Pattern Recognition 52 (2016) 410–432

Contents lists available at ScienceDirect

Pattern Recognition

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

Robust salient motion detection in non-stationary videos via novel integrated strategies of spatio-temporal coherency clues and low-rank analysis

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

[a] Beihang University, China
[b] Stony Brook University (SUNY), United States

ARTICLE INFO

Article history:
Received 13 May 2015
Received in revised form
18 September 2015
Accepted 27 September 2015
Available online 22 October 2015

Keywords:
Salient motion detection
Non-stationary videos
Aligned RPCA
Low-rank analysis
Semantic coherency
Monitor stabilization

ABSTRACT

Salient motion detection is vital for security surveillance, pattern and motion recognition, traffic control, human–computer interaction, etc. Although such a subject has been very well investigated for analysis of stationary videos, many technical challenges still prevail when correctly handling and analyzing non-stationary videos recorded by hand-hold and pan-tilt-zoom cameras. To ameliorate, this paper develops a novel and robust salient motion detection method (especially valuable for quantitative analysis of non-stationary videos) by employing new computational strategies, including low-rank analysis aided by the divide-and-conquer approach, and exploration of the space–time semantic coherency. The key idea in our new approach is to respectively conduct multi-purpose low-rank analysis over a temporal series of well-decomposed frame-batches that have relatively-consistent backgrounds. First, we conduct bilateral random projection (BRP)-based low-rank analysis to accurately keep track of short-term stable-background observations, which consist of frames with similar global appearance and small local variations. Then, to eliminate the side effects due to visual variations induced by view angle changes, we incorporate the low-rank background prior into previous short-term observation to guide robust principal component analysis (RPCA) low-rank revealing based robust salient motion detection over current short-term observation. Meanwhile, a series of saliency clues extracted from the stabilized short-term observations are leveraged to expedite the proper updating of the low-rank background information, which enables us to effectively combat several obstinate problems. Finally, we conduct comprehensive experiments on the public CD2014 benchmark and other five non-stationary videos recorded from the hand-hold camera, and make extensive and quantitative evaluations with six state-of-the-art methods. Experimental results indicate that our method not only outperforms all other methods in the case of non-stationary videos but also obtains outstanding performance for stationary videos.

© 2015 Elsevier Ltd. All rights reserved.

1. Introduction and Motivation

Salient motion detection is one of the most active research subjects in computer vision, which is extremely valuable in many subsequent applications, including object tracking [1], surveillance [2], traffic control [3], and intruder detection [4]. Despite the fact that the motion detection methods have achieved great success in recent years, many obstinate challenges still prevail for widely-used non-stationary videos [5], which may confine the application scope of motion tracking based intelligent systems in successfully coping with realistic and complicated scenarios, such as robotics and security monitoring. Therefore, there has been a strong expectation for more robust, stable, automatic, and versatile salient motion tracking methods.

Among the state-of-the-art salient motion detectors, modeling-based methods tend to have strong assumptions: the videos are obtained from stationary camera [6,7], and the pixels in the consecutive video frames have spatially-aligned coherency; so that they can directly employ the pixel-level background-pattern variations to distinguish the potential salient motions, such approaches include, but are not just limited to, Gaussian model [8], Gaussian mixture model (GMM) [9], and the extended GMM [10]. However, the contemporary modeling-based methods are still facing difficulties to properly handle relatively complicated situations with camera jitter compounded by dynamic backgrounds. Meanwhile, other state-of-the-art motion tracking methods follow the tracking-by-detection strategy [11], and they tend to frame-wisely detect the salient motions from the background via

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

http://dx.doi.org/10.1016/j.patcog.2015.09.033
0031-3203/© 2015 Elsevier Ltd. All rights reserved.

pixel-wise correspondence matching among consecutive video frames [12]. However, on the one hand, such methods usually require to manually initialize the target motions as well as limiting their tracking numbers, and the observation models have to be elaborately tuned; on the other hand, the incorrect matchings caused by occlusion can easily lead to false-alarm detection. To improve, low-rank analysis based methods in recent years began to gain momentum with ever-increasing interest from researchers in the salient motion detection field [13,14]. The principle behind it is that, the low-rank coherency not only implies the intrinsic co-occurring backgrounds among cross-frame observations, but also can serve as the matching criteria of current-frame candidate motions, and thus, the involved salient motions can be regarded as the sparsity derived from low-rank decomposition. Despite recent research advances, for more complex non-stationary videos lasting for a longer period of time, naively using low-rank analysis still encounters certain difficulties for the task of meaningful and robust salient motion detection. In particular, the key technical challenges are highlighted as follows.

First, the long-term robust motion tracking in unconstrained videos remains to an open research problem, especially for long-term non-stationary Pan-Tilt-Zoom (PTZ) video, it tends to contain multiple continuously-varying scenarios, wherein the backgrounds should exhibit stronger scale and affine variations due to camera jitter and view/pose change. Since the traditional low-rank motion detection mainly focuses on the background observation modeling in a global perspective, it will fail to handle such time-varying relationships between the salient motions and the non-aligned backgrounds. Therefore, intuitively speaking, we need a smart divide-and-conquer computing strategy to assist multiple local region-based low-rank analysis for visual clue variation, which is required to the smooth coherency transitions among different scene contents.

Second, although human beings are sensitive to the objects-of-interest with salient appearance/motion relating to the uniqueness, rarity, and surprise of a scene, and the classical robust principal component analysis (RPCA) based low-rank analysis model well follows this physiological principle, which assumes that the salient motions are sparse with respect to the low-rank structured background. However, even if locally-stationary video is given, this sparse assumption may not always conform to the physical reality due to the dynamically-changing appearance pattern of the background and embedded objects, illumination variation, occlusion, etc. Therefore, the already-powerful low-rank based motion saliency model should be further improved to closely couple with the adaptive learning of high-level physiological priors.

Third, robust salient motion detection requires to selectively localize noticeable motions in a scene, however, the existing low-rank trackers commonly ignore the contextual interactions between the intermittent moving targets and the background [13] due to the high computational cost of online maintenance of the context model. Therefore, it is urgently expected to efficiently integrate the spatio-temporal coherency cues into an improved low-rank model by conducting online exploration over the complementary color, structure, and sparse residuals.

To tackle the aforementioned challenges encountered by state-of-the-art low-rank models, our current research endeavors are aiming at the robust salient motion detection from non-stationary videos. In this paper, we advocate a series of novel computational strategies to seamlessly integrate the stable background priors and spatio-temporal coherency clues into a generalized yet much-improved low-rank analysis model. We propose to perform high-level background tracking first to adaptively obtain short-term frame batches with relatively-consistent backgrounds, and then we explore the salient motions via low-rank analysis based background modeling. Meanwhile, inspired by our previous works [1,15], we continue to improve the global compressive sensing based object tracker to afford local low-rank swarm voting based background tracking. In addition, we aim to tackle the nontrivial challenges caused by feature deviations by incorporating the exploited motion coherency occurred in the consecutive frame batches into low-rank analysis. In particular, for each frame batch, we propose to introduce low-rank background prior obtained from previous frame batch into the aligned RPCA low-rank revealing process, wherein multiple saliency clues are simultaneously employed to guide low-rank prior updating. Consequently, our method can take full advantages of both modeling-based methods and matching-based methods, collectively to improve the state-of-the-art performance. Specifically, the salient contributions of this paper towards novel computational strategies can be summarized as follows:

- We propose a versatile and robust background tracking method to decompose original long-term non-stationary videos into a series of short-term frame batches with relatively-consistent backgrounds, which can expedite the intrinsic temporal low-rank information revealing.
- We integrate low-rank background prior with coherency revealing based on the aligned RPCA low-rank analysis, which can guarantee to effectively eliminate the side effects caused by dynamically-changing background and camera jitter.
- We define a series of saliency clues via online exploration of the spatio-temporal coherency over the residual sparsity derived from low-rank decomposition, which naturally gives rise to adaptive the updating of the low-rank background prior for complex scenes with intermittent motions and occasionally-occluded sub-regions serving as backgrounds.

2. Related work

2.1. Background modeling methods

Since the backgrounds of stationary video tend to stay changeless in general, modeling methods are usually adopted to represent the intrinsic feature distribution of the background, and regard its corresponding deviations as the salient motions. Wren et al. [8] adopted single Gaussian model [16] to model the object's intensity distributions. However, because of the existence of background variation, it is infeasible to represent the entire background status solely on single Gaussian. Therefore, GMM [17,9] is adopted to model the complex background. Moreover, Zivkovic et al. [18] proposed to use recursive manner to update parameters of the GMM model adaptively. Although this method can well handle variations of the background, its pixel-level modeling has encountered massive false-alarm detections. To boost the detector's robustness against sudden background variations, recent works tend to concentrate on the exploration of temporal coherency [19,20] and spatial information. Varadarajan et al. [10] proposed to model regions as mixture distributions rather than a collection of individual pixels. Similarly, Bilodeau et al. [21] proposed to perform the salient motion detection in feature space spanned by the local binary similarity patterns (LBSP) descriptor, which exhibits high discriminative power than traditional pixel-wise color information and is robust to noises. Different from the regional strategy (in LBSP), Liang et al. [22] proposed to explore spatial information by seeking local pixel's co-occurrence to model the background in a pixel-pair manner. However, because dynamic backgrounds frequently share similar Gaussian deviations with respect to the moving object, these methods tend to be very vulnerable to massive false-alarm

detections, which are mainly caused by mistakenly classifying dynamic backgrounds as salient motions [23]. Therefore, more and more research works suggest that sparse templates or historical intensity variations based background modeling can further improve the detection results. Thus, StCharles et al. [6] proposed pixel-level feedback strategies to adaptively adjust internal parameters without any prior intervention. Maddalena et al. [24–26] proposed to use artificial neural networks to model the background, which has demonstrated robust performance against motions with frequent interruptions and illumination variations. However, these methods need plenty of training periods to estimate the neural networks' parameters, thus Gregorio et al. [27,28] proposed the weightless neural networks to boost the training period. Most recently, Wang et al. [7] integrated the flux tensor with both foreground model and background model and achieved the best performance so far.

2.2. Matching based motion recognition methods

For non-stationary videos, due to the absence of direct correspondences among consecutive frames, matching based methods tend to integrate correspondences into motion sensors by resorting to discriminative descriptors. Dollar et al. [29] proposed to build cross-frame patch-wise correspondences via temporal coherency, which achieves much stronger discriminative power than pixel-level solutions. Similarly, Liu et al. [11] employed multiple interest point detectors (Harris Laplacian, Hessian Laplacian, and MSER [30]) and strong local descriptor (like SIFT) to span the feature space with higher discriminative power. In addition, based on the temporal motion clues proposed in [29], Liu et al. [11] adopted the page ranking algorithm [31] to prune those false-alarm interest point-pairs, and achieved robust region of interest (ROI) detection for non-stationary videos. Vijay et al. [4] extended traditional pixel-level matching to region-level registration. By convolving with the blur kernel, the differences between these registered regions are taken as the salient motions. Meanwhile, some works employ the camera's motion clues [32] to guide the detection. Kim et al. [33] proposed to use Gaussian regression to represent camera's motion tendency as a stochastic vector field, which can detect the regions of interest (ROIs) in non-stationary videos. Xu et al. [12] found that the robotic's motor signals are heavily correlated to the background motion, and they adopted learning based solution to learn fundamental matrices as functions of motor signals to facilitate the salient motion detection. Recently, instead of one-to-one matching based detection, from the perspective of visual saliency, Kim et al. [34] proposed to leverage the specifics of motion clues to determine salient motions, which explores spatio-temporal directional coherence at each pixel position via well-designed local gradient field. Similarly, Fang et al. [35] adopted multiple features, which perform data fusion of illumination, color, and texture feature with motion features (extracted from the motion vectors), to carry out video saliency computation in a compressed feature domain. Although these methods have achieved great success over ROI detection or motion recognition, the specific application toward salient motion detections is rarely developed.

2.3. Low-rank revealing methods

The main task of low-rank revealing is to recover the low-rank information from corrupted observations, which decomposes the input matrix into the low-rank part and the sparse part. Two most appealing methods to solve this problem are robust principal component analysis (RPCA) [36] and bilateral random projection (BRP) [37], and in practice the accuracy of the RPCA is better than that of BRP, yet its computation cost is much higher. As for the salient object detection problem, due to the nature of the visual saliency, variations resulted from the contrast to its surroundings can be regarded as the most trustworthy saliency clue. Yan et al. [38] proposed to perform RPCA on color information spanned feature space, and the residuals in sparse matrix are directly regarded as the saliency degree. Similarly, Shen et al. [39] proposed the learning based transform to constrain the RPCA based low-rank structure toward the backgrounds. Our previous work [15] attempts to explore the multi-scale saliency degree by way of varying the BRP rank level. As for the object tracking problem, because the appearance of the target object varies over time, the low-rank information of previous appearances can also be regarded as the trust-worthy indicator to locate the target object in the current frame. Zhang et al. [40] proposed to incorporate the low-rank constraints into the appearance model learning process. In a more direct fashion, our previous work [1] adopts the low-rank information of previous target templates as the “low-rank coherency” clue to locate the target object. Since the low-rank information of previous target appearance represents the most common patterns, the frequent occlusion or appearance varying induced drift problem can be well handled achieving robust object tracking results. However, as for the background tracking problem mentioned in this paper, these kinds of object tracker cannot perform well because of the short-term nature of non-stationary videos which need frequent initialization to capture frame batches with stable backgrounds breaking the “low-rank coherency”. Most recently, low-rank based background modeling schemes become prevalent [13,14] in salient motion detection field, which can suppress dynamic background deviations much better than the template based solutions. Zhou et al. [13] proposed to perform low-rank revealing on those pre-aligned video frames, and the residuals to the established low-rank structure are regarded as the salient motions. Following the original work by Gao et al. [14], to further restrain dynamic backgrounds, the frames' pre-alignment steps are introduced into the low-rank revealing process, and multiple low-rank revealing processes (e.g., two-pass RPCA) are also adopted to carry out the coarse-to-fine computational strategy in order to eliminate false-alarm detections brought by the dynamic backgrounds. Moreover, to pursue robust salient motion detections in a global manner, [14] performs the low-rank revealing over the entire input video sequences, which severely adds heavy burden on both memory and computation (FPS < 0.05), and such a method is incapable of handling non-stationary videos. The time consuming characteristic of this method is mainly brought by two components: the “two-pass” RPCA; and the alignment steps to handle camera jitter problems. Because [14] does not adopt the background tracking strategy, for the camera jitter category, the variance of the initial input frames is very large, which causes the alignment steps to be extremely time consuming. Specifically, although [13] can detect the salient motions for non-stationary videos in certain special cases (with user intervention), challenges such as spatio-temporal overlapping induced hollow effects and ghost effects are not yet taken into consideration, which gives rise to our motivation aiming to overcome these limitations.

