1$, $\mathbf{Y}_2 = \tilde{\mathbf{X}}\mathbf{A}_2$ is the BRP of $\tilde{\mathbf{X}}$. Thus, the low-rank approximation of the $\tilde{\mathbf{X}}$ can be obtained by

$$\tilde{\mathbf{L}} = \mathbf{Y}_1(\mathbf{A}_2^T\mathbf{Y}_1)^{-1}\mathbf{Y}_2^T. \tag{7}$$

By respectively conducting QR decomposition over $\mathbf{Y}_1$ and $\mathbf{Y}_2$, the rank-r approximation of $\mathbf{X}$ can be obtained as

$$\mathbf{L} = (\tilde{\mathbf{L}})^{1/(2q+1)} = \mathbf{Q}_1[\mathbf{R}_1(\mathbf{A}_2^T\tilde{\mathbf{X}}\mathbf{A}_1)^{-1}\mathbf{R}_2^T]^{1/(2q+1)}\mathbf{Q}_2^T = (\mathbf{X}\mathbf{Q}_2)\mathbf{Q}_2^T. \tag{8}$$

Therefore, the matrix $\mathbf{X}$ can be decomposed into a canonical "low-rank + sparse" formulation:

$$\mathbf{X} = \mathbf{L}+\mathbf{S}+\mathbf{E},\ \ \mathrm{rank}(\mathbf{L}) \le r,\ \ \mathrm{card}(\mathbf{S}) \le k, \tag{9}$$

where $\mathbf{E}$ denotes the approximation error.

In the visual tracking problem, the generative tracking methods commonly adopt an appearance model $\mathbf{A} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N\}$ to record previous tracking results (the bottom row in Fig. 4(b)). To accommodate fast appearance variation, we propose to establish a partial appearance model $\tilde{\mathbf{A}} = G(\mathbf{A}) = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_M\}$ (the top-right row in Fig. 4(b)) for current object tracking, where $G(\cdot)$ selects top-M $\mathbf{v}_i$ correlated to previous tracking results and $M \ll N$. Since the low-rank component $\tilde{\mathbf{L}}$ in $\tilde{\mathbf{A}}$ (Eq. (10)) can capture the common occurrences of the target object globally, the robust tracking result can be obtained by treating this low-rank coherency as the matching criterion instead of matching current candidate targets with all the individuals in $\tilde{\mathbf{A}}$ separately:

$$\tilde{\mathbf{L}} = (\tilde{\mathbf{A}}\mathbf{Q})^q\mathbf{Q}^T, \quad (\tilde{\mathbf{A}}\tilde{\mathbf{A}}^T)^q\tilde{\mathbf{A}}\tilde{\mathbf{A}}_2 = \mathbf{Q}\mathbf{R}. \tag{10}$$

Based on Eq. (10), we can obtain the low-rank approximation of the partial appearance model $\tilde{\mathbf{A}}$, and then compute the feature distance $\mathrm{dist}(\tilde{\mathbf{A}}, \mathbf{f}_i)$ between $\tilde{\mathbf{L}}$ and candidate target pool $\mathbf{F} = \{\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_k\}$. Finally, we seek the candidate with minimal distance as the true target object. It may be noted that, the above procedure is still time-consuming in principle. We propose to incorporate the low-rank approximation into an integrated coherency tracking procedure via

$$\mathbf{S} = \mathbf{F}-(\mathbf{F}\mathbf{Q})\mathbf{Q}^T, \quad (\tilde{\mathbf{A}}\tilde{\mathbf{A}}^T)^q\tilde{\mathbf{A}}\tilde{\mathbf{A}}_2 = \mathbf{Q}\mathbf{R}. \tag{11}$$

In Eq. (11), the explicit computation of $\tilde{\mathbf{L}}$ is completely avoided and the sparse matrix $\mathbf{S}$ can directly serve as the indicator of the true target object (Fig. 4(e)).

5. Tracking and appearance model updating based on low-rank coherency analysis

5.1. Low-rank coherency tracking

In fact, the non-zero entities of $\mathbf{S}$ in Eq. (11) are sparsely and independently residing in matrix $\mathbf{S}$, and $\mathbf{S}$ column-wisely corresponds to the candidate's partial violation. Thus, we can use $\mathbf{S}$ to compute the matching score, for example, the t-th candidate target will be taken as the true target object if $\sum_{j=1}^{n}\mathbf{S}_{t,j} = \min_i \sum_{j=1}^{n}\mathbf{S}_{i,j}$. However, because the appearance variation usually starts to occur

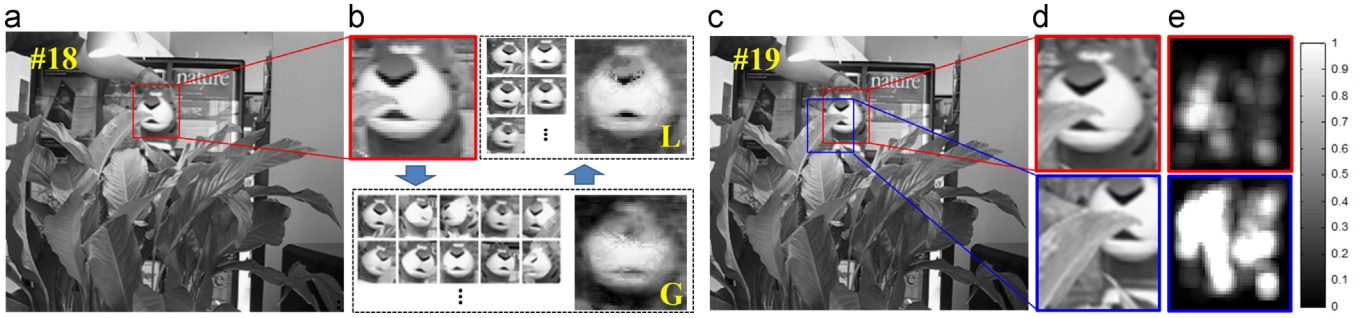

Fig. 4. Illustration of the low-rank coherency approximation. (a) Previous-frame observation of the target object, (b) formulates the partial appearance model based on previous tracking results, wherein the low-rank part of the appearance model (marked by yellow G) and the partial appearance model (marked by yellow L) are also demonstrated, (c) shows two candidate targets during the current-frame tracking procedure and (d) demonstrates **S** matrix's sparse vector (Eq. (11)) corresponding to the given candidate targets. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

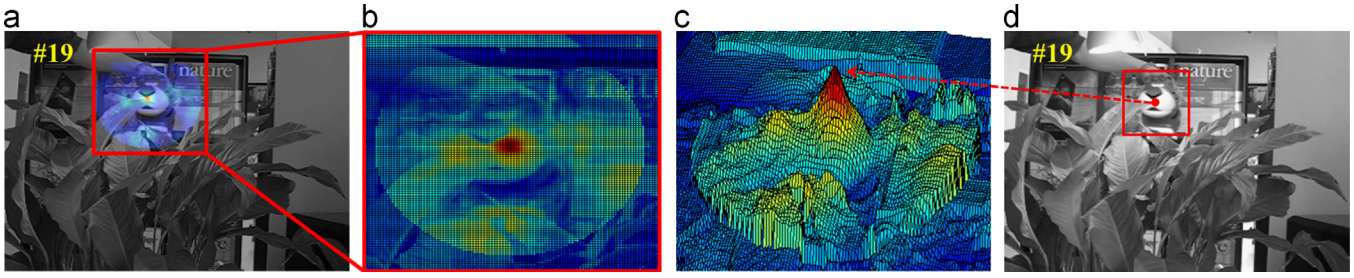

Fig. 5. Tracking based on low-rank coherency analysis. (a) Confidence map computed by Eq. (13), (b) zoom-in effect of the confidence map and (c) the 3D illustration of (b), and (d) the tracking result.

around the object boundary, in practice we use weighted voting to column-wisely compute the l_1-norm of **S** and take it as the matching score (Eq. (12)), wherein the entities near the center of candidate target will be assigned larger weights:

$$P_i = C \cdot \sum_{j=1}^{n} e^{-d_{i,j} \cdot |\mathbf{S}_{i,j}| / \sigma^2}. \tag{12}$$

Here n is the feature dimension, $i \in [1,k]$, k is the number of candidate targets, C is the constant normalization factor, $d_{i,j}$ is the Euclidean distance between the j-th sparse entity and the center of the i-th candidate target. The value of P_i indicates the probability of the i-th candidate target to be selected as the true target object.

Additionally, because of the low-rank coherency existing in the consecutive video frames, columns of **S** with low sparsity value (less variation) in previous tracking results are more trustworthy than those with high sparsity value. Therefore, the sparse matrix **S** corresponding to the previous tracking should also be regarded as the initial voting result prior to being utilized to guide current tracking. Eq. (13) formulates the candidate target's matching score, which can be efficiently computed by matrix multiplication:

$$\mathbf{p} = |\mathbf{S}| \times (\mathbf{D}^T \cdot \mathbf{W}^T), \tag{13}$$

where $\mathbf{p} \in \mathbb{R}^{k \times 1}$, $S \in \mathbb{R}^{k \times n}$, $W \in \mathbb{R}^{1 \times n}$, $D \in \mathbb{R}^{1 \times n}$ is the Gaussian-like distance weighting matrix illustrated in Eq. (12), $\mathbf{W} = C \cdot e^{-S_{(u,\cdot)}}$ is the sparsity prior obtained from the previous tracking result, and $S_{(u,\cdot)}$ is the sparse vector corresponding to the target object in the $t-1$ video frame. Fig. 5(b) and (c) demonstrates a tracking confidence map. It may be noted that the highest matching score is located at the center of the true target object (Fig. 5(d)).

5.2. Coarse-to-Fine tracking strategy

Since the movements of the target object are typically non-ballistic, object search can be limited within the nearby area of the current target location. However, naively conducting exhaustive search by matching the criterion **p** (Eq. (13)) is time-expensive. Hence, we design a coarse-to-fine search strategy to alleviate the computational burden. We record the Euclidean distances between the tracked locations in the consecutive frames, and use r^t to represent the Euclidean distance between tracked positions in the t frame and the $t-1$ frame. When the $t+1$ frame arrives, we take $r^{t+1} = r^t + 5$ as the radius of current searching cycle, and randomly sample k particles as the first-round candidate targets (Fig. 6(b)) via

$$k = \max(\rho \cdot \pi \cdot (r^{t+1})^2, 150), \tag{14}$$

where ρ is a down-sampling parameter and we empirically set it to be 0.7. Then, we compute the matching criteria $\mathbf{p}_i, i \in [1,k]$ using Eq. (13), and select the maximal **p** as the temporary anchor point. Centering around this anchor point (marked with black cycle in Fig. 6(b)), we further reduce the radius of the search region by half, and continue the tracking procedure iteratively. In practice, we totally iterate this course-to-fine tracking procedure 3 times to make tradeoff between computation and accuracy. Meanwhile, it should also be noted that matrix **Q** in Eq. (11) only needs to be computed in the first iteration, which remains constant during the rest of iterations.