2.4. Brief summary

In general, according to the comprehensive evaluations recently performed by Goyette et al. [41,5], modeling based salient motion detection methods are already capable of tackling most of the conventional challenges, and as a result, they frequently outperform matching based methods for the process of stationary videos. However, the situation becomes odd for non-stationary

videos. When matching based methods are used to handle non-stationary videos, suffering from the discontinuous nature of the motions in the wild, they tend to wrongfully take those motions with broken coherency (e.g., occlusions and intermittent moving) as the non-salient backgrounds. Therefore, strongly inspired by the aforementioned methods, we plan to decompose non-stationary long-term videos into stationary short-term frame batches via background tracking, and then conduct low-rank analysis, assisted by a series of saliency clues, to obtain robust salient motion detection. And the high-level overview of our method is described in the following section.

3. Method overview

As shown in Fig. 1, our method mainly consists of three components: low-rank background tracking (Fig. 1(a), (c) and (d)), background prior guided low-rank analysis (Fig. 1(b), (e), (f) and (g)), and background saliency clues based prior updating (Fig. 1(h), (i) and (j)). In strong comparison with traditional modeling based methods, our method emphasizes to leverage background tracking for the convenient conversion of the original input video sequence into frame batches with relatively-stable background (which still affords small-scale affine variations induced by view angle

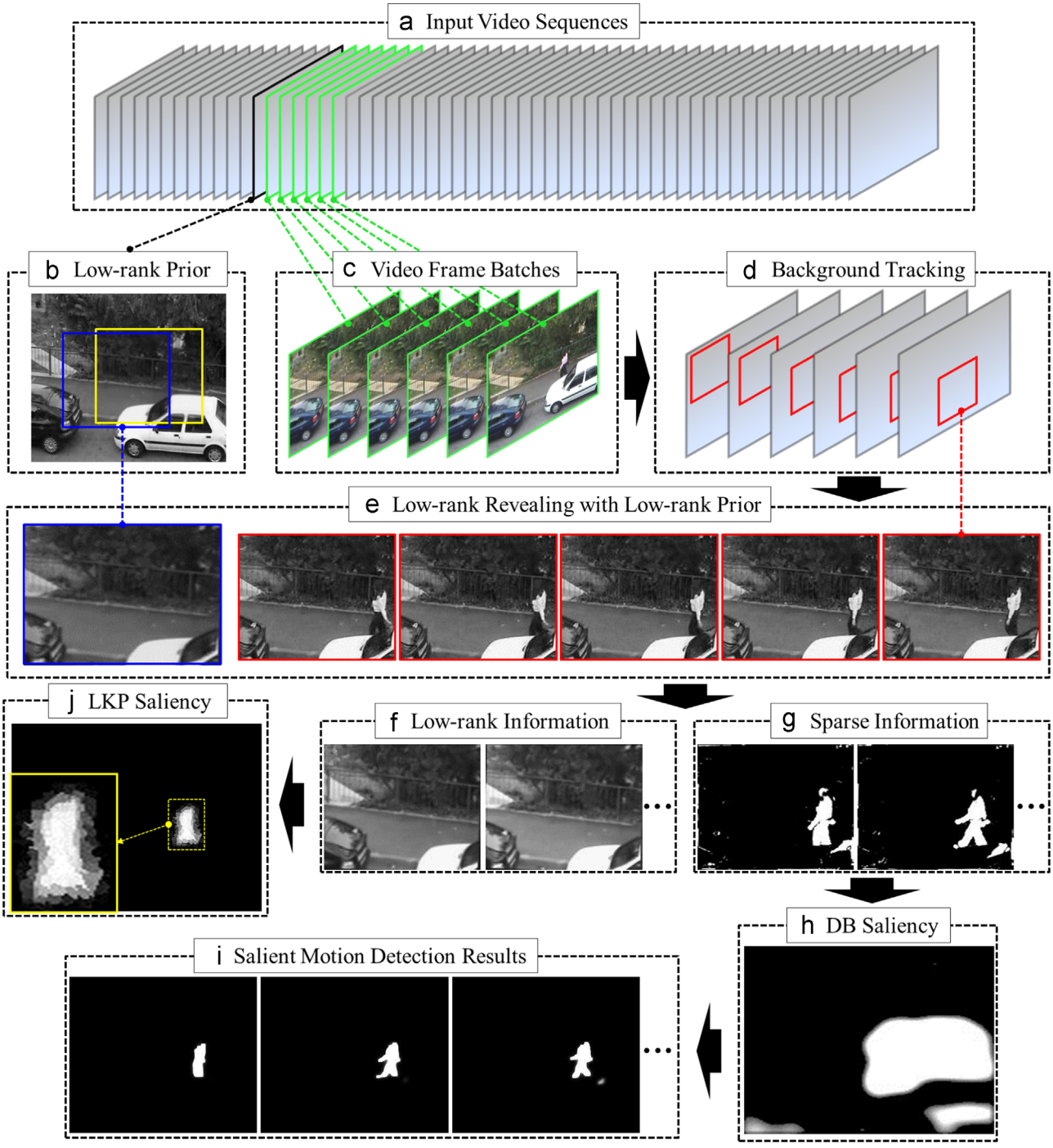

Fig. 1. Computational architecture of our salient motion detection. The yellow rectangle in (b) indicates the previous low-rank information, and the blue rectangle indicates current low-rank prior. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

changes), and the details are discussed in Section 5. To avoid hollow effects that frequently occur in traditional low-rank based methods [13], we further integrate previous frame-batches' low-rank prior into the current batch's low-rank revealing process (Fig. 1(b)), wherein the current frame batch is decomposed into low-rank parts (biased toward the low-rank prior) and sparse parts (Fig. 1(e)). Meanwhile, because the salient motions usually correspond to certain changes or variations, the column-wise L_1-norm of the sparse information is utilized to measure the motion saliency (Fig. 1(g)), and the details are discussed in Section 6. Besides, we introduce the dynamic background saliency clues to guide the saliency-value assignments in dynamic background regions (Fig. 1(h)). And the update of the current low-rank information, which will be used to guide the low-rank revealing of next subsequent frame batch, is guided by the LKP saliency clue (Fig. 1(j)) to avoid ghost effects (Fig. 6(b)), and the details are documented in Section 7.

To assist readers to fully understand our mathematical formulations, Table 1 summarizes the mathematical symbols used in the following sections, wherein normal-case letters denote scalars, bold lower-case letters denote vectors, and bold upper-case letters denote matrices.

4. Brief review of the low-rank revealing methods

The low-rank revealing aims to decompose the original input matrix $\mathbf{A}$ into a low-rank component $\mathbf{L}$ and a sparse component $\mathbf{S}$, that is $\mathbf{A}=\mathbf{L}+\mathbf{S}$. Thus, the problem formulation can be defined as

$$\min_{\mathbf{L},\mathbf{S}} rank(\mathbf{L})+\lambda\|\mathbf{S}\|_0 \quad s.t. \quad \mathbf{A}=\mathbf{L}+\mathbf{S}. \tag{1}$$

Although Eq. (1) represents a highly non-convex optimization problem (which is also NP-hard), it can be approximately solved by its relaxing convex envelope via

$$\min_{\mathbf{L},\mathbf{S}} \|\mathbf{L}\|_*+\lambda\|\mathbf{S}\|_1 \quad s.t. \quad \mathbf{A}=\mathbf{L}+\mathbf{S}. \tag{2}$$

Here $\|\cdot\|_*$ indicates the nuclear norm of $\mathbf{L}$. Two representative methods to solve the low-rank decomposition problem are RPCA [36] and BRP [37]. We shall briefly review these two methods below.

4.1. The RPCA based low-rank revealing method

The key solution of the RPCA low-rank revealing is consisting of two steps: the singular value thresholding based low-rank component estimation (Eq. (3)), and the soft thresholding based sparse component computation (Eq. (4)):

$$\mathbf{L}\leftarrow\mathbf{U}[\Sigma-\mu\mathbf{I}]_+\mathbf{V}, \quad (\mathbf{U},\Sigma,\mathbf{V})\leftarrow svd(\mathbf{Y}_1), \tag{3}$$

$$\mathbf{S}\leftarrow \text{sign}(\mathbf{A}-\mathbf{S}-\mathbf{L})[|\mathbf{A}-\mathbf{S}-\mathbf{L}|-\lambda\mu]_+, \tag{4}$$

where $\mathbf{I}$ denotes the identity matrix, $\mathbf{A}$ denotes the original input matrix, and the details of the remaining symbols are identical to what have been explained earlier in our manuscript. Obviously, the RPCA low-rank revealing iterates these two steps to gradually establish both the low-rank component $\mathbf{L}$ and the sparse component $\mathbf{S}$. In summary, the main characteristic is its good low-rank revealing performance in spite of being a bit time-consuming, whose slow convergency speed is mainly caused by the SVD steps and the data fidelity matrix $\mathbf{Y}_1$, $\mathbf{Y}_1\leftarrow\tilde{\mathbf{L}}-\frac{1}{2}(\tilde{\mathbf{L}}+\tilde{\mathbf{S}}-\mathbf{A})$, and $\tilde{\mathbf{L}},\tilde{\mathbf{S}}$ can be updated via the following equation:

$$\tilde{\mathbf{L}}_j\leftarrow\mathbf{L}_j+\frac{t_{j-1}-1}{t_j}(\mathbf{L}_j-\mathbf{L}_{j-1}), \quad \tilde{\mathbf{S}}_j\leftarrow\mathbf{S}_j+\frac{t_{j-1}-1}{t_j}(\mathbf{S}_j-\mathbf{S}_{j-1}), \tag{5}$$

where the details of the parameter t_j can be found in Algorithm 2.

4.2. The BRP based low-rank revealing method

The key idea of BRP method is to use the approximated random matrix based projection to boost the convergency speed of the low-rank revealing process. The size of the initial random matrix $\mathbf{Y}_2$ is supposed to be *rank* × *cards*, wherein the *rank* and the *cards* are separately used to control the low-rank degree and the sparse degree. The BRP low-rank revealing mainly consists of two steps: the bilateral random projection based low-rank matrix $\mathbf{L}$ revealing, and the entry-wise hard thresholding to compute the sparse matrix $\mathbf{S}$:

$$\mathbf{L}\leftarrow(\mathbf{L}\times\mathbf{Q})\times\mathbf{Q}^T, \tag{6}$$

$$\mathbf{S}\leftarrow\Omega(\mathbf{A}-\mathbf{L}), \tag{7}$$

where $[\mathbf{Q},\mathbf{R}]=qr(\mathbf{Y}_2)$, qr denotes the QR decomposition, $\mathbf{Y}_2\leftarrow\mathbf{L}^T\times(\mathbf{L}\times\mathbf{Y}_2)$ is the power scheme (similar to the power iteration [42] with fixed iteration times) to boost the convergency speed, $\Omega(\cdot)$ denotes the function to select the largest top *cards* elements from $|\mathbf{A}-\mathbf{L}|$. In summary, the main characteristic of BRP based low-rank revealing is its fast computation, but its accuracy is much worse than the RPCA based method. Moreover, the BRP based low-rank revealing method requires the user to provide the approximated *rank* and *cards*, which heavily depends on the user experience, and thus limits its broad applications.

Table 1
The 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, $m=W\times H$
T	The number of tracking candidates
$\mathbf{R},\mathbf{X},\mathbf{V}$	Measurement matrix, original input matrix, and feature matrix
$\mathbf{O}$	Occlusion mask
k	The upper bound of the tracked background frame number
w	The voting confidence of sub tracker
ω,ψ,η	The local/global saliency indicator and the local scope constraint
ε	The penalty coefficients for sub tracker coincidence
λ	The penalty coefficients for low-rank decomposition sparsity degree
β	The compensation strength along the gradient direction
$\mathbf{A},\mathbf{L},\mathbf{S}$	Appearance model, the low-rank matrix, and the sparse matrix
$G,\mathbf{J}$	Frame batch and Jacobian matrix
τ,ϵ,μ	2D transform, standard basis, and a small constant
$UT_{1,2},DT_{1,2}$	Hard thresholds to compute the dynamic background mask

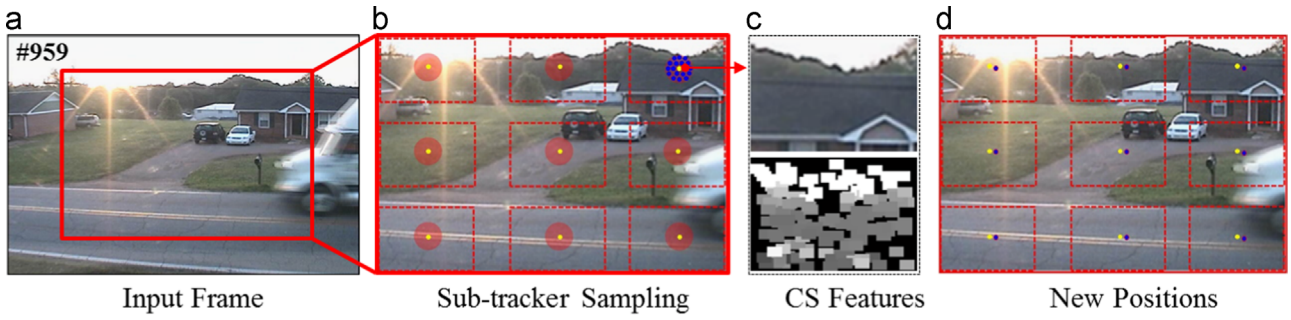

Fig. 2. Illustration of the key steps for background tracking. The yellow dots in (b) denote the tracking results in the previous frame, red cycle indicates the locally-searching area, and the blue points in (d) denote the newly-tracked sub-background locations. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

5. Background tracking

Since the backgrounds of non-stationary video usually vary dramatically, it is difficult to perform salient motion detection on frames with continuously-changing backgrounds. However, due to the existence of semantic coherency, background variations in short-time consecutive frames are oftentimes limited. Therefore, we propose to use localized bilateral random projection (BRP) tracking to decompose the input video frames into subgroups (frame batches) sharing relatively-stable backgrounds.

5.1. Localized background tracking

Following our previous work [1], we adopt localized compressive sensing (CS) based feature representation (Eq. (8) and Fig. 2(c)) to represent the "target background" in a patch-wise manner efficiently:

$$\mathbf{V} = [v_1, v_2, \ldots, v_n]^T = \frac{1}{Z}(\mathbf{R} \odot h)\mathbf{X}, \tag{8}$$

where $\odot$ denotes the column-wise OR operation, Z is the normalization factor, $\mathbf{V} \in \mathbb{R}^{1\times n}$, $\mathbf{R} \in \mathbb{R}^{n\times m}$ is the constrained random Gaussian matrix, $\mathbf{X} \in \mathbb{R}^{1\times m}$, $m = W \times H$, $n \ll m$, and the formulation of the rectangle filter h can be defined as

$$h_{i,j}(x,y) = \begin{cases} 1, & i \le x \le (i+\gamma W),\ j \le y \le (j+\gamma H) \\ 0 & \text{otherwise} \end{cases}, \tag{9}$$

where W and H denote the rectangle width and height respectively, and γ is set to be 1/7.

Supposing the ith video frame as the start of current observation, we use V_i to denote the locally-constrained CS-based representation of the target background patch, and the appearance model of the background (denoted by $\mathbf{A}$) can be initialized as

$$\mathbf{A} \leftarrow [\mathbf{A}\ V_i], \quad s.t. \quad C(\mathbf{A}) \le k, \tag{10}$$

where $\mathbf{A} \in \mathbb{R}^{n\times 1}$, n is the feature dimension of V_i, and $C(\cdot)$ indicates the column size of the matrix $\mathbf{A}$. Since there exist strong correlations among consecutive video frames, the low-rank part $\mathbf{L}$ of $\mathbf{A}$ can be regarded as the intrinsic background clue to locate background patch in the $(i+1)$th frame, which can obtained via

$$BRP([\mathbf{A}\ \mathbf{B}]) = \mathbf{L} + \mathbf{S}, \tag{11}$$

where $BRP(\cdot)$ denotes the bilateral random projection (BRP) based low-rank decomposition, $\mathbf{L}$ and $\mathbf{S}$ respectively indicate the low-rank part and sparse part, $\mathbf{B} = [V_{(t,1)}, V_{(t,2)}, \ldots, V_{(t,T)}]$, $V_{(t,j)}$ indicates the jth background patch candidate in the tth video frame (blue points in Fig. 2(b)), T is the number of the candidates. Obviously, the l_0-norm of $\mathbf{S}$ matrix can be regarded as the metric of the true background candidate, which has the minimum feature distance with respect to the previous intrinsic background feature pattern, and $V_{(t,j)}$ will be selected as the tracking result only if $\sum_{p=1}^{n} |\mathbf{S}_{p,j}| = \min_i \sum_{p=1}^{n} |\mathbf{S}_{p,i}|$ (see blue points in Fig. 2(d)).

In [1], the partial appearance model $\tilde{\mathbf{A}}$, which is formulated by selecting $0.2 \times k$ observations from $\mathbf{A}$ according to $\|S\|_1$ in an ascending order, is regarded as the clue to compute the novel low-rank prior (via another BRP low-rank revealing) to refine the current tracking result. Meanwhile, similar procedures (via performing BRP low-rank revealing over the entire $\mathbf{A}$) are adopted to control updating the appearance model $\mathbf{A}$. In order to effectively track relatively-stable background patches within limited k frames (see details in Section 5.2), the object appearance model and the updating strategy used in [1] should be further simplified to accommodate frequent initialization, because they are originally designed to facilitate the long-term tracking. That is, instead of using the partial appearance model $\tilde{\mathbf{A}}$ to perform the tracking refinement, we directly adopt the appearance model $\mathbf{A}$ with much smaller search radius (i.e., random sparse sampling) to refine the tracking results. Meanwhile, instead of using BRP low-rank revealing guided appearance updating, we directly use currently-tracked target observations to replace the oldest records in the appearance matrix $\mathbf{A}$.

In fact, except for the low computational cost, neither the tracking precision nor robustness of this simplified tracker is remarkable. Fortunately, the spatial information among the sub-backgrounds tends to remain constant within limited consecutive frames, we can utilize such information to facilitate the background tracking, and we will detail them in next section.

5.2. Robust global background tracking

Based on the localized background tracking, for each input video frame, we regard the corresponding sparse matrix $\mathbf{S}$ as the current tracking indicator. To fully utilize the spatial information among different sub-backgrounds, the most practical strategy is to employ sub-background-wise multiple trackers simultaneously and seek the globally optimal tracking result according to the tracking displacements of these trackers. The underlying principle is that the background commonly undergoes mild variations in limited consecutive video frames, such as small-scale affine transformation and camera jitter induced blur.

Therefore, we introduce a global constraint to each individual sub-background tracker (Fig. 2(b)), which guarantees that each tracked sub-background possesses similar displacements and low-degree sparse residual in $\mathbf{S}$. Here the optimization function subject to the global constraint for the tth video frame is defined as

$$\arg\min_{x,y} \sum_{i=1}^{u} \left(w_i^t \cdot \sum_{j=1}^{n} O_j^t \odot |\mathbf{S}_{i,j}^t(x_i, y_i)| \right) - \varepsilon \cdot \left\| \sum_{i=1}^{u} \vec{\boldsymbol{d}}_i \right\|_2, \tag{12}$$

where u is the sub-tracker number (we set it to 9 in this paper), n denotes the feature dimension, $w_i^t = 10/Z \cdot \exp(-\sum O_i^{t-1} \odot |S_i^{t-1}|)$ is a weighting parameter, $\mathbf{S}_{i,\cdot}^t(x_i, y_i)$ denotes the candidate patch centering at (x_i, y_i) of the ith sub-tracker, ε is the parameter to control the influence of the displacement penalty term, $O \in \mathbb{R}^{1\times n}$ denotes the occlusion mask, which has large value for those potential occluded regions. And $\vec{\boldsymbol{d}}_i = (x_i^t - x_i^{t-1}, y_i^t - y_i^{t-1})/\|(x_i^t - x_i^{t-1}, y_i^t - y_i^{t-1})\|_2$ is the individual tracker's displacement between two consecutive frames, and $\|\cdot\|_2$ denotes the l_2-norm, $x = [x_1, x_2, \ldots, x_u]$ and $y = [y_1, y_2, \ldots, y_u]$ indicate the tracked center location of each sub-tracker. Obviously, the first term of Eq. (12) guarantees that the global tracking result should exhibit the minimum feature variation with respect to the intrinsic low-rank pattern of **A**, andthe second term concentrates on the displacement coincidence of each sub-tracker.

Algorithm 1. Numerical implementation of Eq. (11).

Input: Sparse matrix **S**; Sub-tracker weight w; Occlusion mask **O**;
The ith sub-tracker's previous position (x_p, y_p).
Output: The newly detected position of the ith sub-tracker (x_i, y_i).
Initialization: $\varepsilon = 0.4$; Total number of sub-tracker: $u = 9$; The confidence score of the kth target candidate:
$\vartheta_k^0 = \mathbf{O}_{i,\cdot}(x_k, y_k) \odot |\mathbf{S}_{i,\cdot}(x_k, y_k)|, i \in \{1, 2, \ldots, u\}$.

For $t = 1{:}5$
1. Search the local minimum position of the ith sub-tracker:
$(x_i, y_i) = \min_k(\vartheta_k^{t-1}), \quad k \in \{1, 2, \ldots, 150\}$.
2. Compute the displacement direction vector of the ith sub-tracker:
$\vec{\boldsymbol{d}}_i = (x_i - x_p, y_i - y_p)/\|(x_i - x_p, y_i - y_p)\|_2$.
3. Eliminate the influence from occlusion via restraining the largest displacement:
$\vec{\boldsymbol{d}}_j = 0$ if $\vec{\boldsymbol{d}}_j = \max_i(\vec{\boldsymbol{d}}_i)$.
4. Compute the global direction: $\vec{\boldsymbol{D}} = \frac{1}{u-1}\sum_{i=1}^{u} w_i \cdot \vec{\boldsymbol{d}}_i$.
5. Compute displacement direction around (x_i, y_i):
$\vec{\boldsymbol{d}}_k = (x_k - x_i, y_k - y_i)/\|(x_k - x_i, y_k - y_i)\|_2$ if $\|(x_k, y_k) - (x_i, y_i)\|_2 \le 5$.
6. For the ith sub-tracker, compute all the candidates's confidence scores via biasing toward the global direction: $\vartheta_k^t = \mathbf{O}_{i,\cdot}(x_k, y_k) \odot |\mathbf{S}_{i,\cdot}(x_k, y_k)| - \varepsilon \cdot \|\vec{\boldsymbol{D}} + \vec{\boldsymbol{d}}_k\|_2$.

End For

In fact, the optimal solution of Eq. (12) can be obtained via the coordinate descent or dynamic programming method, yet the computational cost of both choices is expensive. However, since the variations among sub-backgrounds tend to stay uniform, we can seek the sub-optimal solution by being bias toward the second term ($\|\sum_{i=1}^{u} \vec{\boldsymbol{d}}_i\|_2$). That is, we constrain the searching directions of each sub-tracker to be identical as much as possible, and then the center location of the global background can be determined by seeking the minimum feature variation with respect to the previous low-rank information (i.e., the first term in Eq. (12)). The performance improvement directly benefitting from this strategy is demonstrated in Fig. 3. Since the traditional object tracker is easily affected by the moving object, the tracked background areas tend to bias toward the moving object when its surrounding areas are similar. Obviously, our background tracking method exhibits much robust tracking result, and more quantitative evaluation to verify the effectiveness of our background tracking can be found in the Experimental Result section. The entire computation procedure of Eq. (12) is summarized in Algorithm 1.

After the current target background being located, we need to update each sub-tracker's appearance model to accommodate background variations. However, because the target background tends to occupy the majority of the given frame, we need to re-initialize the corresponding sub-tracker by moving the current tracking window toward the opposite direction when it reaches frame boundary. Therefore, due to the frequent initializing process, the updating process of appearance model **A** only needs to replace the oldest observation with the current tracking observation (as we mentioned in the previous sections).

6. Exploration of semantic coherency based on the integrated computing strategy of divide-and-conquer and low-rank analysis

Till now, we can obtain batches of tracked frames, which are relatively consistent with slight variations induced by tracking, view angle change, or camera jitter, etc. Since the salient motions usually exhibit high sparse residual, we can further conduct low-rank analysis over these frame batches respectively, which will decompose each frame batch into low-rank part and sparse part. Nevertheless, there still exist two challenges which hinder the straight-forward utility of low-rank analysis for salient motion detection: (1) the tracking drift induced displacements easily lead to false-alarm detection; and (2) the slow movements induced feature overlapping easily results in hollow effect, which tends to leave the inner region of the detected moving object being empty, see details in Fig. 4 (more demonstrations can be found in Section 3 in our Supplement Material). Therefore, we propose to incorporate alignment into the low-rank revealing process (Section 6.1), and integrate the previous-batch low-rank prior to guarantee the robustness of salient motion detection (Section 6.2).

6.1. Low-rank background recovery based on aligned RPCA

Given a frame batch containing m-frame similar backgrounds $\mathbf{G} = [G_1, G_2, \ldots, G_m]$, the low-rank decomposition [36] is defined as

$$(\mathbf{L}^*, \mathbf{S}^*) = \arg\min_{\mathbf{L},\mathbf{S}}(|\,|\mathbf{L}\|_* + \lambda\|\mathbf{S}\|_1), \quad s.t. \quad \mathbf{G} = \mathbf{L} + \mathbf{S}, \tag{13}$$

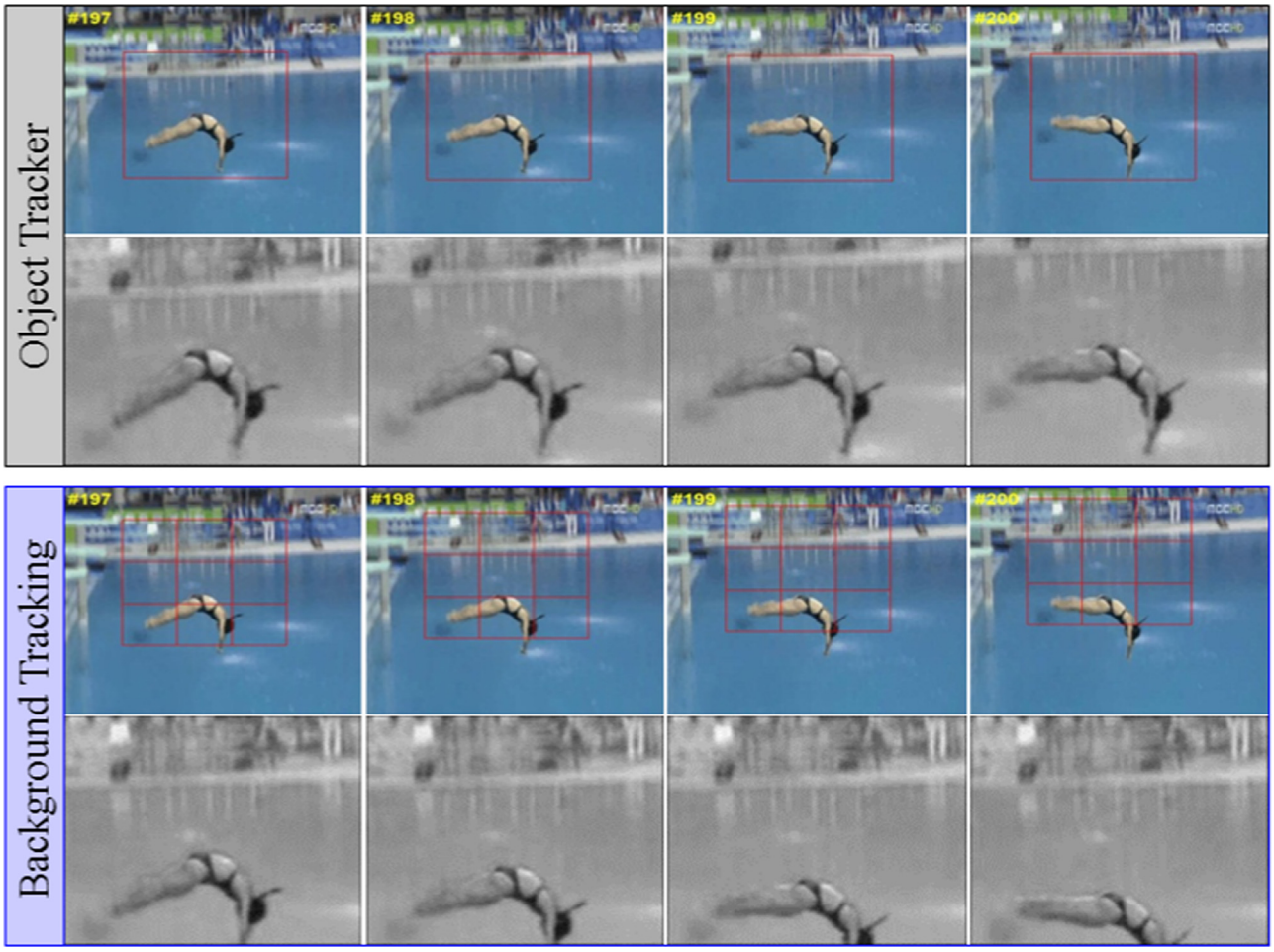

Fig. 3. Demonstration of the performance improvements via our background tracking strategy. The top part shows the results of the traditional object tracker [1], and the bottom part shows the results of our background tracking. For each demonstration, the top row is the tracking result, and the bottom row is the demonstration of tracked backgrounds.