5.3. Occlusion handling

One critical issue of voting-based global matching criteria is how to eliminate the untrustworthy occlusions. Since the sparsity values of the occluded positions in **S** are on average larger than the normal ones, the global matching score of an occluded target object may become smaller than a false-alarm candidate target, which tends to result in false tracking result. One commonly used solution is to regard the entities with sparsity value larger than a pre-defined threshold as occluded area. However, because this pre-defined threshold depends on which candidate is the true target object, in fact its selection is a chicken-and-egg problem. Fortunately, we can alleviate this dilemma by additionally incorporating constraints into our low-rank tracking

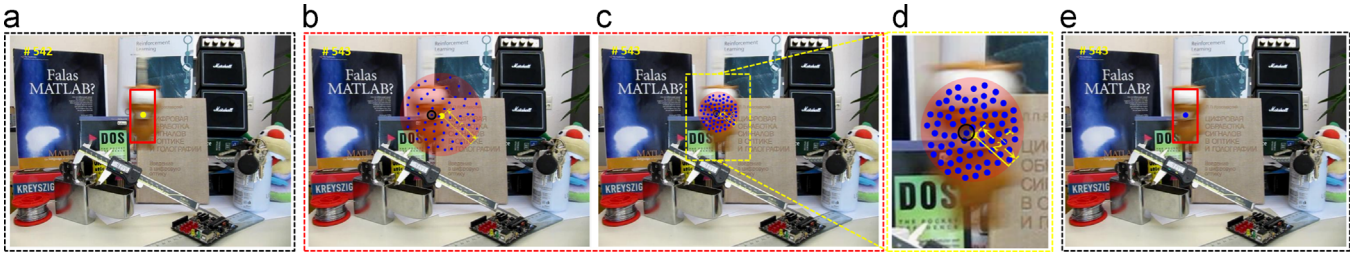

Fig. 6. Demonstration of the coarse-to-tine tracking strategy. (a) The tracking result corresponding to the 542 frame of the lemming sequence. (b) The first tracking iteration in 543 frame with search radius r, blue points denote k sampled candidate targets, and the candidate targets with maximum score are marked by black cycle. (c) The second tracking iteration in 543 frame with search radius $r/2$ centering around the black cycle. (d) The zoom-in effect of (c). (e) The final tracking result after 3 tracking iterations. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

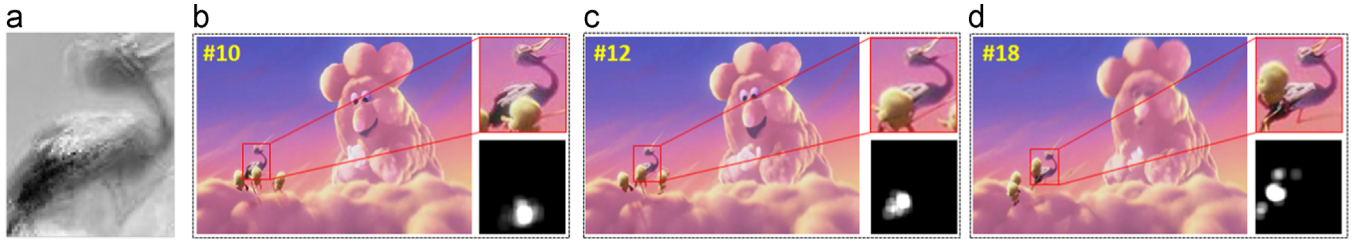

Fig. 7. Occlusion mask demonstration. (a) One of the approximated low-rank components for bird-2 sequence. (b)–(d) demonstrate the occlusion masks resulted from our method.

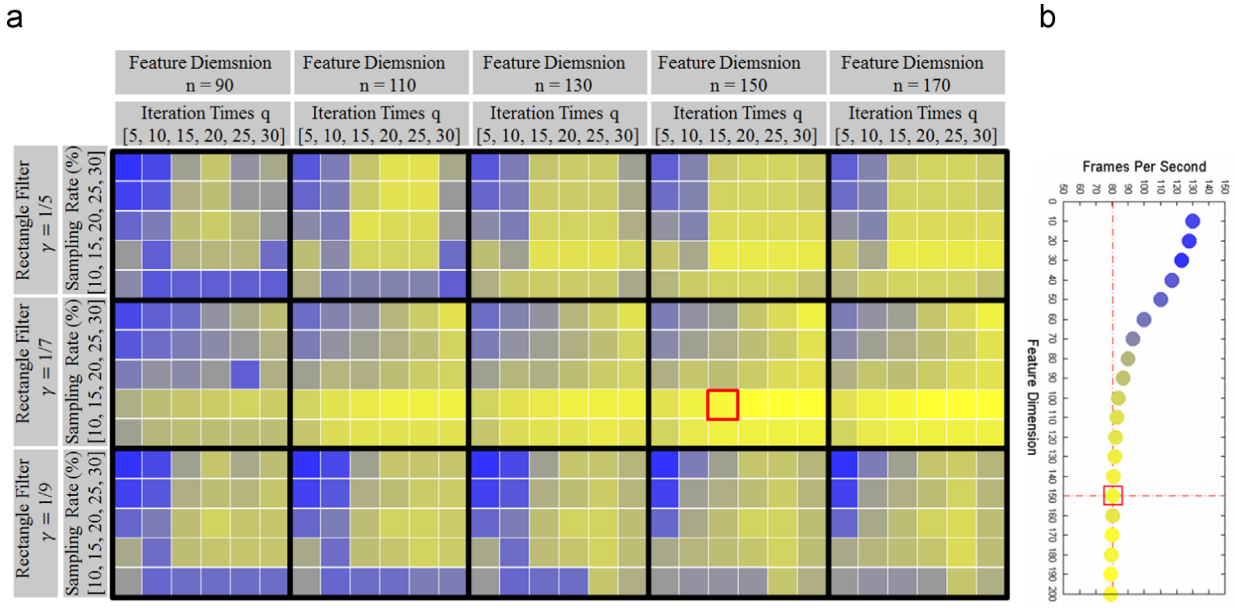

Fig. 8. Parameter selection analysis. (a) The tracking performance with different parameters is illustrated. (b) The tracking performance (CLE and FPS) with different feature dimensions. The color from blue to yellow indicates the performance from worse to better. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

framework:

$$\mathbf{O}_{i,j} = \begin{cases} 1 & \text{if } (\mathbf{S}_{i,j} - \alpha T) < 0 \\ 0 & \text{if } (\mathbf{S}_{i,j} - \alpha T) \geq 0 \end{cases}. \tag{15}$$

In Eq. (15), $\mathbf{O}_{i,j}$ indicates the binary occlusion mask of the i-th candidate target, whose non-zero entities correspond to non-occluded position. Here $i \in [1, k]$, $j \in [1, n]$, $\alpha = 1.5$, and we set T to be the absolute mean of $\mathbf{S}$ (Eq. (11)) obtained in the **first** tracking iteration (see Section 5.2). By incorporating the occlusion mask $\mathbf{O}$ into each candidate target, in the first tracking iteration, the matching score of the i-th candidate target corresponding to the j-th occlusion mask can be redefined as:

$$\mathbf{P}_{i,j} = (\mathbf{O}_{(j,\cdot)} \odot |\mathbf{S}_{(i,\cdot)}|) \times (\mathbf{D} \cdot \mathbf{W}^T) + \beta \cdot T \cdot (n - \|\mathbf{O}_{(j,\cdot)}\|_0). \tag{16}$$

Here n is the feature dimension, $\odot$ is element-wise Hadamard product, $\|\cdot\|_0$ denotes l_0-norm, $\beta = 1.5$. And the second part of Eq. (16) is a penalty term used to avoid occlusion bias. Thus, we can obtain the $\mathbf{P}$ matrix as

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{(1,1)} & \mathbf{P}_{(1,2)} & \cdots & \mathbf{P}_{(1,k)} \\ & & \ddots & \\ \mathbf{P}_{(k,1)} & \mathbf{P}_{(k,2)} & \cdots & \mathbf{P}_{(k,k)} \end{bmatrix}, \tag{17}$$

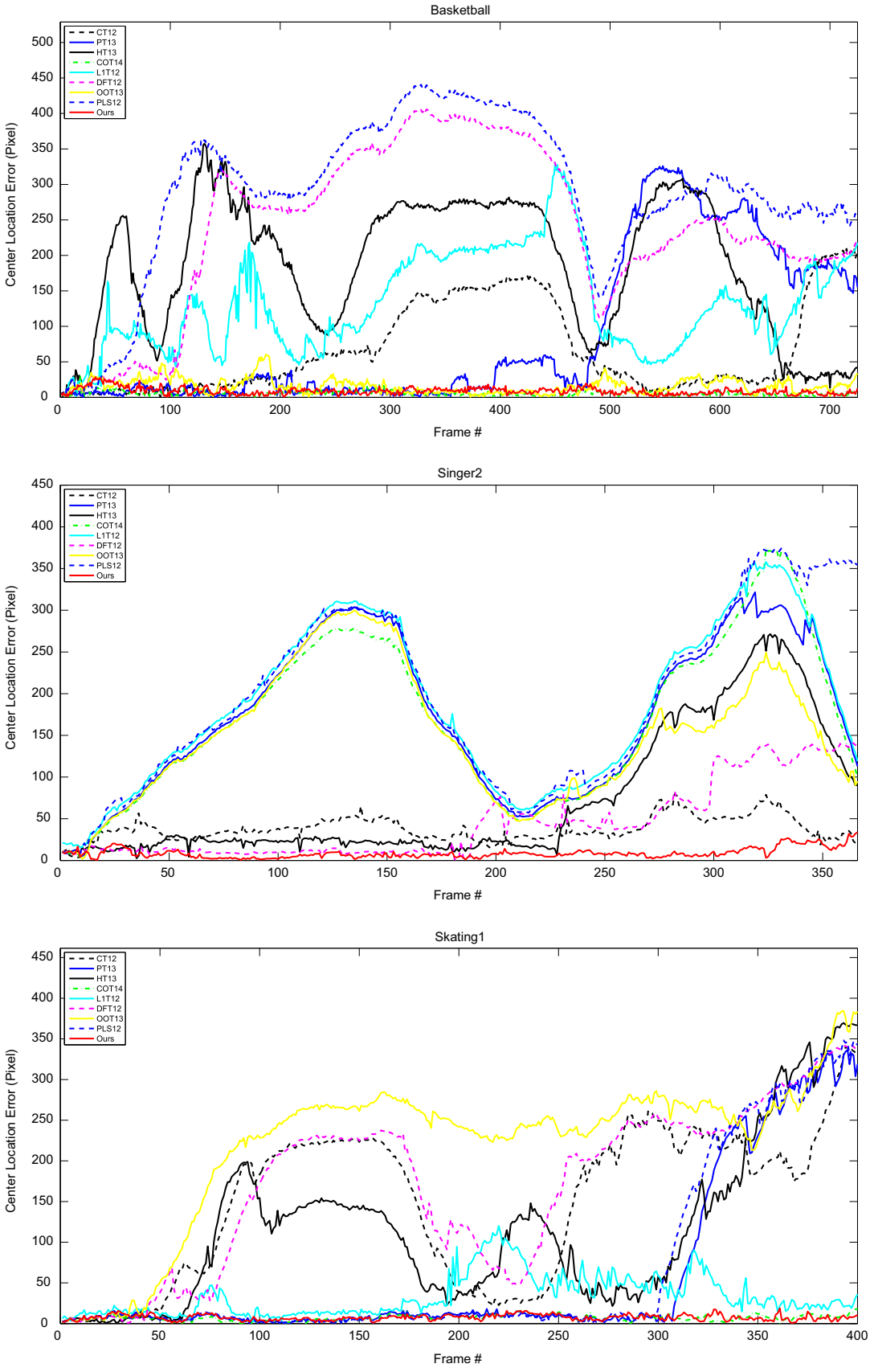

Fig. 9. Center location error (CLE) comparisons with 8 state-of-the-art methods over 36 video sequences. In the interest of space, we only list 8 comparison results here, and more results can be found in our supplementary material.