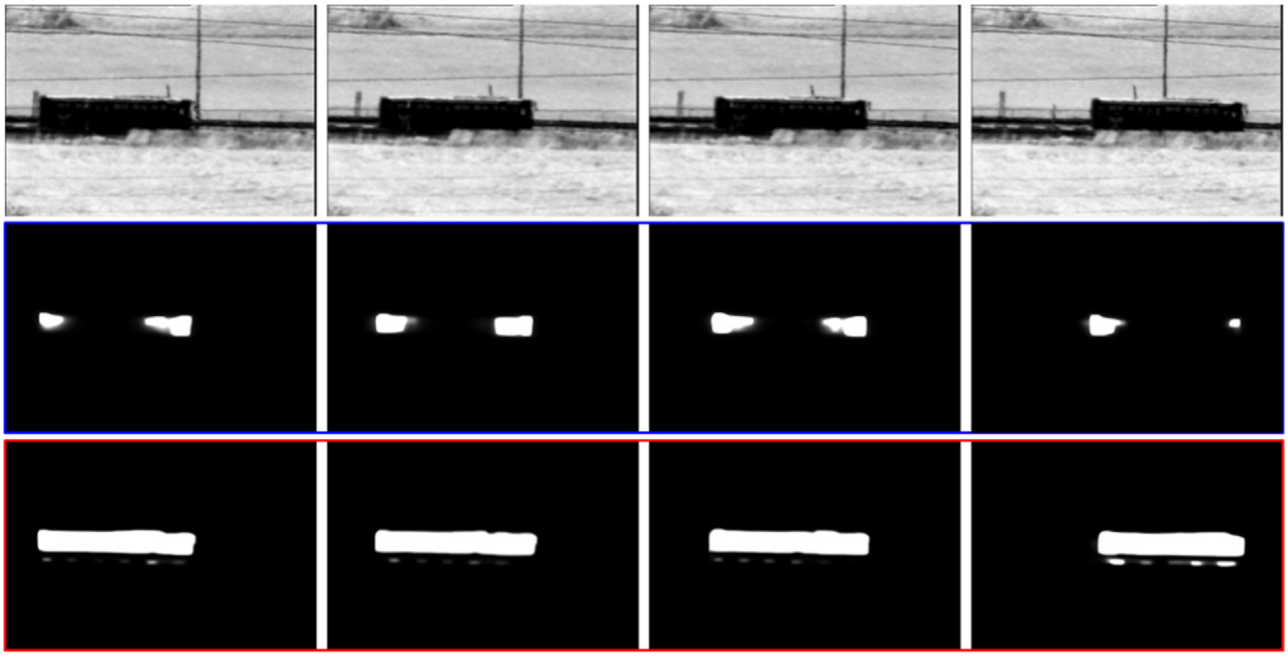

Fig. 4. The demonstration of the hollow effect on Turbulence3 sequence. The middle row marked by blue rectangle shows the results produced without the low-rank prior, and the bottom row marked by red rectangle shows the results produced with the low-rank prior. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

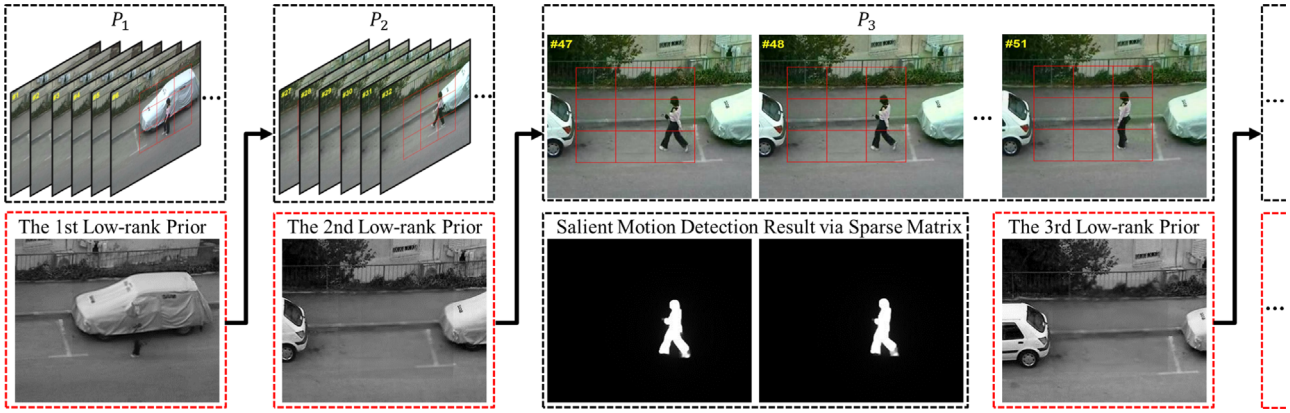

Fig. 5. Demonstration of the introduction of previous low-rank prior. Each tracked frame batches (with stable backgrounds) are denoted by P_i. Low-rank information are marked with dash-line rectangle, and the corresponding salient motion detection results are shown in the middle bottom.

where $\mathbf{L}$ and $\mathbf{S}$ denotes the low-rank part and sparse part respectively. Since the residuals in sparse matrix $\mathbf{S}$ indicate the saliency degree of the corresponding frame patch, the column-wise l_1-norm of $\mathbf{S}$ can be regarded as the motion saliency indicator. Suppose that these tracked frame patches $\mathbf{G}$ are aligned by seeking a set of 2D transform $\tau=[\tau_1,\tau_2,\ldots,\tau_m]$, the formulation of Eq. (13) becomes

$$(\mathbf{L}^*,\mathbf{S}^*)=\arg\min_{\mathbf{L},\mathbf{S},\tau}(\|\mathbf{L}\|_*+\lambda\|\mathbf{S}\|_1),\quad s.t.\quad \mathbf{G}\circ\tau=\mathbf{L}+\mathbf{S}. \tag{14}$$

In fact, because the constraint $\mathbf{G}\circ\tau=\mathbf{L}+\mathbf{S}$ is non-linear, it is difficult to solve Eq. (14) directly. Following the methods in [14,43], we rewrite Eq. (14) with the following linearized formulation:

$$(\mathbf{L}^*,\mathbf{S}^*)=\arg\min_{\mathbf{L},\mathbf{S},\Delta\tau}(\|\mathbf{L}\|_*+\lambda\|\mathbf{S}\|_1),\quad s.t.\quad \mathbf{G}\circ\tau+\sum_{i=1}^{m}\mathbf{J}_i\Delta\tau_i\epsilon_i^T=\mathbf{L}+\mathbf{S}, \tag{15}$$

where $\mathbf{J}_i$ is the Jacobian of the ith background patch, and the 2D transform τ_i and ϵ_i denote the standard basis in $\mathbb{R}^m$. Hence, the aligned low-rank recovering process consists of three main steps: (1) compute the Jacobian matrices $\mathbf{J}_i\leftarrow(\partial/\partial\zeta)(vec(G_i\circ\zeta)/\|vec(G_i\circ\zeta)\|_2)|_{\zeta=\tau_i}$, where $i\in[1,2,\ldots,m]$; (2) warp and normalize the input background patches $\mathbf{G}\circ\tau$; and (3) solve the linearized convex optimization Eq. (15) via the accelerated proximal gradient (AGP) method. Therefore, the drifting displacement can be well handled, and then we should further make use of the previous low-rank prior to guide the RPCA based low-rank recovery in Eq. (15).

6.2. Low-rank prior biased RPCA for salient motion detection

The traditional solution of RPCA based low-rank (Eq. (13)) revealing can be solved via the proximal gradient approach (Eq. (16)) [36,44], which is a relaxed version of Eq. (13):

$$(\mathbf{L}^*,\mathbf{S}^*)=\arg\min_{\mathbf{L},\mathbf{S}}\left(\mu\|\mathbf{L}\|_*+\lambda\mu\|\mathbf{S}\|_1+\frac{1}{2}\|\mathbf{G}-\mathbf{L}-\mathbf{S}\|_F^2\right), \tag{16}$$

where $\|\cdot\|_F$ denotes the Frobenius norm, μ is a small constant, and the solutions of Eq. (16) are approximately equal to those of Eq. (13) if $\mu\searrow 0$. In fact, we can simply perform quadratic approximation on $\mathbf{L}$ and $\mathbf{S}$ separately as

$$\mathbf{L}_{k+1}=\arg\min_{\mathbf{L}}\mu\|\mathbf{L}\|_*+\|\mathbf{L}-\left(\tilde{\mathbf{L}}_k-\frac{1}{4}\nabla_L\|\mathbf{G}-\mathbf{L}-\mathbf{S}\|_F^2|_{\tilde{L}_k,\tilde{S}_k}\right)\|_F^2, \tag{17}$$

$$\mathbf{S}_{k+1}=\arg\min_{\mathbf{S}}\lambda\mu\|\mathbf{S}\|_1+\|\mathbf{S}-\left(\tilde{\mathbf{S}}_k-\frac{1}{4}\nabla_S\|\mathbf{G}-\mathbf{L}-\mathbf{S}\|_F^2|_{\tilde{L}_k,\tilde{S}_k}\right)\|_F^2, \tag{18}$$

And these two sub-problems can be efficiently solved via imposing soft thresholding on $\mathbf{S}$ and singular value thresholding [45] on $\mathbf{L}$.

Since we have decomposed the original video into many small sub-groups sharing relatively-stable backgrounds via background tracking, the low-rank information obtained via RPCA should well represent the backgrounds. However, due to the limited-number observations in each frame batch, some small parts, which closely connect the salient moving object, may stay salient throughout the entire observation (see Fig. 6(a)). Such "feature overlapping" phenomenon ultimately results in incorrect RPCA low-rank structures and cause false-alarm detections (see Fig. 6(b)). Therefore, we propose to introduce previous low-rank background information into current-batch low-rank revealing process (Fig. 5), and convert the thresholding (soft thresholding on $\mathbf{S}_{t+1}=\text{sign}(\mathbf{S}_t)(|\mathbf{S}_t|-\lambda\mu)$ and singular value thresholding on $\mathbf{L}$) based iterative solution into the following form:

$$\mathbf{L}_{t+1}=\Phi\left(\mathbf{U}\left[\Sigma-\mu\begin{bmatrix}0.5 & 0\\ 0 & \mathbf{I}\end{bmatrix}\right]_+\mathbf{V}\right),\quad (\mathbf{U},\Sigma,\mathbf{V})=svd(\mathbf{Y}_t\ \mathbf{H}). \tag{19}$$

Here $\mathbf{L}_{t+1}\in\mathbb{R}^{n\times k}$ denotes the low-rank information obtained from the $(t+1)$th RPCA low-rank revealing process, $\mathbf{I}$ denotes the identity matrix, n is the feature dimension, k is the total frame number of current frame batch, function $\Phi(\cdot)$ selects the left k columns from the input matrix. By putting the previous low-rank information into the last column of $\mathbf{G}$, we have $\mathbf{Y}_t=\tilde{L}_t-\frac{1}{2}(\tilde{L}_t+\tilde{S}_t-\mathbf{G})$, $\tilde{L}_t=\mathbf{L}_t+\delta(\mathbf{L}_t-\mathbf{L}_{t-1}),\tilde{S}_t=\mathbf{S}_t+\delta(\mathbf{S}_t-\mathbf{S}_{t-1})$, δ is a parameter controlling the iteration step size, and $\mathbf{H}\in\mathbb{R}^{n\times k/2}$ denotes the low-rank prior matrix.

Therefore, the previous low-rank information will dominate $\mathbf{Y}_t$, and the singular value thresholding can facilitate the low-rank revealing process to bias toward these low-rank prior (by setting the thresholding of the largest singular value (0.5) to be smaller than others, and refer to the details in Eq. (19)). The entire computation procedure is summarized in Algorithm 2, and Fig. 6(c) show the performance improvement thanks to the introduction of low-rank prior.

Algorithm 2. The coupling of short-term aligned RPCA and low-rank prior.

Input: Tracked the t-th frame batch $\mathbf{G}^t$; Low-rank prior in $(t-1)$th frame batch $\mathbf{H}^{t-1}$.

Output: The revealed low-rank information $\mathbf{L}^t$ and sparse residual $\mathbf{S}^t$.

Initialization: $\mu=\|\mathbf{G}^t\|_2/1.25$; Compute the Jacobian matrix $\mathbf{J}^t$; $t_0,t_1=1$.