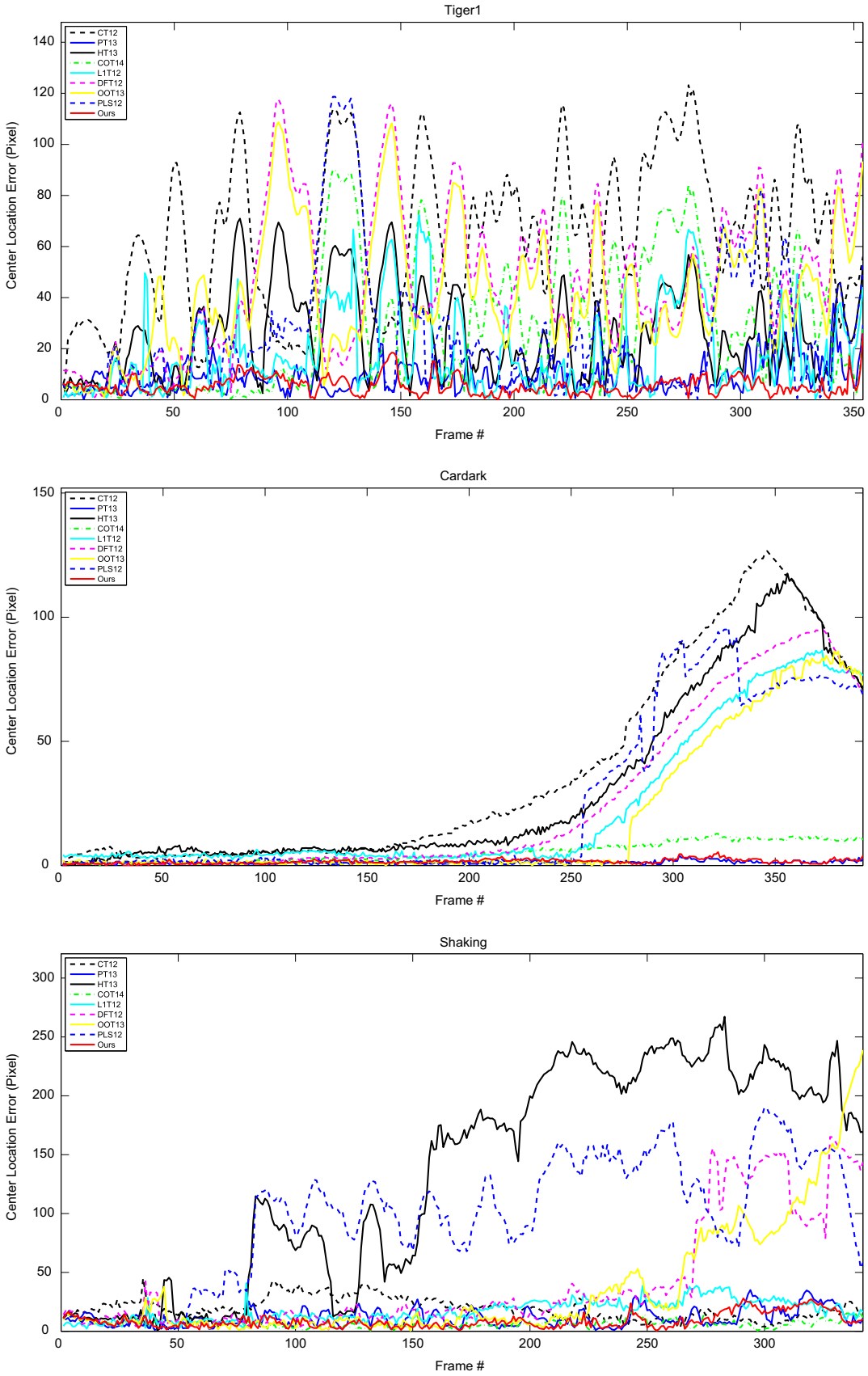

Fig. 9. (*continued*)

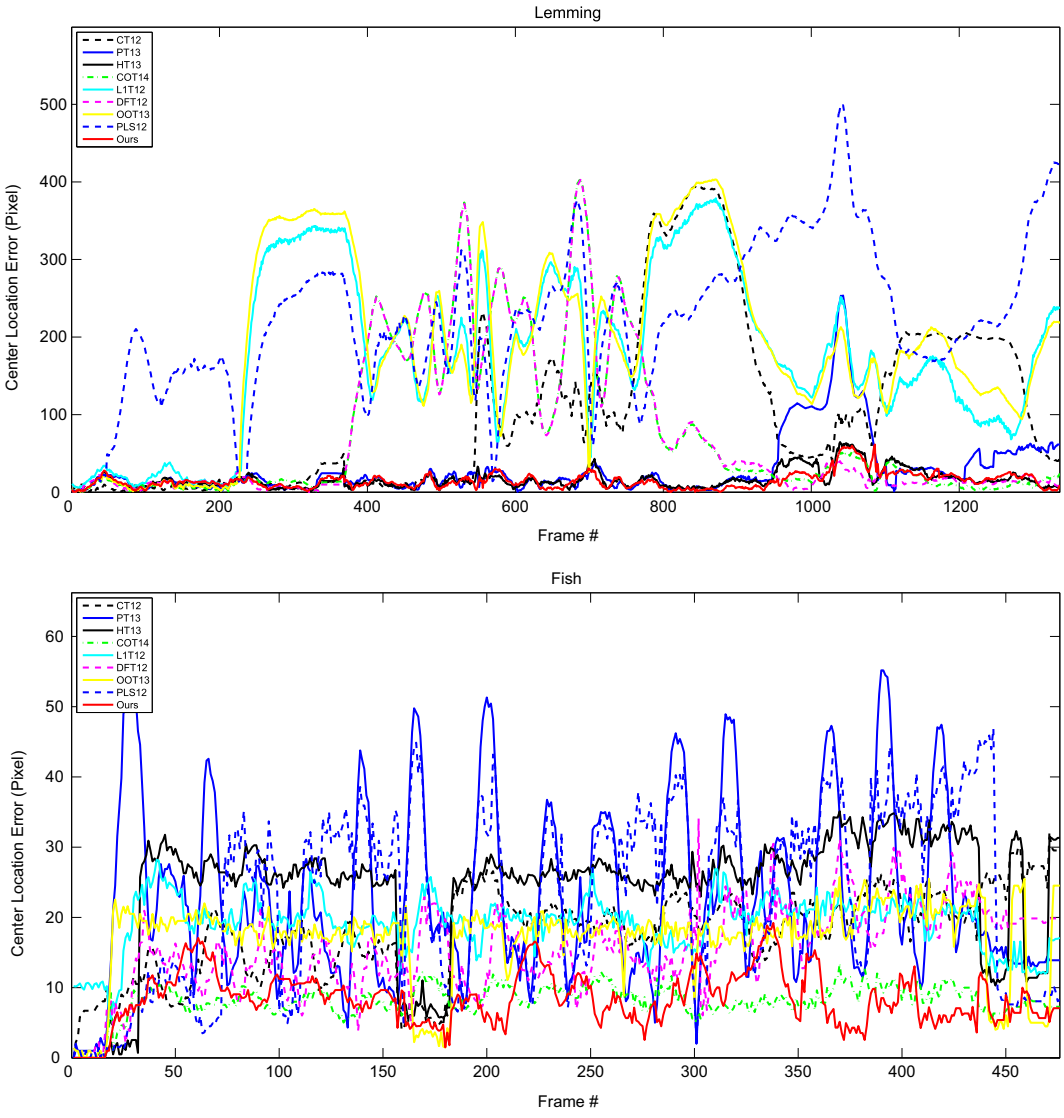

Fig. 9. (*continued*)

Table 2
The detailed characteristics of 36 challenging video sequences used in our experiments (**Part A**). The total number of evaluated video frames is 21065.

Video sequence	3D pose	Illumination	Occlusion	Blur	Fast motion	Frames
Animal [47]	×	×	√†	√†	√‡	71
Basketball [47]	√‡	√†	√‡	×	√†	725
Bird2 [10]	√‡	×	√†	×	×	103
David2 [15]	√‡	√‡	√†	√†	√†	462
Coupon [6]	√‡	×	√†	×	×	327
OccludedFace [6]	×	×	√‡	×	√†	899
OccludedFace2 [6]	√†	×	√‡	×	×	815
Girl [6]	√‡	√†	√‡	×	√†	502
Gym [47]	√‡	×	×	×	√†	767
Jumping [20]	×	√†	×	√‡	√‡	313
Lemming [48]	√‡	√†	√‡	√‡	√‡	1336
Shaking [47]	√†	√‡	√†	√†	√†	342
Skating [47]	√‡	√‡	√‡	√‡	√‡	400
Skating2 [47]	√‡	√†	√‡	√‡	√‡	707
Sylvester [15]	√‡	√‡	×	√†	√†	1345
Tiger1 [6]	√‡	√‡	√‡	√‡	√‡	345
Twining [6]	√‡	×	×	×	×	472
Woman [32]	√†	×	√‡	×	√‡	597

√‡ indicates heavy variation and √† indicates mid-level variation.

Table 3
The detailed characteristics of 36 challenging video sequences used in our experiments (**Part B**). The total number of evaluated video frames is 21065.

Video sequence	3D pose	Illumination	Occlusion	Blur	Fast motion	Frames
Singer2 [47]	×	√‡	×	√†	×	366
Crossing [32]	×	√†	×	×	√†	120
Mhyang [32]	√†	√†	×	×	×	1490
Fleetface [32]	√‡	×	×	×	×	699
Subway [32]	×	×	√†	√†	×	175
Car4 [15]	×	√‡	×	×	×	659
Walking [32]	×	×	√‡	×	×	412
Freeman1 [32]	√‡	×	√‡	√†	√†	326
Cardark [15]	×	√†	×	√†	√†	393
Carscale [32]	×	√†	√‡	√‡	√†	252
Football1 [49]	×	√†	√†	√‡	√‡	81
Freeman3 [32]	√†	×	√†	√†	√†	474
Fish [32]	×	√‡	×	√†	√†	476
Dog1 [15]	√†	√†	×	×	×	1350
Boy [32]	√‡	×	×	√†	√†	602
Bike [50]	√‡	√†	×	×	×	228
David [20]	√‡	√†	×	×	×	537
Tiger2 [6]	√‡	√‡	√‡	√‡	√†	365

√‡ indicates heavy variation and √† indicates mid-level variation.