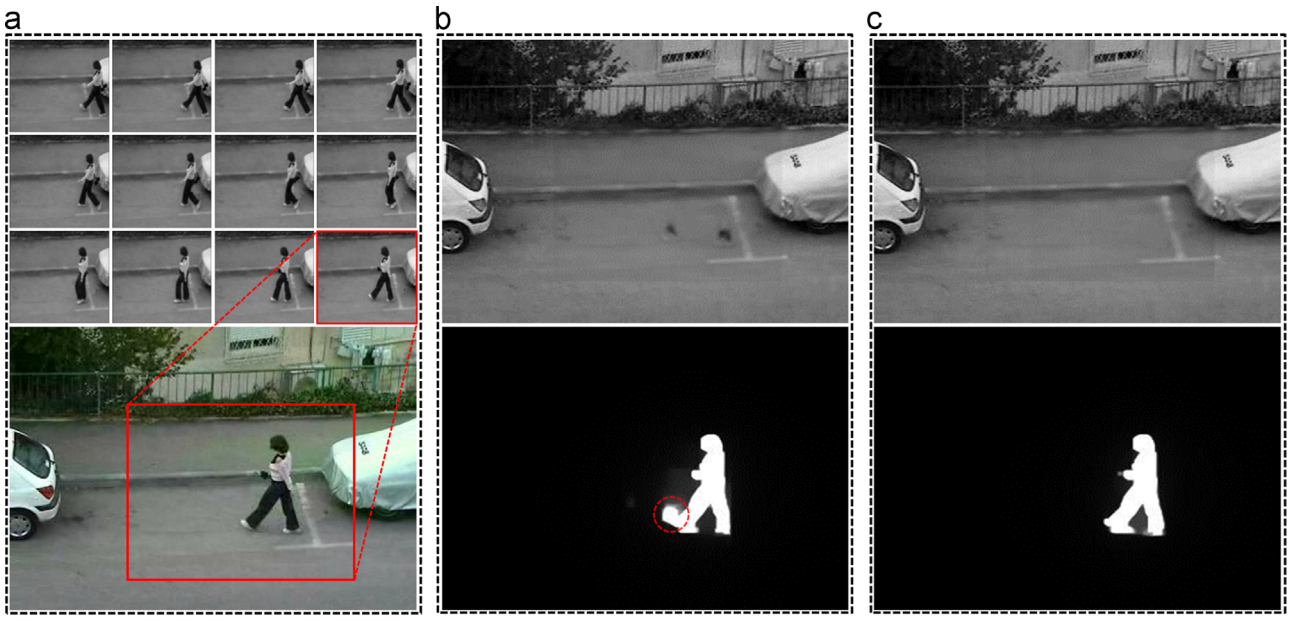

Fig. 6. Demonstration of the performance improvement benefitting from the introduction of the low-rank prior. (a) shows the tracked background patches. The top row in (b) is the low-rank information derived from (a) without previous low-rank prior, and the salient detection results are given in the bottom row (false-alarm detections are marked by red dash cycle). (c) demonstrates the low-rank decomposition results guided by the previous low-rank prior, wherein the top row is the low-rank information and the bottom row is the sparse information (salient motion detection result). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

For $j=1{:}10$

1. $\mathbf{Y}_j=\mathbf{G}^t+\mathbf{D}_j-\mathbf{S}_j^t+\mu\cdot\mathbf{R}_j$.
2. Introduce low-rank prior via $\mathbf{Y}_j=[\mathbf{Y}_j\ \mathbf{H}]$.
3. Compute current low-rank $\mathbf{L}_j^t$ via Eq (19), and update $\mathbf{Y}_j$ with $\delta=\frac{t_{j-1}-1}{t_j}$.
4. Compute the sparse matrix via $\mathbf{S}_j^t=\mathrm{sign}(\mathbf{Y}_j)(|\mathbf{Y}_j|-\lambda\mu)$.
5. Update $\mathbf{Y}_j=\mathbf{G}^t-\mathbf{L}_j^t-\mathbf{S}_j^t+\mu\cdot\mathbf{R}_j$.
6. Approximate gradient of aligned $\mathbf{Y}_j$ via $\mathbf{D}_j=\mathbf{J}^t(\mathbf{J}^t)^T\mathbf{Y}_j$.
7. Update the gradient of current residual via $\mathbf{R}_j=\mathbf{R}_j+\mu(\mathbf{G}^t+\mathbf{D}_j-\mathbf{S}_j^t-\mathbf{L}_j^t)$, and set $\mu=\mu/1.25$, $t_{j+1}=0.5(1+\sqrt{1+4t_j^2})$.

End For

7. Saliency clues for dynamic background modeling

In principle, the major challenges in robust salient motion detection come from two aspects: (1) ghost effects caused by intermittent object moving (IOM); and (2) false-alarm detections induced by dynamic backgrounds (DB). As for the IOM problem, we can leverage the previously detected salient motion to guide the update of the current low-rank prior, and see the details in Section 7.1. As for the DB problem, the basic principle of our solution is "short-term thresholding", which separates the stable region from the dynamic backgrounds by performing statistics on the variation of sparse residuals, and see the details in Section 7.2. In fact, to further avoid the false-alarm detections (especially for non-stationary videos), we also adopt the voting strategy to improve the accuracy and robustness, and see the details in Section 7.3.

7.1. Saliency clues for low-rank background prior updating

Given a long-term video sequence, the background tends to be time-varying, background updating is indispensable for most modeling based salient motion detection methods. However, the IOM problem can be practically solved with the help of the following two strategies: (1) we should suspend the background updating for those regions where the detected moving object comes to standstill by making the static object keep a high saliency value; and (2) as for the newly-exposed background areas that are previously covered by the "current static object", their updating should be further boosted. In fact, because traditional methods usually model the foreground and the background separately in a pixel-wise manner, the IOM problem can be well handled. Yet, because massive computation is needed for each observed frame batch to frequently conduct initialization, the pixel-wise modeling strategy becomes invalid in our method. Therefore, we resort to employing the saliency-degree analysis over the previously detected salient motion to guide the updating of current low-rank information. Here the key rationality is that the newly-exposed backgrounds should exhibit strong similarity with respect to its non-salient surroundings in RGB feature space, while the currently-stopped object should maintain high contrasts. Thus, the updating strength of the low-rank information, which is obtained from the current aligned short-term RPCA analysis, should fully respect this saliency metric. The "saliency degree" is defined as

$$Sal_{(i,j)}=\frac{1}{n}\sum_{D\in\eta}\omega\cdot\|(\mathcal{R}_{(i,j)},\mathcal{G}_{(i,j)},\mathcal{B}_{(i,j)})-(\mathcal{R}_{(p,q)},\mathcal{G}_{(p,q)},\mathcal{B}_{(p,q)})\|_2, \tag{20}$$

where $\mathcal{RGB}$ indicates the RGB color value of the original frame, $D=\|(i,j)-(p,q)\|_2$ and $\eta=100$ control the local scope of the contrast computation of our saliency clue, $\omega\in\{0,1\}$ is an indicator used for saliency computation, which guarantees to exclude ($\omega=0$, *if* $\mathbf{S}_{(p,q)}<0.1\times\overline{S}$) the pixels with high residuals in the sparse matrix $\mathbf{S}$. And $\overline{S}$ denotes the mean of $|\mathbf{S}|$. Obviously, the saliency value of the newly-exposed background should be consistently lower than the moving object, and we leverage this information to obtain the saliency clue mask to guide the updating

of the low-rank prior via Eq. (21).

$$SM_{(i,j)} = \frac{1}{Z}\psi \odot \exp(\xi \cdot \mathbf{S}^t_{(i,j)} \cdot Sal_{(i,j)}), \tag{21}$$

where Z denotes the normalizing factor, ξ is a scaling parameter, and we empirically set it to 10, $\mathbf{S}^t$ denotes the sparse residual of the last video frame in the tth frame batch, and $\psi \in \{0,1\}$ guarantees SM only concentrates on the pixels with high $\mathbf{S}_{(i,j)}$ ($\psi = 1$, *if* $S_{(i,j)} > 2 \times \overline{S}$). Furthermore, instead of computing the saliency clues in a pixel-wise manner, we employ super-pixels [46] to alleviate the computation burden, and Fig. 7(e) demonstrates the computation of the saliency clues. Although the newly-exposed "white arrow" is regarded as salient motion (ghost effect) in the 160th frame (see the yellow dashed rectangle in Fig. 7(d)), the corresponding region in the saliency clue mask exhibits low saliency. As shown in Fig. 7(e) and (f)), it guarantees correct low-rank prior updating, wherein the newly-exposed arrow is correctly classified as low-rank information.

After obtaining the Saliency Clue Mask (SM), the low-rank prior updating procedure can be formulated as

$$\mathbf{L}^t = (1 - SM) \cdot \mathbf{L}^t + SM \cdot \mathbf{L}^{t-1} + \beta \cdot SM \cdot (\mathbf{L}^t - \mathbf{L}^{t-1}), \tag{22}$$

where $\mathbf{L}^t$ denotes the low-rank information obtained in the tth frame batch, and $\beta = 0.1$ controls the compensation strength along the gradient direction. The salient motion detection improvements are demonstrated in Fig. 8. Compared to the method that directly performs low-rank revealing strategy over frame batch (Fig. 8(b)), the introduced low-rank information can greatly alleviate the hollow effects caused by feature overlapping. However, incorrect low-rank revealing still exists when the moving object undergoes extremely slow movements, and the ghost effects can be easily found in the last three rows of Fig. 8(c). Directly benefitting from our saliency clue strategy, Fig. 8(d) shows that the incorrect low-rank revealing has been properly handled.

7.2. Saliency clues for dynamic background

Another critical issue in salient motion detection is how to detect the correct moving object while eliminating the influences from complex dynamic backgrounds. Because the dynamic backgrounds and the true motions tend to have variance of similar degree, if solely depending on the pixel-level feature distance measurement with respect to the moving object model, it is difficult to obtain satisfactory salient motion detection results. Moreover, for non-stationary videos, it is equally hard to leverage the pixel-wise thresholding strategy due to the absence of pixel's correspondence among consecutive video frames.

Therefore, following our batch-wise salient motion detection, we propose to define another saliency clue over the sparse matrix $\mathbf{S}$ to filter out the dynamic backgrounds globally. Our rationality originates from the following observations: (1) the sparsity degree of dynamic background regions varies more frequently around their mean value than the stable background region; (2) the sparsity degree of dynamic background regions tends to become relatively weak when some moving objects are passing through; and (3) the sparsity degree of moving objects can be either frequently changing (e.g., moving vehicles in a lineup fashion) or not (e.g., slow walking pedestrians), but both its amplitude and its duration are larger than those of dynamic background regions. Thus, according to the "third" observation above, we batch-wisely count the "switch" times around four pre-defined hard thresholds to measure the saliency degree, and "switches" are marked with blue cycle in Fig. 9(C). Given the ith pixel, the formulations of these four hard thresholds (UT_1, UT_2, DT_1, DT_2 are illustrated in Fig. 9(c)): $DT_1 = 0.9 \times \overline{S_i}$; $DT_2 = 0.8 \times \overline{S_i}$; $UT_1 = 1.1 \times \overline{S_i}$; $UT_2 = 1.2 \times \overline{S_i}$. Here $\overline{S_i}$ denotes the mean value of the ith row of $\mathbf{S}$ (see Fig. 9(b)). According to the "first" observation above, both UT_1 and DT_1 guarantee to only consider those "large" variations. According to the "second" observation above, both UT_2 and DT_2 guarantee to exclude the variations caused by moving objects. Thus, the "switch" time ST_1 (Eq. (23)) around UT_1 and the "switch" time ST_2

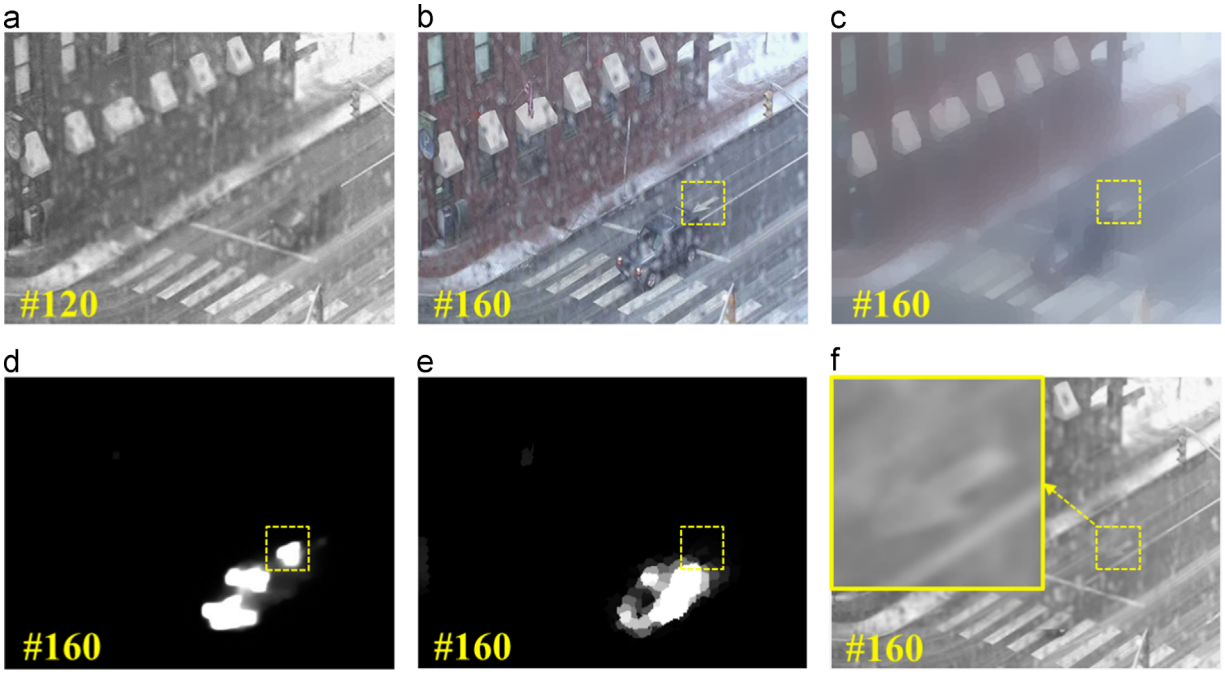

Fig. 7. Illustration of the low-rank prior updating. (a) demonstrates the low-rank information of the 120th video frame, (b) is the original 160th frame, (c) is (b)'s super-pixel representation, (d) shows the salient motion detection result of the 120th frame, (e) is the saliency clue mask obtained via Eq. (21), (f) is the newly updated low-rank information of the 160th frame via Eq. (22). (For interpretation of the references to color in the text, the reader is referred to the web version of this paper.)