Table 4
Center location error (CLE) (in pixels), Part A. Bold fonts indicate the best performance while the italic fonts indicate the second-best ones, and the underlined values indicate the third-best ones.

Method	Ours	COT14	PT13	HT13	OOT13	PLS12	CT12	L1T12	DFT12
Animal	**10**	**10**	*10*	14	212	136	38	215	78
Woman	11	13	**7**	*8*	115	153	110	125	20
David2	**8**	11	*9*	13	18	67	10	13	14
Girl	**12**	35	14	*13*	14	*13*	36	*13*	30
Tiger1	**5**	28	*9*	26	41	25	63	18	46
Sylvester	12	12	**9**	61	56	51	*11*	46	51
Skating1	**6**	*7*	60	116	219	66	148	33	168
Skating2	**31**	*45*	52	226	194	73	131	191	180
Occludedface1	16	58	16	**11**	16	*14*	28	20	21
Occludedface2	15	**11**	*13*	15	20	23	21	48	39
Bird2	**12**	*18*	21	102	55	19	19	26	81
Shaking	*8*	**6**	11	133	35	96	19	17	40
Coupon	16	**4**	20	*5*	*5*	19	19	62	6
Twining	10	**7**	8	16	**6**	34	17	9	11
Gym	*11*	13	**9**	105	67	301	28	119	89
Basketball	*8*	**5**	86	15	185	287	66	131	237
Jumping	**10**	*13*	*13*	23	21	48	24	32	94
Lemming	**14**	*16*	29	*16*	187	222	104	178	79

where $\mathbf{P} \in \mathbb{R}^{k \times k}$, k is the number of candidate targets. Since the true target object should constantly have maximal $\mathbf{P}$ value for different occlusion masks, matrix $\mathbf{P}$ in Eq. (17) should be further filtered to obtain robust occlusion mask by using

$$\tilde{\mathbf{p}}_i = \begin{cases} 1 & \text{if } \operatorname{diag}(\mathbf{P}_{(i,\cdot)}) = \min_j(\mathbf{P}_{(i,j)}) \\ 0 & \text{otherwise} \end{cases}, \tag{18}$$

where $\tilde{\mathbf{p}}_i \in \mathbb{R}^{k \times 1}$ and $\operatorname{diag}(\cdot)$ means the i-th diagonal element. Then, we take $\mathbf{O}_{(u,\cdot)}$ as the final occlusion mask $\tilde{\mathbf{O}}$ if

$$\operatorname{diag}(\mathbf{p}_u) = \min(\operatorname{diag}(\mathbf{P}) \odot \tilde{\mathbf{p}}). \tag{19}$$

And the occlusion mask $\tilde{\mathbf{O}}$ remains constant for the second and the third tracking iterations (see demonstrations in Fig. 7(b–d)). Therefore, the final formulation of the matching criteria for the second and third tracking iterations can be formulated as

$$\mathbf{p} = |\mathbf{S}| \times (\tilde{\mathbf{O}} \cdot \mathbf{D}^T \cdot \mathbf{W}^T). \tag{20}$$

5.4. Partial appearance model

When the target object has been located in the current frame, we resort to the biased low-rank analysis to construct a partial appearance model for next video frame using Eq. (21), which is similar to the low-rank coherency tracking procedure (Eq. (11)) and can be defined as

$$\mathbf{S} = \mathbf{A} - (\mathbf{AQ})\mathbf{Q}^T, \ (\mathbf{BB}^T)^q\mathbf{BA}_2 = \mathbf{QR}. \tag{21}$$

Here $\mathbf{A}$ denotes the appearance model, $\mathbf{B} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_5]^T$, $\mathbf{v}_i \in \mathbb{R}^{1 \times n}$ indicates previous tracking results (five target observations neighboring the target location in the $t-1$ frame), and the remaining configurations are all the same as those in Eq. (11). Then, the partial appearance model can be established by selecting M observations from $\mathbf{A}$ according to $\|\mathbf{S}\|_1$ in an ascending order. A large value of M means that the partial appearance model $\tilde{\mathbf{A}}$ concentrates on a long-term stable structure of the target object, thus, the low-rank coherency of this partial model may fail to

Table 5
Center location error (CLE) (in pixels), Part B. Bold fonts indicate the best performance while the italic fonts indicate the second-best ones, and the underlined values indicate the third-best ones.

Method	Ours	COT14	PT13	HT13	OOT13	PLS12	CT12	L1T12	DFT12
Singer2	**8**	170	174	71	150	195	*37*	186	44
Crossing	**2**	*4*	**2**	50	7	4	5	53	22
Mhyang	**3**	*4*	**3**	**3**	**3**	**3**	24	9	6
Fleetface	*29*	38	51	25	**19**	74	55	51	41
Subway	*4*	158	**2**	*4*	140	158	11	148	*4*
Car4	**4**	*9*	**4**	**4**	78	4	79	84	17
Walking	**4**	6	8	12	8	**4**	*5*	*5*	*5*
Freeman1	**11**	179	*13*	177	164	26	155	149	133
Cardark	**1**	*5*	**1**	30	18	24	37	22	24
Carscale	**31**	*35*	*35*	138	66	93	73	45	119
Football1	**12**	23	18	18	**12**	21	20	53	*17*
Freeman3	**7**	**7**	46	46	46	*10*	71	42	30
Dog1	*4*	**3**	6	7	5	47	10	9	67
Boy	7	*5*	7	7	23	**3**	13	17	8
Bike	**6**	**6**	*11*	**6**	146	*11*	214	12	**6**
David	*13*	14	*13*	**12**	27	27	68	34	16
Tiger2	12	**9**	*11*	75	44	33	21	47	75

obtain correct tracking results when the object's appearance is changing drastically. Meanwhile, partial appearance model constructed with small value M is sensitive to current observation, which gives rise to tracking drift. Therefore, we empirically assign $M=15$ (out of 100 observations of $\mathbf{A}$) to balance with the above tradeoff, which will be detailed in Section 6.1.

5.5. Appearance model updating

Since the appearance of the target object oftentimes changes in consecutive frames, the dynamic updating is indispensable for any good appearance model. To handle the fast appearance change, the majority of generative tracking methods tend to only update the elements that mostly correlate to current observation. However, due to the low-rank coherency existing in consecutive video frames, such updating strategies can easily fall into bootstrap and cause tracking drift, wherein it will keep updating the identical observation elements in the appearance model in each frame.

In sharp contrast, we employ unbiased low-rank analysis to update the appearance model globally toward the direction of rank increase. Similar to the construction procedure of partial appearance model, we use Eq. (22) to compute the sparse matrix $\mathbf{S}$ of the appearance model:

$$\mathbf{S}=\mathbf{A}-(\mathbf{AQ})\mathbf{Q}^T,\quad (\mathbf{AA}^T)^q\mathbf{AA}_2=\mathbf{QR}. \tag{22}$$

Because the smallest sparsity value in $\mathbf{S}$ implies the center of feature distance based clustering, we directly use current tracking observation to replace the top-K observations of $\mathbf{A}$ according to $\|\mathbf{S}\|_1$ in an ascending order. And this aggressive updating strategy has two specific advantages: (1) it greatly enhances the diversity of the appearance model, and thus make our low-rank coherency tracking more robust to the drift problem; and (2) it maintains a group of observations that strongly correlate to current observation, which enables efficient appearance adaptation.

Besides, the adaptive choice of K should also facilitate the robustness of the appearance updating. We associate the choices of K with the current tracking result via

$$K=\begin{cases}1, & \|\mathbf{S}_{(t,\cdot)}\|_1 \le T_1\\ 2, & \|\mathbf{S}_{(t,\cdot)}\|_1 > T_1\\ 0, & 1-\|\tilde{\mathbf{O}}\|_0/n > T_2\end{cases}, \tag{23}$$

where $\|\mathbf{S}_{(t,\cdot)}\|_1$ indicates the sparse vector of $\mathbf{S}$ corresponding to current tracking result, T_1 is a threshold to control the updating strength, and we empirically set it to be 5. We set $T_2=0.55$, and it means the updating procedure will be suspended when heavy occlusion occurs. Here, $\|\tilde{\mathbf{O}}\|_0$ is the number of non-zero entities in occlusion mask $\tilde{\mathbf{O}}$ (Eqs. (18) and (19)).

When occlusion happens, the partial updating strategy should also be adopted to prevent the appearance model from leaning towards the occlusion objects. Suppose $\mathbf{F}^{t-1}$ is the observation to be replaced by current tracking observation $\mathbf{F}^t$, we reconstruct $\mathbf{F}^{t-1}$ by

$$\mathbf{F}_i^{t-1}=\begin{cases}\mathbf{F}_i^t & \text{if } \tilde{\mathbf{O}}_i=0\\ (\mathbf{F}_i^t+2\times\tilde{\mathbf{F}}_i)/3 & \text{if } \tilde{\mathbf{O}}_i=1\end{cases}. \tag{24}$$

Here $\tilde{\mathbf{F}}\in\mathbb{R}^{1\times n}$ is a partial observation with minimal $\|\mathbf{S}\|_1$ in the third tracking iteration (see Section 5.2), and $\tilde{\mathbf{O}}$ is the occlusion mask obtained in the first tracking iteration. As a result, our appearance model can be adaptive to the partial appearance change when the target object is being occluded. To better convey our entire technical solutions, all the main steps of our low-rank coherency tracking are documented in Algorithm 1.

Algorithm 1. Low-rank coherency tracking.

Input: Tracking location l_{t-1} in $t-1$ frame and appearance model $\mathbf{A}$.
Output: Tracking location l_t in t frame and the updated appearance model $\mathbf{A}$.
Initialization: $C=l_{t-1}$.
1. Compute the feature representation (Eq. (5)) of previous tracking result t to obtain $\mathbf{f}_{t-1}$.
2. Use $\mathbf{f}_{t-1}$ to formulate the partial appearance model $\tilde{\mathbf{A}}$ via low-rank analysis (Section 5.4).
For i=1:3
3. Sample candidate targets centering around C with search radius r (Section 5.2).
4. Compute the feature representation (Eq. (5)) of candidate targets.
5. Apply the low-rank analysis to obtain S (Eq. (11)).
6. Compute feature prior D and W (Eq. (13)).
7. Estimate the occlusion mask $\tilde{\mathbf{O}}$ and integrate it with matrix $\mathbf{S}$ (Section 5.3).
8. Perform the low-rank coherency based tracking to obtain the tracking result l_t (Section 5.1), and assign $C=l_t$.
End For
9. Update the appearance model $\mathbf{A}$ via unbiased low-rank analysis (Section 5.5), and output current tracking result l_t.

Table 6
Average center location error (CLE) (in pixels). Bold fonts indicate the best performance while the italic fonts indicate the second-best ones, and the underlined values indicate the third-best ones.