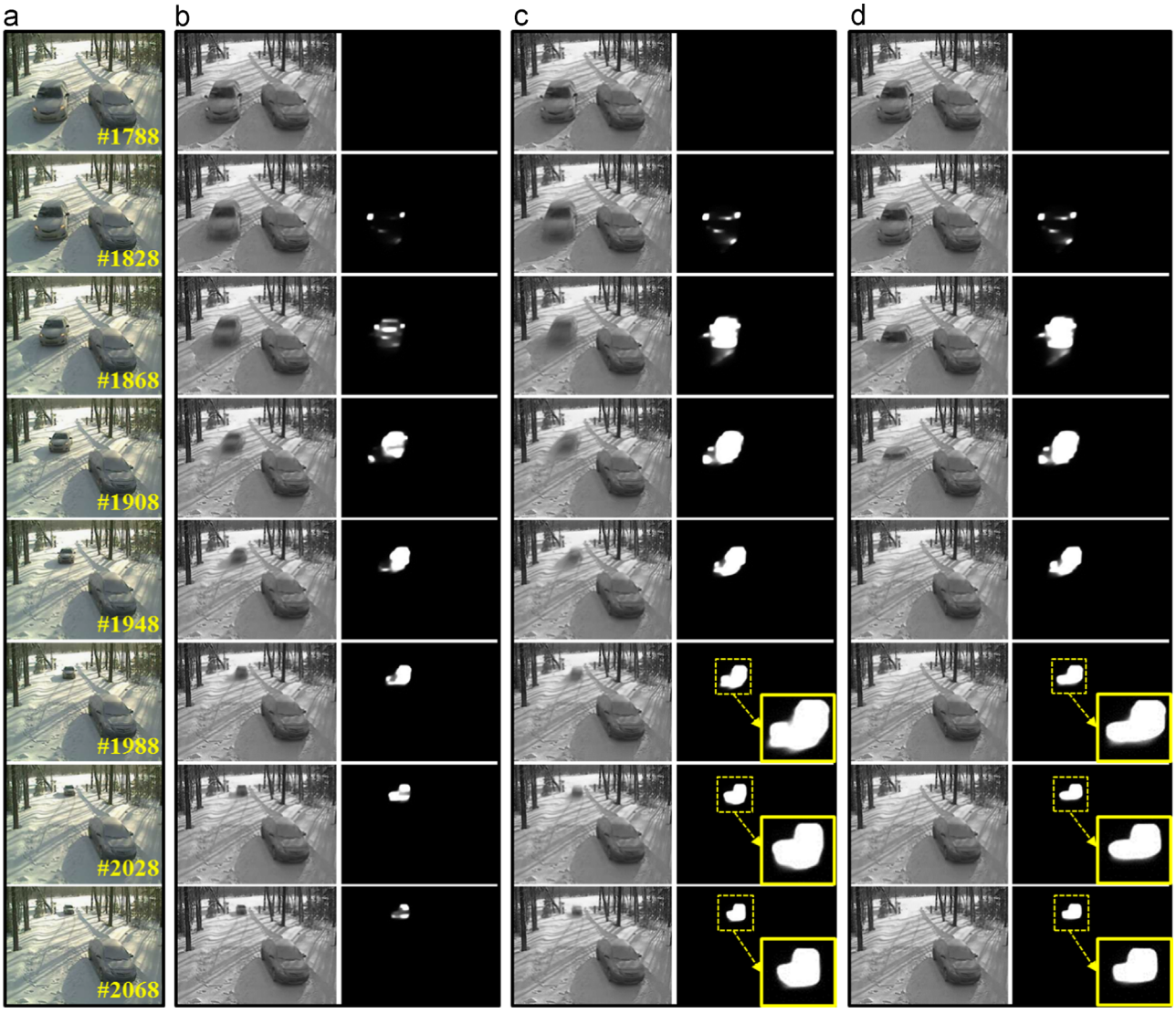

Fig. 8. Demonstration of the performance improvement thanks to the introduction of the *Low-rank* Prior and *Saliency Clue*. (a) is the original video sequence, and the corresponding frame indices are marked with yellow color in the right bottom, (b) demonstrates the low-rank information (left column) and the salient motion detection results (right column) obtained from our short-term RPCA, (c) demonstrates the results after incorporating the low-rank prior, (d) demonstrates the results when using saliency clue to guide the updating of the low-rank prior. Both hollow effects and ghost effects have been properly handled by our saliency clue strategy. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

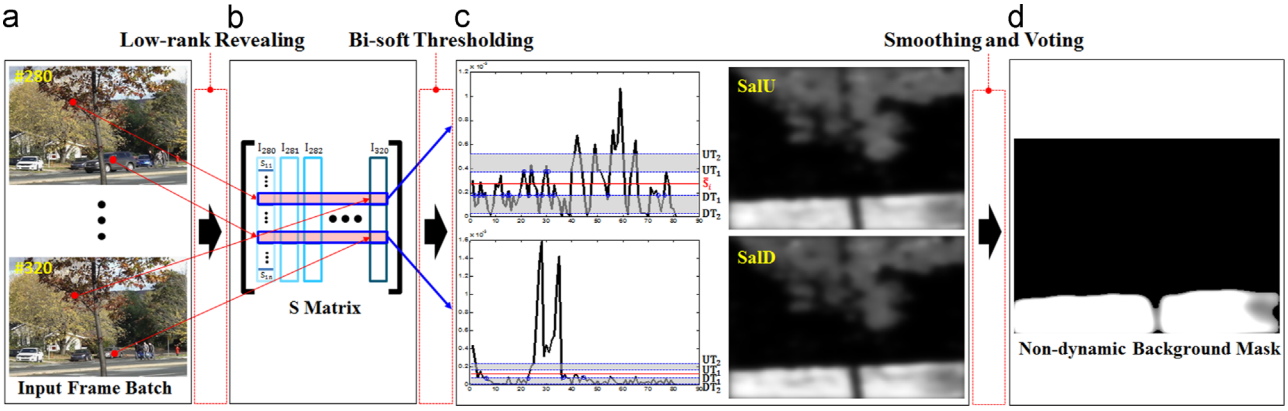

Fig. 9. The pipeline of the stable background mask generation. Red points in (a) separately indicate the pixels belonging to dynamic background regions and foreground motion areas. The left column of (c) demonstrates the column-wise sparse residual distribution of S, the mean value of S is marked with red line, and the gray regions indicate the thresholding range, which correctly separate the dynamic background regions from the true motions. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

(Eq. (24)) around DT_1 can be computed as follows:

$$ST_1 = \sum_{j=1}^{k-1} \|(|\mathbf{S}_{(i,j)}| - UT_1)\|_0 \cdot \|(|\mathbf{S}_{(i,j+1)}| - UT_1)\|_0 \cdot \|(|UT_2 - |\mathbf{S}_{(i,j)}|)\|_0, \quad (23)$$

$$ST_2 = \sum_{j=1}^{k-1} \|(DT_1 - |\mathbf{S}_{(i,j)}|)\|_0 \cdot \|(DT_1 - |\mathbf{S}_{(i,j+1)}|)\|_0 \cdot \|(\|\mathbf{S}_{(i,j)}|) - DT_2\|_0, \quad (24)$$

where $\|\cdot\|_0$ denotes l_0-norm, and k denotes the image number of current frame batch. Then, we define the saliency clues as $SalU = f(1/ST_1)$ and $SalD = f(1/ST_2)$, as shown in the right column of Fig. 9 (c), $f(\cdot)$ denotes a 9×9 Gaussian filter. Therefore, the formulation of our stable saliency mask (Fig. 9(d)) can be obtained according to the voting result of *SalU* and *SalD*, and the final salient motion

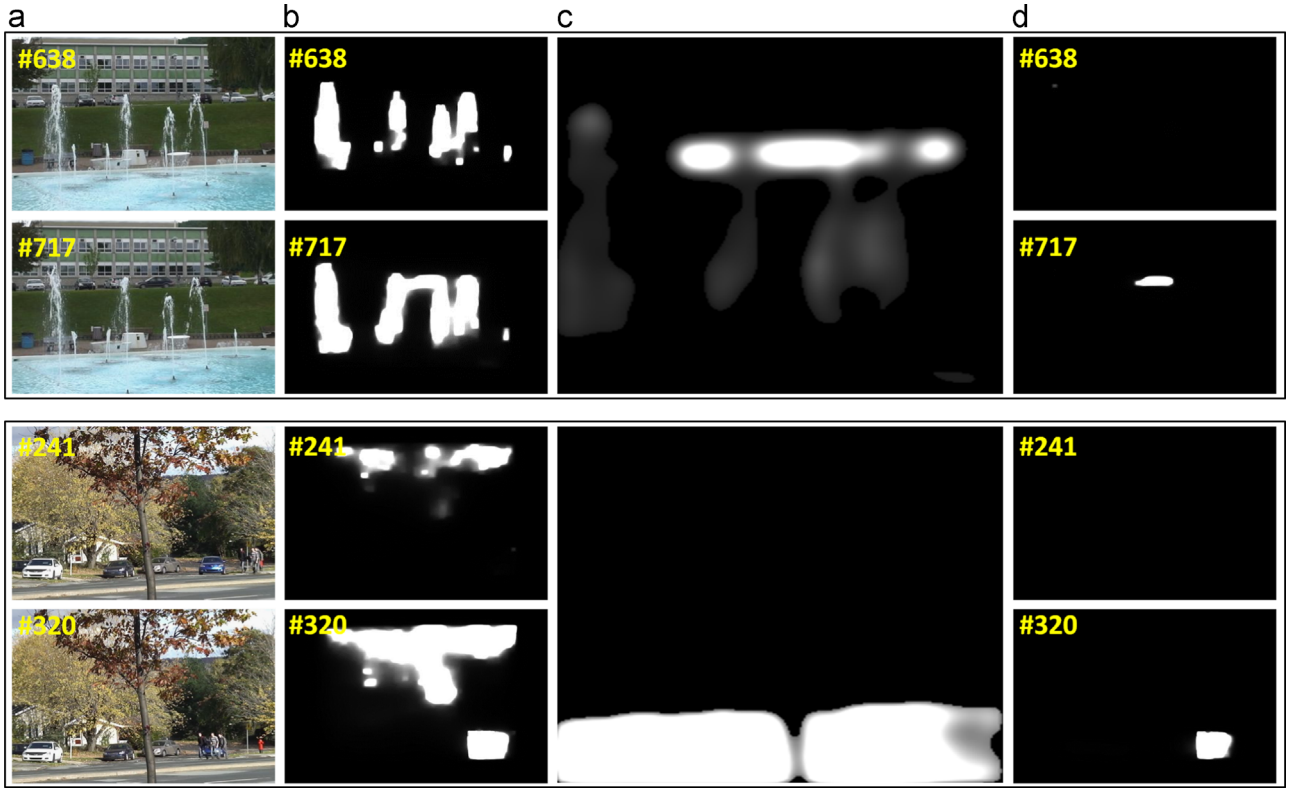

Fig. 10. Demonstration of the performance improvement thanks to the stable background mask. (a) shows the video frames separately from Fountain01 and Fall sequences, (b) demonstrates the salient motion detection results without considering the stable background mask, (c) demonstrates the saliency degree based stable background mask, (d) demonstrates the detection results by considering the stable background mask. Obviously, most of the false-alarm detections caused by the dynamic background are filtered out and this is a direct gain from our novel computational strategy.

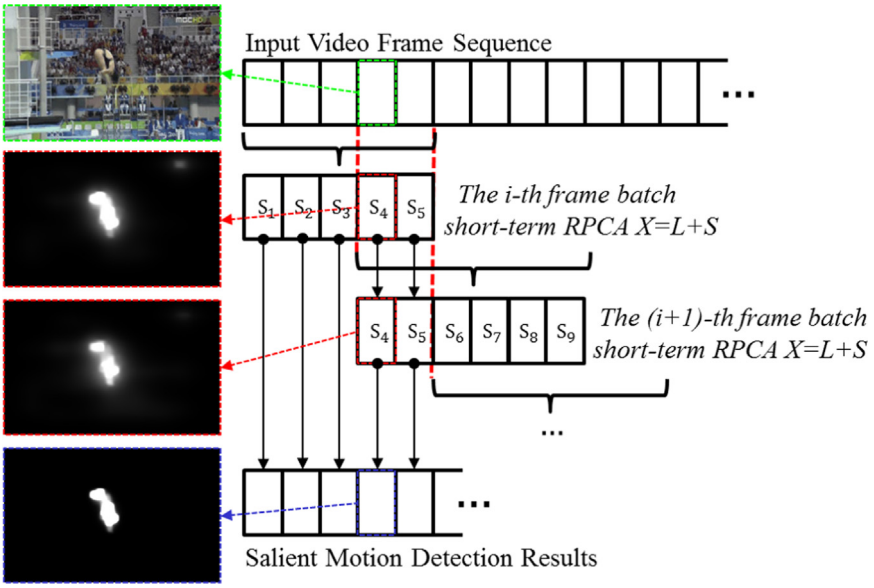

Fig. 11. Illustration of our multiple-observation coincidence computing strategy.

detection result can be computed by $\|S\|_1 \cdot f((SalU \times SalD)^2)$ (see Fig. 10(d)).

7.3. Multiple observation coincidence

As mentioned in Section 5, the frame number of each observing batch (k) is determined by the background tracking procedure. However, because the tracking procedure heavily relies on the choice of the initial video frame, given an identical frame, different batch assignments may lead to different detection results. Thus, the detection performance (accuracy rate) can be further boosted by introducing a separate voting strategy. That is, given the ith frame batch with frame index from t to $t+N$, we regard the $t+N/2$ (instead of the traditionally-chosen $t+N+1$ frame) frame as the initial tracking position for the $(i+1)-$th frame batch. Hence, we can obtain two salient motion detection results for each video frame, and the coincidence of these two independent detection definitely indicates the most trust-worthy salient motions. Therefore, we can obtain the final salient motion detection result via $f(\|\mathbf{S}_t^i\|_1) \odot f(\|\mathbf{S}_{t-N/2}^{i+1}\|)$, and see the details in Fig. 11. However, this computational strategy is optional because it will inevitably double the overall computation cost, and we suggest users to adopt this step when the accuracy is of utmost importance with the highest priority.