Method	Ours	COT14	PT13	HT13	OOT13	PLS12	CT12	L1T12	DFT12
Average CLE	**10.6**	27.6	*22.9*	49.9	62.9	66.8	50.1	63.3	53.6

Table 7
Summary of total *Best*, *Second-Best* and *Third-best* times for each algorithm over 36 video sequences based on center location error (CLE). Bold fonts indicate the *Best* performance while the *italic* fonts indicate the *Second-best* ones, and the underlined values indicate the *Third-best* ones.

Method	Ours	COT14	PT13	HT13	OOT13	PLS12	CT12	L1T12	DFT12
Best	**21**	*10*	8	5	4	4	0	0	1
Second-best	7	**11**	*10*	4	2	5	3	2	4
Third-best	*6*	4	**8**	*6*	2	2	4	4	5

Table 8
Success rate (SR) (%), *Part A*. Bold fonts indicate the *Best* performance while the *italic* fonts indicate the *Second-best* ones, and the underlined values indicate the *Third-best* ones.

Method	Ours	COT14	PT13	HT13	OOT13	PLS12	CT12	L1T12	DFT12
Animal	**100**	**100**	90.1	*91.5*	4.22	5.63	35.2	4.22	8.48
Woman	*92.1*	68.3	**95.1**	82.0	18.9	10.7	17.5	15.5	62.8
David2	**91.9**	88.3	*91.3*	82.0	81.3	30.0	86.1	79.8	71.4
Girl	85.4	43.4	95.8	86.0	23.5	**99.4**	89.4	*96.2*	49.2
Tiger1	**99.1**	46.3	*92.0*	37.5	14.6	43.7	3.95	67.5	17.5
Sylvester	81.9	81.1	**96.8**	27.7	22.3	30.7	*82.8*	32.7	25.0
Skating1	*95.5*	**96.7**	74.0	15.5	9.25	73.7	12.5	36.2	10.7
Skating2	**78.0**	*58.6*	48.9	7.92	11.1	21.2	3.67	2.54	4.52
Occludedface1	**100**	84.8	98.8	98.3	**100**	97.1	57.2	*99.2*	60.9
Occludedface2	**100**	**100**	97.1	*98.7*	89.6	72.1	94.6	58.4	61.3
Bird2	**94.9**	*88.8*	57.5	7.07	56.5	60.6	61.6	50.5	15.1
Shaking	**100**	**100**	*94.4*	25.1	68.1	15.7	69.2	73.9	65.4
Coupon	79.1	**99.0**	69.7	*98.7*	*98.7*	87.1	88.0	39.7	97.5
Twining	83.8	94.2	*94.9*	60.8	**96.3**	40.8	59.5	92.3	77.1
Gym	**96.3**	87.2	*95.3*	4.82	24.9	1.82	36.7	3.38	19.6
Basketball	*96.4*	**99.8**	54.0	1.79	85.3	4.55	27.0	3.17	6.62
Jumping	**100**	*91.6*	90.1	47.6	53.9	28.7	42.1	24.9	15.9
Lemming	**78.0**	42.1	49.6	*74.1*	16.9	4.34	36.3	13.7	42.7

6. Experimental results and evaluation

6.1. Parameter selection

In principle, there are four parameters that could influence the efficiency and performance of our low-rank coherency tracking: (1) the dimension of the feature representation n; (2) the iteration times q for low-rank approximation; (3) the sampling rate of the partial appearance model M/N; and (4) the rectangle filter size γ. As the dimension of feature representation can simultaneously affect the performance and efficiency, we shall first concentrate on the feature dimension, and later, we analyze the remaining three parameters.

The dimension of feature representation: Since our feature representation is based on compressive sensing, the more non-zero entities in measurement matrix, the more accurate the distance distribution can be preserved. As shown in Fig. 8(a), better tracking performance (it may be noted that, the color turns from blue to yellow indicates the performance varying from worse to better) can be obtained by increasing the feature dimension. Consider the accuracy and efficiency tradeoff, an optimal feature dimension can greatly improve the overall performance. As we can see from Fig. 8(b), the *FPS* drops rapidly when we increase the feature dimension from 20 to 100, and then it has slight performance improvement when the dimension continues to increase. Therefore, to balance the tradeoff, we assign $n=150$ as the optimal feature dimension (Fig. 9).

The iteration times q for low-rank approximation: In fact, the accuracy of low-rank approximation can also be improved by increasing the iteration times. However, an extremely precise low-rank approximation is not the top priority of the real-time visual tracking, because the principal low-rank pattern is strongly related to the largest eigenvalue of the initial power iterations. Meanwhile, our localized compressive sensing based representation constructs a feature space with constrained feature diversity, which can guarantee accurate low-rank approximation without numerous iterations. Hence, according to the Fig. 8(a), we assign $q=20$ to avoid unnecessary time consumption.

The sampling rate of partial appearance model: Obviously, a large sampling number indicates the partial appearance model concentrates on the long-term stable coherency, while a small sampling number corresponds to fast partial appearance variation (which in turn easily results in possible drift problems). Hence, an extremely large or small sampling number is not appropriate for robust visual tracking, and a dynamically changed sampling rate is not feasible either because of the unpredictable nature of incoming video frames. From Fig. 8(a), we can find that the optimal choice of the sampling rate is $N=15$ (the capacity of the appearance model is $M=100$).

The rectangle filter size: From the perspective of patch-based feature representation, a rectangle filter with small size γ can improve the tracker's occlusion handling capability at the expense

Table 9
Success rate (SR) (%), *Part B*. Bold fonts indicate the *Best* performance while the *italic* fonts indicate the *Second-best* ones, and the underlined values indicate the *Third-best* ones.

Method	Ours	COT14	PT13	HT13	OOT13	PLS12	CT12	L1T12	DFT12
Singer2	**100**	3.55	3.55	*62.2*	3.55	3.27	29.2	3.27	51.3
Crossing	**100**	**100**	**100**	14.1	*99.1*	83.3	96.6	23.3	67.5
Mhyang	**100**	94.5	**100**	*99.8*	**100**	**100**	33.3	96.6	91.4
Fleetface	63.9	53.6	32.1	*75.8*	**88.8**	19.0	71.1	58.1	51.0
Subway	*98.2*	22.2	*98.2*	**100**	21.7	16.0	77.7	22.2	98.2
Car4	**100**	*97.2*	**100**	**100**	34.2	**100**	35.3	34.9	48.1
Walking	**100**	83.0	*99.7*	38.1	96.6	**100**	**100**	**100**	**100**
Freeman1	**64.1**	38.3	*45.0*	2.14	11.6	29.7	4.90	11.3	11.0
Cardark	**100**	*90.0*	**100**	58.0	70.7	64.8	44.2	66.4	63.1
Carscale	**67.8**	63.4	59.1	55.5	*65.0*	63.8	62.3	64.2	6.74
Football1	**92.5**	42.5	51.2	57.5	*77.5*	32.5	33.7	7.50	73.7
Freeman3	*58.3*	56.0	38.0	21.9	31.2	**72.7**	0.00	39.5	52.8
Fish	**100**	**100**	62.1	21.6	98.7	45.3	90.5	*99.3*	94.7
Dog1	*98.8*	**99.9**	87.1	79.4	91.3	73.7	79.7	80.8	31.6
Boy	96.8	**100**	94.0	*98.0*	65.9	**100**	62.6	62.4	92.6
Bike	**100**	**100**	87.1	**100**	39.8	*88.4*	17.2	80.5	**100**
David	**100**	**100**	*99.8*	**100**	0.18	53.0	38.1	93.2	75.0
Tiger2	*85.2*	**94.5**	81.3	3.56	21.0	35.8	56.4	20.5	3.28

Table 10
Average success rate (SR) (%). Bold fonts indicate the *Best* performance while the *italic* fonts indicate the *Second-best* ones, and the underlined values indicate the *Third-best* ones.

Method	Ours	COT14	PT13	HT13	OOT13	PLS12	CT12	L1T12	DFT12
Average CLE	**90.9**	77.9	*78.4*	56.5	55.9	50.2	48.1	47.0	47.0

Table 11
Summary of total *Best* and *Second-Best* times for each algorithm over 36 video sequences based on success rate (SR) (%). Bold fonts indicate the *Best* performance while the *italic* fonts indicate the *Second-best* ones, and the underlined values indicate the *Third-best* ones.

Method	Ours	COT14	PT13	HT13	OOT13	PLS12	CT12	L1T12	DFT12
Best	**23**	*13*	5	4	4	6	1	1	2
Second-best	7	5	**9**	*8*	3	2	1	3	1
Third-best	2	*5*	**10**	1	*5*	0	*5*	4	4

Table 12
Frames per second (FPS) of each method. Bold fonts indicate the *Best* performance while the *italic* fonts indicate the *Second-best* ones, and the underlined values indicate the *Third-best* ones. All of these methods run on a computer with Quad Core i7-3770 3.4 GHz, 8 GB RAM.

Method	Ours	COT14	PT13	HT13	OOT13	PLS12	CT12	L1T12	DFT12
Average FPS	*70*	**94**	5.6	0.7	2.9	10.5	42	12.3	6

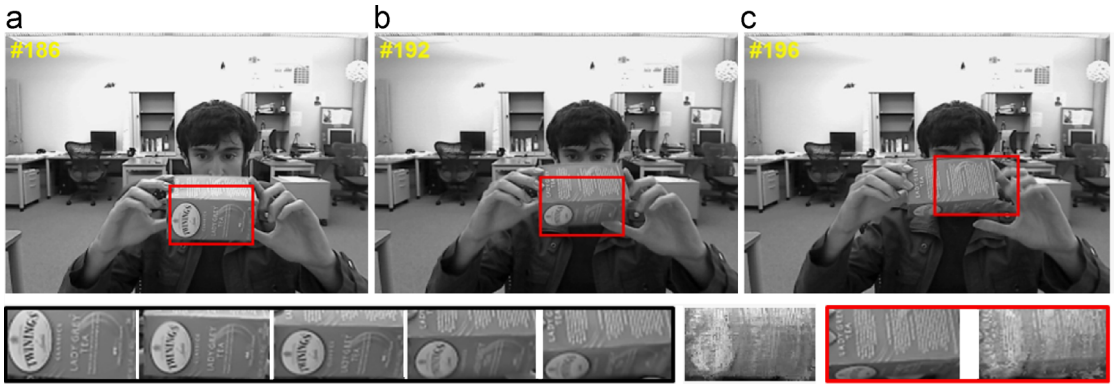

Fig. 10. Illustration of our method's limitation. The top row shows the tracking result of our method. The bottom row (a) demonstrates a part of the partial appearance model used in 192 frame. (b) Its corresponding approximated low-rank part. (c) The target observation and the approximated low-rank part used in 196 frame.