8. Experimental results and evaluations

8.1. Experiment settings

We compare our method with six state-of-the-art methods via comprehensive experiments over CD2014 benchmark [5] (mainly consisting of stationary video sequences except the PTZ category, see Fig. 12 and five non-stationary video sequences [47].

The involved six state-of-the-art methods include FTSG14 [7], SuBSENSE14 [6], CwisarDH14 [28], MOD13 [13], ViBe11 [32], and KNN06 [18]. Of which, FTSG14, SuBSENSE14, and CwisarDH14 are three top-performance salient motion detection methods suggested by CD2014 benchmark, and MOD13 adopts a low-rank analysis based solution similar to our method, while ViBe11 and KNN06 are two methods with the highest reference rate. All the quantitative comparison metrics in this paper are based on three widely-used criteria, which are Precision ($TP/(TP+FP)$), Recall

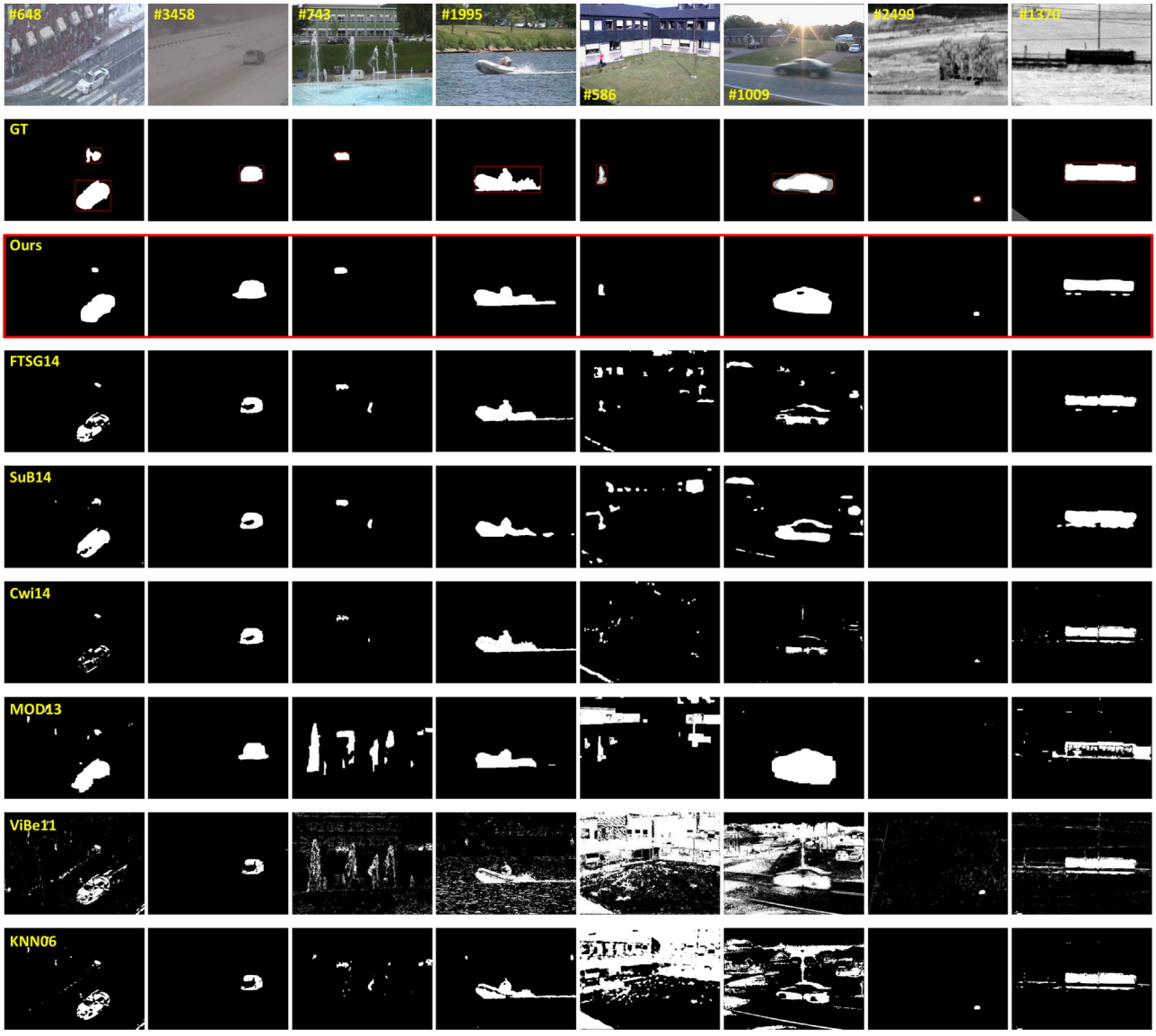

Fig. 12. Performance comparison on CD2014 benchmark. The second row demonstrates the Ground Truth (GT), rows 3–9 depict the results of our method, FTSG14 [7], SuBSENSE14 [6], CwisarDH14 [28], MOD13 [13], ViBe11 [32], and KNN06 [18], respectively.

($TP/(TP+FN)$), and F-measure ($2*Precision*Recall/(Preciosion+Recall)$). And *TP* denotes the true positive, *FP* denotes the false positive, and *FN* denotes the false negative.

We test the methods over seven categories of CD2014 benchmark dataset, including Baseline, Dynamic Background, Intermittent Object Moving, Bad Weather, Turbulence, Camera Jitter, and PTZ. The Baseline category has four stationary videos consisting of simple motions. The Dynamic Background category has six sequences, wherein the involved motions are frequently interrupted by dynamic backgrounds (e.g., swaying leafs in tree or gushing fountains). Similarly, the Bad Weather and Turbulence category separately contains four videos with interrupted motions induced by time-varying weather. The Intermittent Object Moving category contains six intermittent-motion videos, wherein the challenge is how to keep high saliency value for those stop-and-go objects while avoiding ghost effects when objects start moving again. The Camera Jitter category has four video sequences, wherein the motions are frequently interrupted due to camera jitter. The PTZ category contains four most challenging video sequences obtained from pan-tilt-zoom (PTZ) camera. Because of the non-stationary feature of PTZ, to the best of our knowledge, none of the state-of-the-art methods can very well handle such category. Particularly, the other give additional non-stationary video sequences are all from object tracking benchmark, including Woman, Diving, Ski, David3, and Bike sequences, wherein both the moving objects and the salient-motion ground truth are marked with rectangles.

Table 2
The detailed standard variance ($\times 10^{-4}$) of different tracking strategies, including the traditional object tracker [1] (1×1), our background tracking 2×2, 3×3, 4×4 and 5×5 over three datasets: Camera Jitter, PZT, and Additional Five categories. The best performances are marked with bold font.

Sequences	1×1	2×2	3×3	4×4	5×5
Badminton	0.472	0.424	0.404	**0.400**	0.411
Boulevard	1.359	0.822	**0.749**	0.757	0.752
Sidewalk	0.291	**0.243**	0.249	0.246	0.250
Traffic	0.472	0.467	0.357	**0.343**	0.353
ContinuousPan	0.722	0.689	**0.678**	0.690	0.694
IntermittentPan	0.081	**0.073**	0.074	0.080	0.078
TwoPositionPZTCam	0.135	**0.109**	0.117	0.113	0.113
ZoomInZoomOut	1.088	1.088	**1.087**	1.088	**1.087**
Bike	0.237	**0.227**	0.233	0.236	0.231
Diving	0.135	0.156	**0.132**	0.159	0.149
Ski	0.256	**0.248**	0.253	0.251	0.247
Woman	0.366	0.357	**0.364**	**0.364**	0.368
David	0.252	0.251	**0.242**	0.250	0.245
Average	0.451	0.396	**0.380**	0.382	0.382

Table 3
The averaged F-measure comparisons between state-of-the-art methods and our individual system components. Bold fonts indicate the best performance, the italic fonts indicate the second-best ones, and bold italic fonts indicate the third-best ones. BL: baseline, DB: dynamic background, BW: bad weather, CJ: camera jitter, Tur: turbulence, IM: intermittent object motion, Five: five additional categories, and '–' indicates the result is not available.

Methods	BL	DB	BW	CJ	Tur	IM	PTZ	Five
SuBSENSE14	**0.950**	0.817	*0.861*	**0.815**	*0.779*	*0.601*	0.347	0.050
FTSG14	*0.933*	*0.879*	0.822	0.751	0.712	**0.789**	0.324	–
MOD13	0.921	0.708	0.624	0.777	0.383	***0.594***	0.558	0.520
CwisarDH14	0.914	0.827	0.683	0.788	0.722	0.575	0.321	–
ViBe11	0.870	0.719	0.391	0.753	0.159	0.509	0.124	0.019
KNN06	0.841	0.685	0.758	0.689	0.519	0.502	0.212	0.055
Baseline System	0.572	0.626	0.747	0.461	0.720	0.400	0.534	0.203
+Background Tracking	0.572	0.650	0.747	0.471	0.720	0.400	0.703	0.574
+Low-rank Prior	0.464	0.703	0.771	0.553	0.739	0.395	*0.739*	0.585
+Saliency Clue 1	0.754	0.702	0.796	0.717	0.762	0.532	0.726	0.678
+Saliency Clue 2	0.752	0.863	0.851	0.716	0.762	0.537	0.726	*0.679*
+Saliency Refinement	0.814	**0.913**	**0.873**	0.740	**0.780**	0.554	**0.745**	**0.668**

Table 4
The comparisons of average precision and recall over Baseline sequences.

Method	Precision	Recall	F-measure
SuBSENSE14	**0.9495**	**0.9520**	**0.9503**
FTSG14	0.9170	0.9513	0.9330
MOD13	0.9126	0.9306	0.9215
CwisarDH14	0.9337	0.8972	0.9145
ViBe11	0.9288	0.8204	0.8700
Ours	0.8568	0.8379	0.8442
KNN06	0.9245	0.7934	0.8411

Table 5
The comparisons of average precision and recall over Dynamic Background sequences.

Method	Precision	Recall	F-measure
Ours	**0.9216**	**0.9072**	**0.9132**
FTSG14	0.9129	0.8691	0.8792
CwisarDH14	0.8499	0.8144	0.8274
SuBSENSE14	0.8915	0.7768	0.8177
ViBe11	0.7291	0.7616	0.7197
MOD13	0.7538	0.6682	0.7084
KNN	0.6931	0.8047	0.6854

Table 6
The comparisons of average precision and recall over Intermittent Object Moving sequences.

Method	Precision	Recall	F-measure
FTSG14	**0.8512**	**0.7813**	**0.7891**
SuBSENSE14	0.8149	0.5626	0.6012
MOD13	0.5032	0.7262	0.5945
CwisarDH14	0.7417	0.5549	0.5753
Ours	0.7213	0.5371	0.5540
ViBe11	0.7513	0.4729	0.5093
KNN06	0.7121	0.4617	0.5026

Table 7
The comparisons of average precision and recall over Bad Weather sequences.

Method	Precision	Recall	F-measure
Ours	0.8482	**0.9055**	**0.8730**
SuBSENSE14	**0.9091**	0.8213	0.8619
FTSG14	0.9231	0.7457	0.8228
KNN06	0.9114	0.6537	0.7587
CwisarDH14	0.8762	0.6228	0.6837
MOD13	0.5016	0.9159	0.6248
ViBe11	0.3086	0.7249	0.3918

Table 8
The comparisons of average precision and recall over Camera Jitter sequences.

Method	Precision	Recall	F-measure
SuBSENSE14	0.8115	**0.8243**	**0.8152**
CwisarDH14	**0.8516**	0.7437	0.7886
MOD13	0.7832	0.7721	0.7776
ViBe11	0.8064	0.7293	0.7538
FTSG14	0.7645	0.7717	0.7513
Ours	0.7985	0.7127	0.7403
KNN06	0.7018	0.7351	0.6894

8.2. Parameter selection

We quantitatively evaluate the performance improvements by comparing our *Background Tracking* (with different sub-tracker numbers, Section 5) with traditional object tracker [1]. Since the aim of our *Background Tracking* is to obtain frame batches with relatively-stable backgrounds, the level of standard variance of each frame batches can be regarded as the true tracking performance indicator. Meanwhile, to obtain objective and qualitative results, areas related to salient motions (indicated by Ground Truth) are excluded from the computation of frame batch's standard variance. Detailed results are documented in Table 2.

Because the traditional object tracker [1] mainly concentrates on the candidate target with global minimal residual in sparse matrix **S**, the standard variance of the '1×1' background tracking solution (in the traditional object tracker) stays at the highest level. In fact, the level of the standard variance gradually decreases with the increasing of the sub-tracker number. However, because the tracking performance of the sub-tracker drops fast with limited tracking area (due to too many sub-trackers, see details in the '5×5' column in Table 2), we choose '3×3' as the optimal choice to make balance between efficiency and performance (the efficiency and performance results can be found in Fig. 15(a)).

Table 9
The comparisons of average precision and recall over Turbulence sequences.

Method	Precision	Recall	F-measure
Ours	**0.7855**	0.7943	**0.7806**
SuBSENSE14	0.7814	**0.8050**	0.7792
CwisarDH14	0.8942	0.6068	0.7227
FTSG14	0.9035	0.6109	0.7127
KNN06	0.5117	0.7682	0.5198
MOD13	0.4160	0.6260	0.3835
ViBe11	0.1363	0.6272	0.1594

Table 10
The comparisons of average precision and recall over PTZ sequences.

Method	Precision	Recall	F-measure
Ours	**0.8014**	0.7127	**0.7454**
MOD13	0.4777	**0.8329**	0.5585
SuBSENSE14	0.2840	0.8306	0.3476
CwisarDH14	0.4824	0.3363	0.3218
FTSG14	0.2861	0.6730	0.3241
KNN06	0.1979	0.6980	0.2126
ViBe11	0.0801	0.6728	0.1246

Table 11
The comparisons of average precision and recall over five non-stationary sequences.

Method	Precision	Recall	F-measure
Ours	**0.8346**	0.5703	**0.6680**
MOD13	0.5319	0.5696	0.5204
KNN06	0.0289	**0.7983**	0.0550
SuBSENSE14	0.0312	0.2280	0.0508
ViBe11	0.0100	0.6348	0.0196

We also evaluate the performance influence of different ε in (12). In fact, a large value of ε means to emphasize the displacement penalty term, which makes the background tracking result bias toward the averaged initial position at the first iteration. However, a small value of ε tends to degenerate our *Background Tracking* into the traditional object tracker. Following the quantitative evaluation results in Fig. 15(b), we set $\varepsilon = 0.4$ as the optimal choice in our *Background Tracking* component.