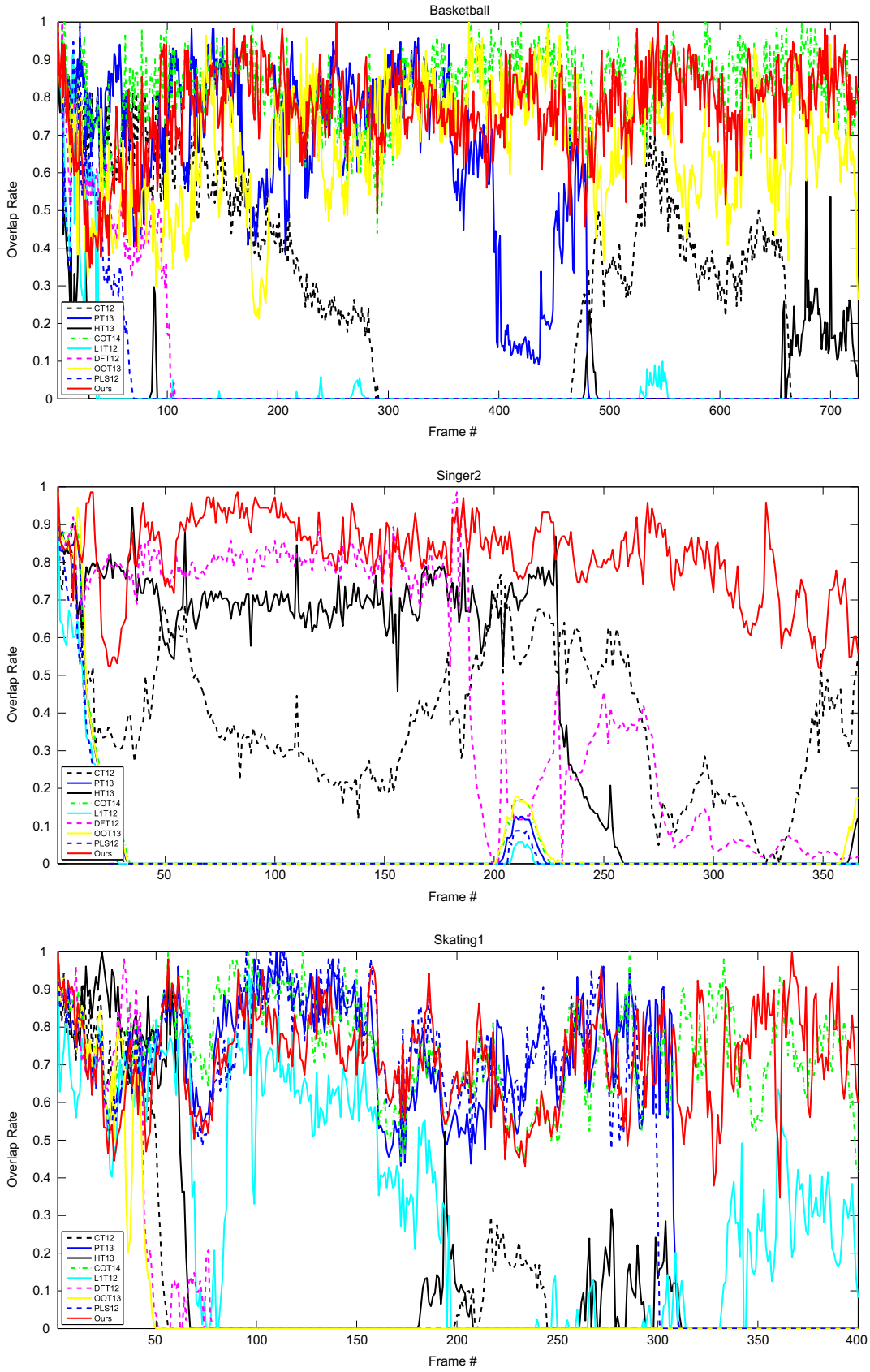

Fig. 11. Overlapping rate (OR) based comparisons with 8 state-of-the-art methods over 36 video sequences. In the interest of space, we only list 8 comparison results here, and more results can be found in our supplementary material.

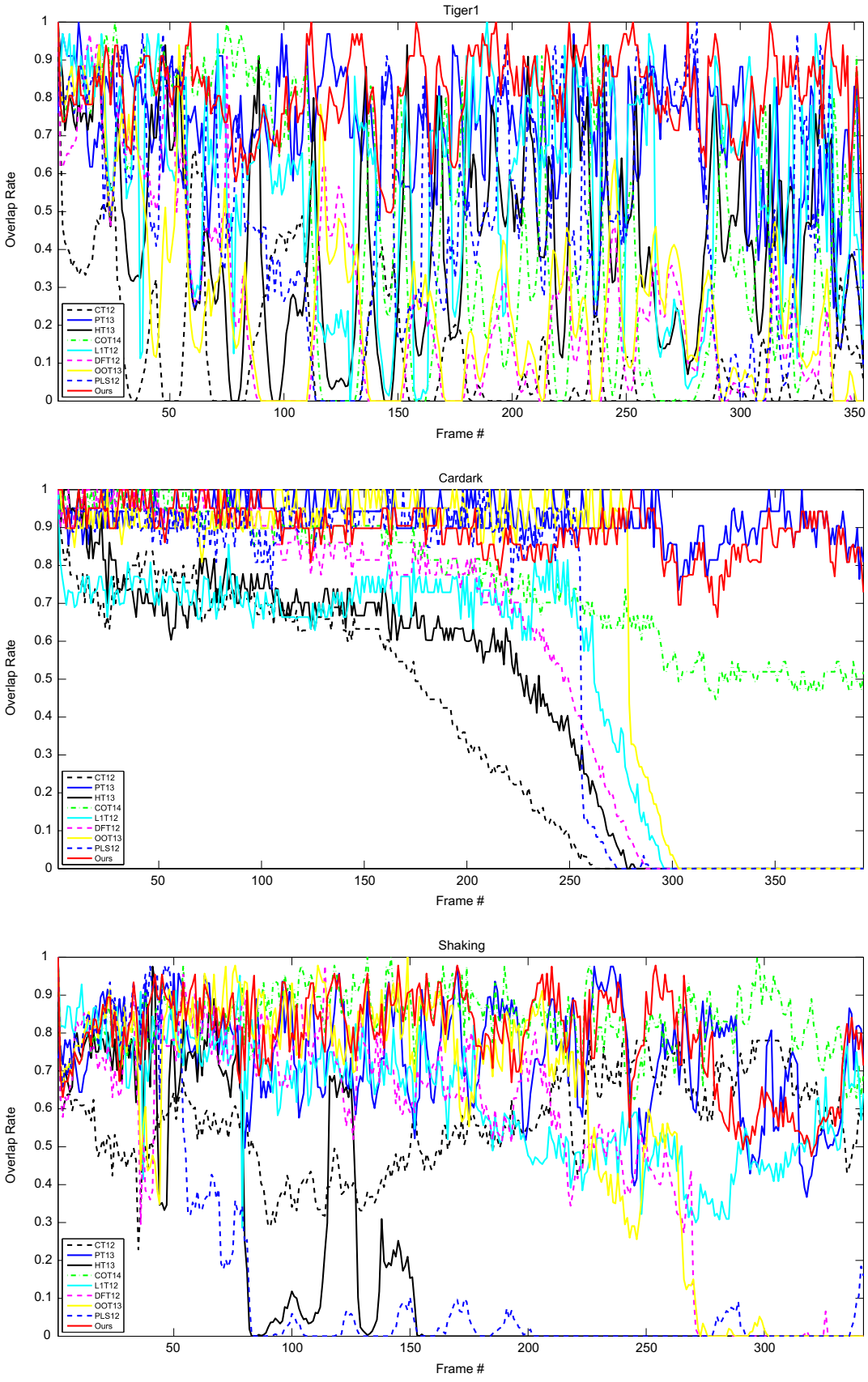

Fig. 11. (*continued*)

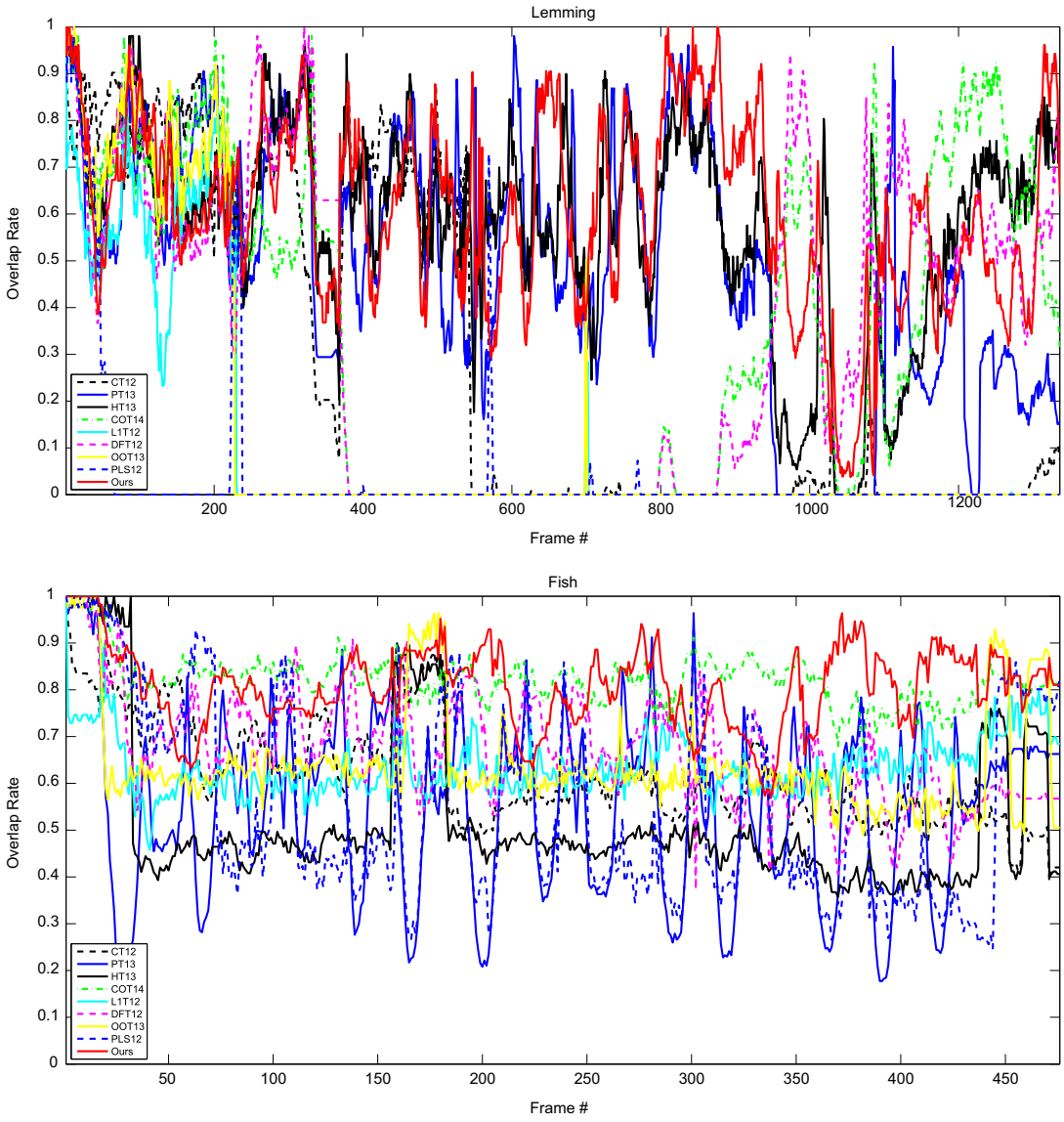

Fig. 11. (*continued*)

of being less robust. Therefore, we assign $\gamma=1/7$ as the optimal choice to have a good balance between robustness and efficiency.

6.2. Comparisons and evaluations

We shall now compare our tracking method with 8 state-of-the-art tracking methods via comprehensive experiments over 36 public, challenging video sequences (the details can be found in Tables 2 and 3). The 8 state-of-the-art tracking methods include the color tracking method (COT) [7], part-based tracking method (PT) [9], locality sensitive histograms based tracking method (HT) [28], online object tracking method (OOT13) [43], partial least squares tracking method (PLS12) [44], compressive sensing tracking (CT) [8], distribution field based tracking method (DFT) [45], and l_1 tracker (L_1T) [46]. All the quantitative comparison metrics in this paper are based on two widely used comparison metrics, which are the Center Location Error (CLE) and Overlapping Rate (OR). CLE is computed according to manually labeled ground truth at the pixel level. For each dataset, the CLE-based comparison results are detailed in our supplementary material, and the CLE-based performance and statistics are documented in Tables 4 and 5, while the average CLE comparisons of 36 video sequences are shown in Table 6. The CLE-based overall performance and statistics are provided in Table 7.

OR is be computed using

$$OR=\frac{\text{area}\{ROI_T\bigcap ROI_G\}}{\text{area}\{ROI_T\bigcup ROI_G\}}, \tag{25}$$

where ROI_T is the tracking rectangle of the given tracking method, and ROI_G is the ground truth. The OR-based comparison details can be found in our supplementary material. For each dataset, OR/SR (success rate, for the t-th video frame, if $OR_t>50\%$ then $SR_t=1$ else $SR_t=0$)) based performance and statistics are listed in Tables 8 and 9, and the average OR (SR) comparisons of 36 video sequences are documented in Table 10. The OR (SR) based overall performance and statistics are documented in Table 11.

Experiments over pose-deformable datasets: The target objects undergo lager-scale pose deformation in Girl, Bird-2, David, Tiger-1 and Lemming sequences (see details in Tables 2, 8 and Fig. 12). As we can see from Tables 4 and 8, that both COT and CT methods fail to update its basic classifier according to the dynamically changing target appearance (see Fig. 12, and more details in our supplementary material). Besides, the feature representation adopted by HT, L_1T, and DFT are all histogram-like, which tends to violate the intrinsic coherency

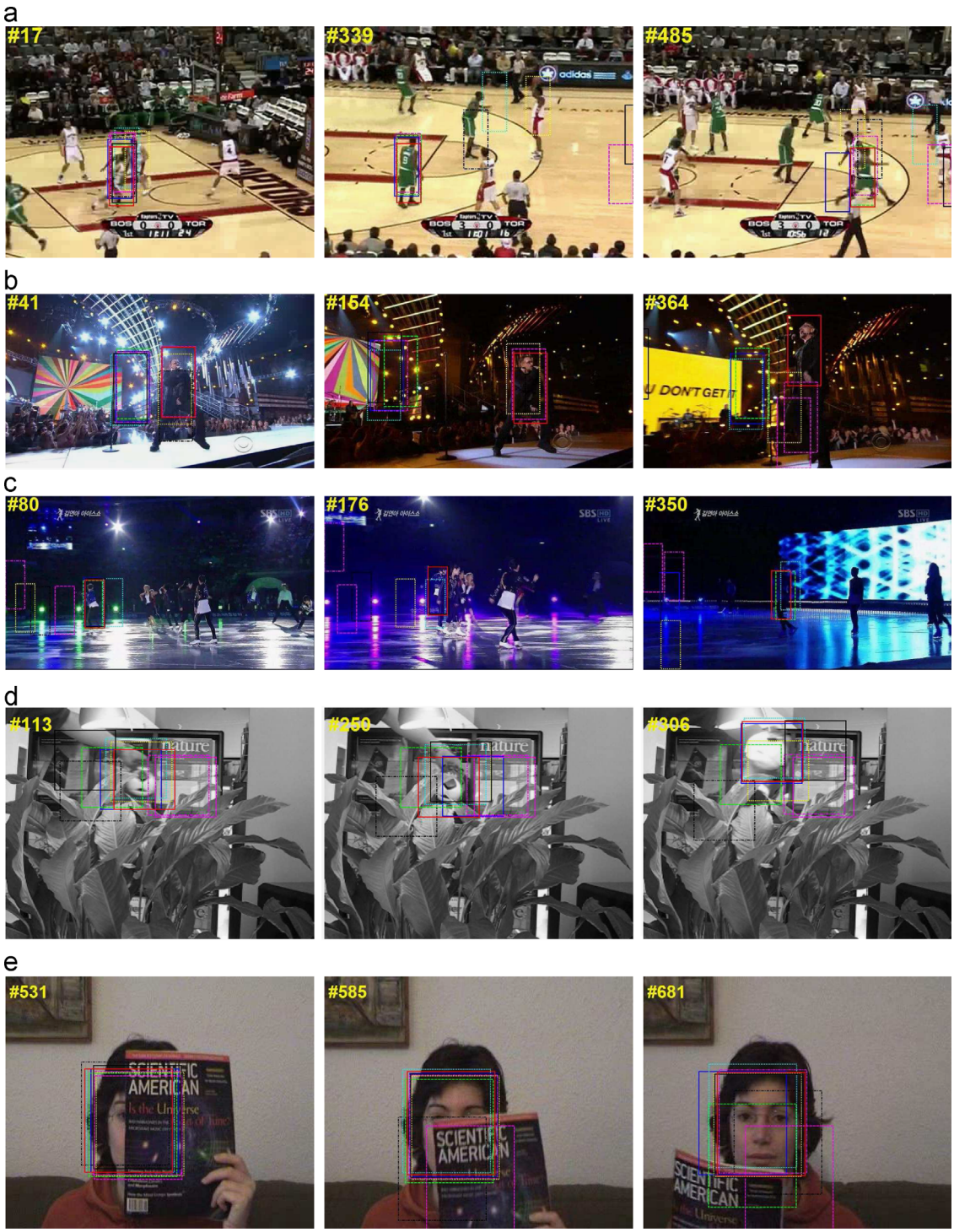

Fig. 12. Comparisons of the video object tracking results from different methods, please refer to our supplementary material for more and detailed results. (a) Basketball, (b) Singer2, (c) Skating1, (d) Tiger1, (e) OccludedFace1, (f) Shaking, (g) Lemming, (h) Coupon and (i) Woman.

when pose deforms. Although PT method also adopts the histogram-like feature representation, it can be regarded as searching the common occurrences between consecutive frames in feature space (which is slightly similar to our low-rank matching criteria), which separately matches the target's different parts with the appearance model and obtains favorable success rate (ranked as the second one in Table 10, and ranked as the third one in Table 11). However, due to the lack of a global constraint in PT, the drift problem still exists when 3D pose changes drastically, which leads to large CLE in Table 6. Specifically, the target object in Lemming sequence (961) undergoes 360° 3D pose movement, except for our method, the other methods gradually drift to background and produce unsatisfactory tracking results (Fig. 12(g)).

Experiments over illumination-varying and blur-varying datasets: The target objects experience strong illumination variation and blur in David, Tiger-1, Shaking, and Skating-1 datasets. Actually, the commonly used solution to handle such problems is to normalize the feature representation, which can globally maintain a relatively stable feature distribution. As shown in Fig. 12(f), the target object is undergoing heavy illumination change, and all the tracking methods can obtain correct tracking results. However, as

Fig. 12. (*continued*)

shown in Fig. 12(b) and (c), the illumination variation is usually coupled with partial appearance variation, and only our method and COT method can obtain correct tracking results. Actually, the robustness of COT method heavily depends on the color information, whose performance deteriorates rapidly when the tracking sequence only has grey information (see tracking results in Tiger-1 and David sequence). In Tiger-1 sequence (Fig. 12(d)), 3D pose change, illumination variation, and motion blur simultaneously occur, other methods lose track of the target object in numerous frames, however, our tracking method still performs well in terms of all metrics with only 5 CLE and achieve 99.1% success rate.

Experiments over object-occluded datasets: The target object in Occludedface-1, Occludedface-2, Woman and Skating-2 sequences has partial or heavy occlusion. In Fig. 12(e), all tracking methods except DFT and L_1T can handle the occlusion well. Nevertheless, it should be noted that the target objects in these two sequences are both being "*gradually*" occluded, which span 50 frames for the target object becoming fully occluded. Since this slow occlusion transition provides abundant buffering time for discriminative tracking to update their classifiers, the time span of the occlusion can be considered equivalently as certain type of the appearance variation, and favorable tracking result can also be obtained even when no occlusion handling strategy is invoked. However, this adaptiveness for appearance variation (occlusion) easily results in the drift problem, and unsatisfactory tracking results can found in Skating-2 (Tables 4 and 8). Specially, observing the CLE and OR statistics in Tables 7 and 11, our tracking method has obvious advantage when sudden occlusion occurs (fewer than 10 frames), please refer to Fig. 12 and the supplementary material for more robust object tracking results over Skating-1, Skating-2, Basketball, Tiger-1, Tiger-2, and Woman sequences.

Experiments over rapid-motion datasets: The target objects in Animal, Lemming, Football-1, Boy, Jumping, and Basketball sequences undergo fast movement (over 40 pixels for two consecutive video frames). The precondition of correct tracking for

fast movement is that the search area must cover the true target object. However, large constant search area results in heavy burden for the exhaustive matching procedure in CT, L_1T, PT, HT, and DFT, and their FPS drops rapidly (FPS is reduced by half) when the target object is fast moving. Benefiting from the coarse-to-fine tracking strategy, our low-rank coherency tracking can dynamically increase the search area while keeping the entire matching computation well under control (here, the upper-bound of the number of candidate targets is 450), and we achieve favorable FPS rate (slightly slower than COT, and see details in Table 12).

6.3. Limitation and discussion

Because our tracking method is based on the low-rank coherency analysis of different target observations, unsatisfactory tracking results may be produced when such low-rank coherency is broken due to drastic 3D shape change. As shown in the top row of Fig. 10, the box is undergoing fast 3D rotation, whose corresponding partial appearance model is demonstrated in the bottom row of Fig. 10 (see Fig. 10(a)). Meanwhile, since the similar parts of different observations have bias toward the middle-right position, the low-rank approximation of this partial appearance model is constant and similar to Fig. 10(b) until 192 frame arrives. However, a large difference can be found in the observation of 196 frame (where the low-rank coherency is broken), whose corresponding low-rank approximation is demonstrated in Fig. 10(c), and it finally results in drift (similar situation can also be found in Fig. 12(h)). The failure of low-rank coherency analysis may be caused by the absence of global spatial information in candidate representation, and thus it deteriorates the discriminative power of our feature space. One feasible way to combat this shortcoming is to introduce additional high-level global constraints to improve the matching criteria, which should be independent of the intrinsic feature space. Besides, at present our intrinsic feature space is derived from localized gray intensity only, so in order to achieve better tracking performance, color clues should also be taken into consideration by tightly coupling them with multi-level low-rank analysis. Both of these two limitations deserve our future investigation (Fig. 11).

7. Conclusion and future work

In this paper, we have proposed a simple yet effective video object tracking method based on rapid low-rank coherency analysis. Our robust and real-time tracking approach comprises many novel technical elements such as: (1) localized compressive sensing based representation method, which is both computationally more efficient and more sensitive than previously published methods; (2) low-rank coherency analysis based matching and updating criteria, which collectively enable robust visual tracking; and (3) natural integration of low-rank approximation and the tracking procedure, which enables real-time visual tracking. Comprehensive experiments and extensive comparisons with current state-of-the-art methods have demonstrated our method's salient advantages in terms of accuracy, reliability, robustness, and versatility.

Meanwhile, our method also has some limitations when low-rank coherency is broken due to the target's drastic and sudden rotation. Our ongoing efforts are first geared towards formulating high-level global constraints to further improve the matching criteria. Moreover, extending our key ideas to conduct subspace analysis based on low-rank coherency clustering for simultaneous multi-target tracking, motion segmentation, and content-based video retrieval also deserves our immediate research endeavors.

Conflict of interest

None declared.

Acknowledgments

This research is supported in part by National Natural Science Foundation of China (Nos. 61190120, 61190121, 61190125, and 61300067) and National Science Foundation of USA (Nos. IIS-0949467, IIS-1047715, and IIS-1049448).

Appendix A. Supplementary data

Supplementary data associated with this paper can be found in the online version at http://dx.doi.org/10.1016/j.patcog.2015.01.025.

References

[1] B. Benfold, I. Reid, Stable multi-target tracking in real-time surveillance video, in: IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 3457–3464.
[2] J. Hsieh, S. Yu, Y. Chen, W. Hu, Automatic traffic surveillance system for vehicle tracking and classification, IEEE Trans. Intell. Transp. Syst. 7 (2) (2006) 175–187.
[3] Y. Lu, L. Wang, R. Hartley, H. Li, D. Xu, Compressive evaluation in human motion tracking, in: Asia Conference on Computer Vision, 2010, pp. 177–188.
[4] A. Yilmaz, O. Javed, M. Shah, Object tracking: a survey, ACM Comput. Surv. 38 (4) (2006) 1–45.
[5] S. Hare, A. Saffari, P. Torr, Struck: structured output tracking with kernels, in: IEEE International Conference on Computer Vision, 2011, pp. 263–270.
[6] B. Babenko, M. Yang, S. Belongie, Robust object tracking with online multiple instance learning, IEEE Trans. Pattern Anal. Mach. Intell. 33 (8) (2011) 1619–1632.
[7] D. Martin, S. Fahad, F. Michael, V. Joost, Adaptive color attributes for real-time visual tracking, in: IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1090-1097.
[8] K. Zhang, L. Zhang, M. Yang, Real-time compressive tracking, in: European Conference on Computer Vision, 2012, pp. 864–877.
[9] R. Yao, Q. Shi, C. Shen, Y. Zhang, A. Hengel, Part-based visual tracking with online latent structural learning, in: IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 2363–2370.
[10] S. Wang, H. Lu, M. Yang, Superpixel tracking, in: IEEE International Conference on Computer Vision, 2011, pp. 1323–1330.
[11] B. Liu, J. Huang, L. Yang, C. Kulikowsk, Robust tracking using local sparse appearance model and K-selection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 1313–1320.
[12] E. Erdem, S. Dubuisson, I. Bloch, Fragments based tracking with adaptive cue integration, Comput. Vis. Image Understand. 116 (7) (2012) 827–841.
[13] A. Adam, E. Rivlin, I. Shimshoni, Robust fragments-based tracking using the integral histogram, in: IEEE Conference on Computer Vision and Pattern Recognition, 2006, pp. 798–805.
[14] X. Jia, H. Lu, M. Yang, Visual tracking via adaptive structural local sparse appearance model, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 1822–1829.
[15] D. Ross, J. Lim, R. Lin, M. Yang, Incremental learning for robust visual tracking, Int. J. Comput. Vis. 77 (1) (2008) 125–141.
[16] H. Li, C. Shen, Q. Shi, Real-time visual tracking using compressive sensing, in: IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 1305–1312.
[17] H. Grabner, H. Bischof, On-line boosting and vision, in: IEEE Conference on Computer Vision and Pattern Recognition, 2006, pp. 260–267.
[18] X. Jia, H. Lu, M. Yang, Online object tracking with sparse prototypes, IEEE Trans. Image Process. 22 (1) (2013) 314–325.
[19] H. Grabner, M. Grabner, H. Bischof, Real-time tracking via on-line boosting, in: British Machine Vision Conference, 2006, pp. 47–56.
[20] Z. Kalal, J. Matas, K. Mikolajczyk, P-N Learning: bootstrapping binary classifiers by structural constraints, in: IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 49–56.
[21] J. Gao, C. Chen, D. Zhen, Q. Zhu, An efficient version of inverse boosting for classification, Trans. Inst. Meas. Control 35 (2) (2012) 188–199.
[22] J. Henriques, R. Caseiro, P. Martins, J. Batista, Exploiting the circulant structure of tracking-by-detection with kernels, in: European Conference on Computer Vision, 2012, pp. 702–715.
[23] D. Bolme, J. Beveridge, B. Draper, Y. Lui, Visual object tracking using adaptive correlation filters, in: IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 2544–2550.

[24] A.L. Julia, M.B. Christopher, P.M. Thomas, Principled hybrids of generative and discriminative models, in: IEEE Conference on Computer Vision and Pattern Recognition, 2006, pp. 87–94.
[25] A. Radhakrishna, S. Appu, S. Kevin, L. Aurelien, F. Pascal, S. Sabine, Slic Superpixels, EPFL Technical Report, 2010.
[26] Y. Zhou, X. Bai, W. Liu, L. Latecki, Fusion with diffusion for robust visual tracking, in: Advances in Neural Information Processing Systems, 2012, pp. 2978–2986.
[27] F. Yang, H. Lu, W. Zhang, G. Yang, Visual tracking via bag of features, IET Image Process. 7 (2) (2012) 115–128.
[28] S. He, Q. Yang, R. Lau, J. Wang, M. Yang, Visual tracking via locality sensitive histograms, in: IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 2427–2434.
[29] W. Zhong, H. Lu, M. Yang, Robust object tracking via sparsity-based collaborative model, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 1838–1845.
[30] T. Zhang, B. Ghanem, S. Liu, N. Ahuja, Low-rank sparse learning for robust visual tracking, in: European Conference on Computer Vision, 2012, pp. 470–484.
[31] T. Zhang, B. Ghanem, S. Liu, N. Ahuja, Robust visual tracking via multi-task sparse learning, Int. J. Comput. Vis. 101 (2) (2013) 367–383.
[32] Y. Wu, J. Lim, M. Yang, Online Object Tracking: a benchmark, in: IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 2411–2418.
[33] M. Leordeanu, M. Hebert, A spectral technique for correspondence problems using pairwise constraints, in: IEEE International Conference on Computer Vision, 2005, pp. 1482–1489.
[34] X. Shen, Y. Wu, A unified approach to salient object detection via low rank matrix recovery, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 853–860.
[35] J. Yan, M. Zhu, H. Liu, Y. Liu, Visual saliency detection via sparsity pursuit, IEEE Signal Process. Lett. 17 (8) (2010) 739–742.
[36] R. Baraniuk, M. Davenport, R. DeVore, M. Wakin, A simple proof of the restricted isometry property for random matrices, Construct. Approx. 28 (3) (2008) 253–263.
[37] J. Wright, A. Yang, A. Ganesh, S.S. Sastry, Y. Ma, Robust face recognition via sparse representation, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2) (2008) 210–227.
[38] P. Li, T.J. Hastie, K.W. Church, Very sparse random projections, in: ACM SIGKDD, 2006, pp. 287–296.
[39] M. Fazel, E. Candes, B. Recht, P. Parrilo, Compressed sensing and robust recovery of low rank matrices, in: Proceedings of the 40th Asilomar Conference on Signals, Systems and Computers, 2008, pp. 1043–1047.
[40] F. Woolfe, E. Liberty, V. Rokhlin, M. Tygert, A fast randomized algorithm for the approximation of matrices, Appl. Comput. Harmon. Anal. 25 (3) (2008) 335–366.
[41] S. Roweis, Em Algorithms for PCA and SPCA, in: Neural Information Processing Systems, 1998, pp. 626–632.
[42] T. Zhou, D. Tao, Bilateral random projections, in: IEEE International Symposium on Information Theory Proceedings, 2011, pp. 1286–1290.
[43] D. Wang, H. Lu, M. Yang, Online object tracking with sparse prototypes, IEEE Trans. Image Process. 22 (1) (2013) 314–325.
[44] Q. Wang, F. Chen, W. Xu, M. Yang, Object tracking via partial least squares analysis, IEEE Trans. Image Process. 21 (10) (2012) 4454–4465.
[45] S.L. Laura, L.M. Erik, Distribution Fields for Tracking, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 1910–1917.
[46] C. Bao, Y. Wu, H. Ling, H. Ji, Real time robust L1 tracker using accelerated proximal gradient approach, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 1830–1837.
[47] J. Kwon, K. Lee, Visual tracking decomposition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 1269–1276.
[48] J. Santner, C. Leistner, A. Saffari, T. Pock, H. Bischof, PROST: parallel robust online simple tracking, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 723–730.
[49] D. Comaniciu, V. Ramesh, P. Meer, Kernel-based object tracking, IEEE Trans. Pattern Anal. Mach. Intell. 25 (5) (2003) 564–577.
[50] M. Godec, P.M. Roth, H. Bischof, Hough-based tracking of non-rigid objects, in: IEEE International Conference on Computer Vision, 2011, pp. 81–88.

Chenglizhao Chen received the M.S. degree in computer science from Beijing University of Chemical Technology, in 2012. He is currently pursuing the Ph.D. degree in Technology of Computer Application from Beihang University, Beijing, China. His research interests include pattern recognition, computer vision, and machine learning.

Shuai Li received the Ph.D. degree in computer science from Beihang University. He is currently an assistant professor at the State Key Laboratory of Virtual Reality Technology and Systems, Beihang University. His research interests include computer graphics, pattern recognition, computer vision, physics-based modeling and simulation, and medical image processing.

Hong Qin received the B.S. and M.S. degrees in computer science from Peking University. He received the Ph.D. degree in computer science from the University of Toronto. He is a professor of computer science in the Department of Computer Science, Stony Brook University. His research interests include geometric and solid modeling, graphics, physics-based modeling and simulation, computer-aided geometric design, visualization, and scientific computing. He is a senior member of the IEEE.

Aimin Hao is a professor in Computer Science School and the Associate Director of State Key Laboratory of Virtual Reality Technology and Systems at Beihang University. He received his B.S., M.S., and Ph.D. in Computer Science at Beihang University. His research interests are on virtual reality, computer simulation, computer graphics, geometric modeling, image processing, and computer vision.