Meanwhile, the quantitative evaluations of the hard thresholds (UT_1, UT_2, DT_1, DT_2) in our *Saliency Clues for Dynamic Background* (Section 7.2) component are demonstrated in Fig. 15(c). Because the value ranges of the threshold pairs UT_1, UT_2 and DT_1, DT_2 are symmetric under the constraints $DT_1 > DT_2, UT_2 > UT_1$, Fig. 15 (c) only demonstrates the averaged F-measure with different choices of DT_1 and DT_2. Obviously, the optimal choice is to set $DT_1 = 0.9$, $DT_2 = 0.8$, and we set $UT_1 = 1.1$, $UT_2 = 1.2$ accordingly.

8.3. Experiment comparisons and evaluations

Experiments over videos with intermittent motions: As documented in Tables 4 and 6, our method achieves poor performance on Baseline and Intermittent Object Moving datasets. In fact, all these methods except MOD13 belong to modeling-based methods, and they keep high background updating rate at the beginning while setting low updating rate for remaining frames. Directly benefit from this strategy, the stopped moving objects can be easily detected, and we can notice plausible performance. However, we do not adopt such a strategy to achieve biased performance. Besides, because of the short-term nature of our method (batch-wise low-rank revealing), our low-rank information updating strategy can not guarantee to obtain 100% pure background model, wherein incorrect updating may occur when the currently-moving object turns into static all of the sudden. Meanwhile, the poor performance of our method on Camera Jitter (Table 8) category is mainly caused by the worse performance on Side Walk sequence (with the recall rate of 0.41), which contains static moving pedestrians (the frames with static pedestrians exceed 80%). MOD13 also conducts the pre-alignment steps before low-rank revealing process, and it has strong constraints that the newly-detected salient motions must adjoin previous detections. Thus, it outperforms our method on these two datasets, however, it inevitably gives rise to poor precision rate for the sequences with frequent interruptions (see MOD13 in Tables 5, 7 and 9).

Experiments over videos with frequent interruptions: As for Bad Weather (Table 7), Dynamic Background (Table 5), and Turbulence (Table 9) categories, all these three categories have frequently-interrupted motions caused by dynamic background (large duplicate variations within fixed regions) or weather induced variations (unfixed regions with lower variation degree than dynamic background). Our method achieves comparable performance over Dynamic Background category, benefitting from our saliency DB mask. Although traditional modeling-based methods also adopt thresholding strategies to suppress saliency assignments in dynamic regions, our method has much better local properties than those of other methods (i.e., our saliency DB masks are batch-wisely independent), especially for Fountain02 and Over Pass

sequences with moving objects passing through dynamic background regions. For Bad Weather and Turbulence categories, since the variations induced by external factors are much lower than those induced by salient motions, our low-rank revealing solution can automatically eliminate those small false-alarm variations. Specifically, it should be noted that pixel-wise modeling-based methods (FTSG14 and ViBe11) tend to exhibit poor performance due to the absence of spatial constraints, wherein both of the variations, induced by salient motion and external factors, tend to be similar locally.

Experiments over non-stationary videos: As shown in Fig. 14, all the modeling-based methods perform worse when being employed to handle non-stationary videos. For PTZ category, the ZoomInZoomOut video is the most challenging sequence due to continuous scale variations. Because our method organically integrate the alignment steps into the low-rank revealing process, the background variations induced by scale change can be easily classified as low-rank background information, thus, we can effectively avoid false-alarm detection. But MOD13 has poor performance in this case, because its PCP-based alignment only serves as a preprocessing step before low-rank revealing. As for the ContinuousPan sequence, the camera undergoes slow pan movements, which is the easiest situation for non-stationary videos. However, because of the continuous background variation caused

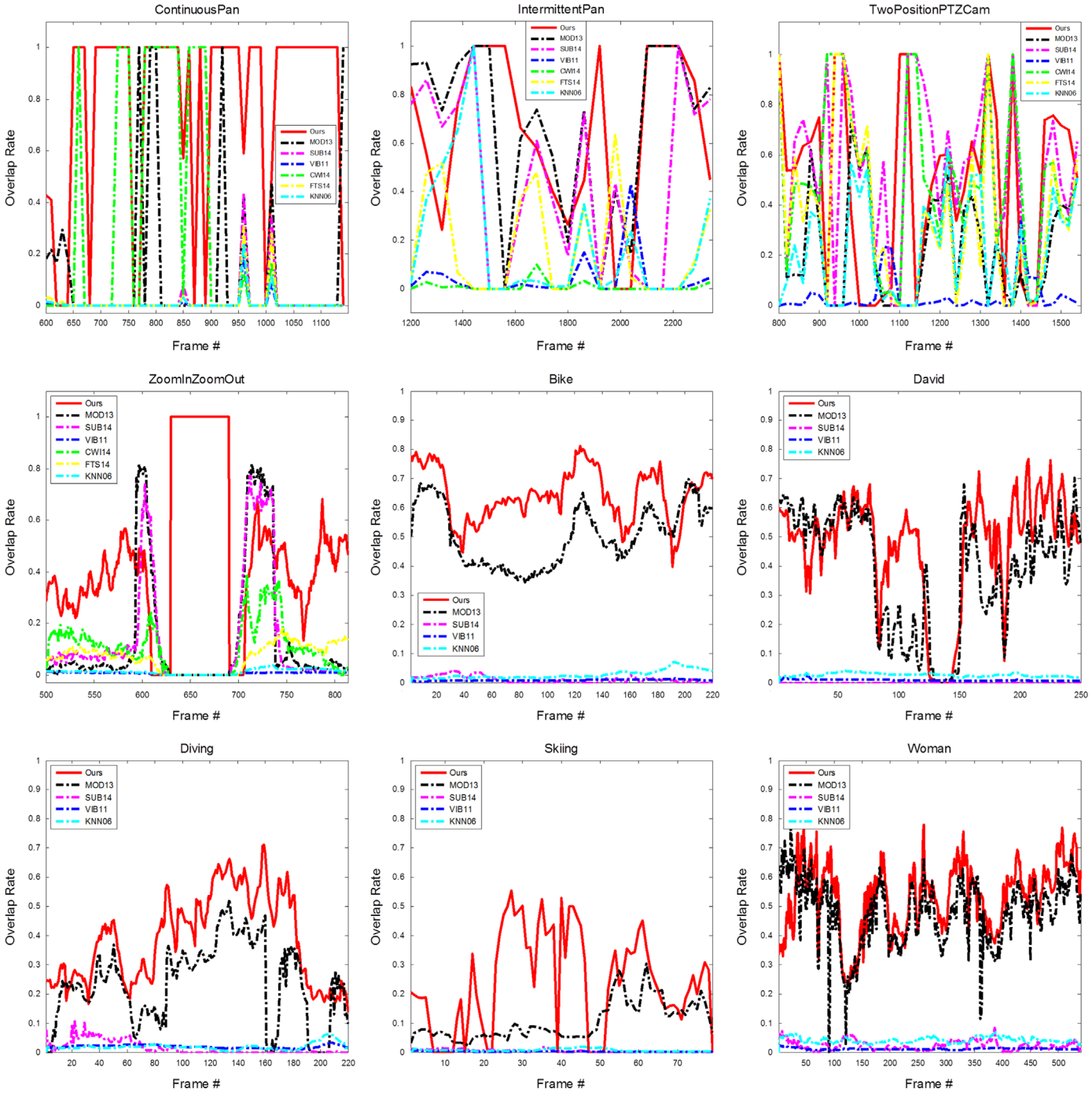

Fig. 13. Overlapping rate (OR) based comparison results on PTZ and five additional non-stationary video sequences. Because the source code of CwisarDH14 (CWI14) [28] and FTSG14 (FTS14) [7] are not available, we cannot evaluate the performance of these two methods on the five non-stationary video sequences of our own.

by camera movements, all the modeling-based methods in comparison exhibit worse performance. Meanwhile, for the additional five non-stationary sequences, comparing to other six state-of-the-art methods, our method has absolute advantages in recall rate and precision rate (see the details in Tables 10 and 11). Especially for Ski sequences, which contain fast moving objects and drastic camera movements within 79 frames, none of the compared methods can successfully overcome such challenges, because they must rely on training in order to establish a robust background model.

Meanwhile, to clearly demonstrate the superiority of our method for robust salient motion detection in non-stationary videos, as shown in Fig. 13, we also adopt the Overlapping Rate (OR) to conduct quantitative comparison over nine non-stationary sequences (Four of them are from PTZ category). And OR can be computed as

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

where ROI_T is the detected salient motion and ROI_G is the ground truth.

Besides, it should be noted that the ground truths for PTZ category are human-labeled foreground masks, while the ground truths for the other five non-stationary sequences are human-marked rectangles. Therefore, the quantitative results (Precision rate and Recall rate) over PTZ dataset are commonly better than those over the other five non-stationary sequences.

We further evaluate the performance of each component involved in our method by disabling individual components, including: (1) the *Baseline System*, which solely leverages the RPCA low-rank analysis strategy to detect the salient motion; (2) the *Background Tracking* component; (3). the *Low-rank Prior*, which is integrated into current low-rank revealing; (4) the *Saliency Clue 1*, which guides the low-rank prior updating; (5) the *Saliency Clue 2*, which restrains dynamic background; (6) the *Saliency Refinement* component. The overall experimental results (averaged F-measure over 8 video categories) are listed in Fig. 16, and the detailed results are documented in Table 3.

Obviously, the *Background Tracking* component can greatly improve the performance for the *PTZ* category and the *Five Additional* categories. This is because, video sequences in these categories are all non-stationary, and the traditional solution (the *Baseline System*) is incapable of handling these scenarios. Benefitting from the introduction of the *Background Tracking* component, the traditional long-term videos are converted into short-term frame batches, and the low-rank revealing solution finally becomes available, which greatly improves the salient motion detection performance.

However, for both stationary and non-stationary videos, because of the limited frame number (mainly in the non-stationary sequences) and potential feature overlapping caused by slow motion (which mainly exists in the stationary sequences), it is still hard to obtain satisfactory salient motion detection via performing low-rank analysis over short-term batches directly (easily resulting in hollow effect, and refer to Section 3 of our Supplementary Material). Therefore, we introduce the *Low-rank Prior* component to guide the low-rank revealing, which can well conquer the above limitation and simultaneously improve the overall performance for all 8 categories, and refer to Fig. 16 and Section 1 of our Supplementary Material for details.

It can be easily found from Fig. 16, the *Saliency Clue* 1 imposes strong positive effect on the baseline category, the camera jitter category, the intermittent motion category, and the turbulence category. In fact, the performance improvement over the camera jitter category mainly comes from the Sidewalk sequence for its long period of stop-and-go motions, while incorrect low-rank prior updating can easily result in "missing detection". Benefitting from the *Saliency Clue* 1 guided low-rank prior updating, the challenges over traditional stationary videos can be well conquered (including ghost effect caused by intermittent motions and hollow effect caused by slow movements), and Fig. 16 demonstrates the large F-measure improvement.

The *Saliency Clue* 2 is designed for the Dynamic Background category to avoid assigning large saliency value to those dynamic backgrounds. Because of the *Saliency Clue* 2 based dynamic background mask (see details in Section 7), this component contributes to enhancing the precision rate over the dynamic background category by a large margin.

Because the observations (salient motion detection results) from different frame batches are partly different, the voting procedure (see details in Section 7) can greatly increase the precision rate. Thus, the *Saliency Refinement* component can simultaneously improve the overall performance for all these 8 categories (see Section 1 of our Supplementary Material).

It can be easily observed from Fig. 16, our method outperforms the state-of-the-art methods by a large margin over non-stationary sequences (the PTZ category and the Five Additional categories) even without the *Saliency Clue* 1 component, the *Saliency Clue* 2 component and the *Saliency Refinement* component. As for the traditional stationary sequences (e.g., Bad Weather, Dynamic Background and Camera Jitter), the performance of our method is comparable to others even without the *Saliency Refinement* component. However, the traditional modeling based methods outperform our method, because the short-term frame batch strategy breaks the low-rank coherency.

Meanwhile, the corresponding time cost comparison results can be found in Table 12. Obviously, both the *Baseline System* (costing 0.182 s per frame) and the *Saliency Refinement* component (total consumption $\times$ 2) are the most time-consuming steps, yet the remaining components (*Saliency Clue* 1: 0.021 s per frame; *Low-rank Prior*: 0.001 s per frame) can improve the performance greatly with much less computational cost. It also should be noted that the *Background Tracking* component (costing 0.014 s per frame) can greatly improve the performance over non-stationary videos, and the *Saliency Clues* 2 component (costing 0.024 s per frame) can greatly improve the performance over the dynamic background category (DB) and the bad weather category (BW).

8.4. Limitations

Due to the short-term low-rank revealing strategy, our low-rank information updating method tends to gradually accumulate those temporally static moving objects, which are expected to give rise to the possible error of taking "static motions" as part of the non-salient backgrounds. As we can see in Fig. 17, because the walking person stands still at identical position for over 160 frames (from Fig. 17(a)–(c)), and the original correct low-rank prior (Fig. 17(d)) was gradually updated (Fig. 17(e)) with perhaps accumulated potential errors. Hence, the salient motion detection in the #893 frame fails to detect some parts of the target object. One way to overcome this limitation is to further develop new foreground models aided by certain motion clues (e.g., flux tensor or optical flow), so that we can keep assigning saliency values to the regions that are similar to the foreground templates, even if such regions belong to the currently-stopped objects which have been moving till this very moment.

Besides, our method is a bit time-consuming, which can only achieve the time performance of three frames per second for the 300*300 video frames (on a computer with Quad Core i7-3770 3.4 GHz, 8GB RAM). Table 12 demonstrates the detailed time cost comparison results. In fact, for the time-consuming Multiple Observation Coincidence strategy documented in Section 7.3, it can be implemented in a parallel fashion, which should greatly improve the

Fig. 14. Performance comparison on five additional non-stationary video sequences. Ground Truth is marked with red rectangle in the first row, and rows 2–6 depict the results of our method, MOD13 [13], SuBSENSE14 [6], ViBe11 [32], and KNN06 [18], respectively. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

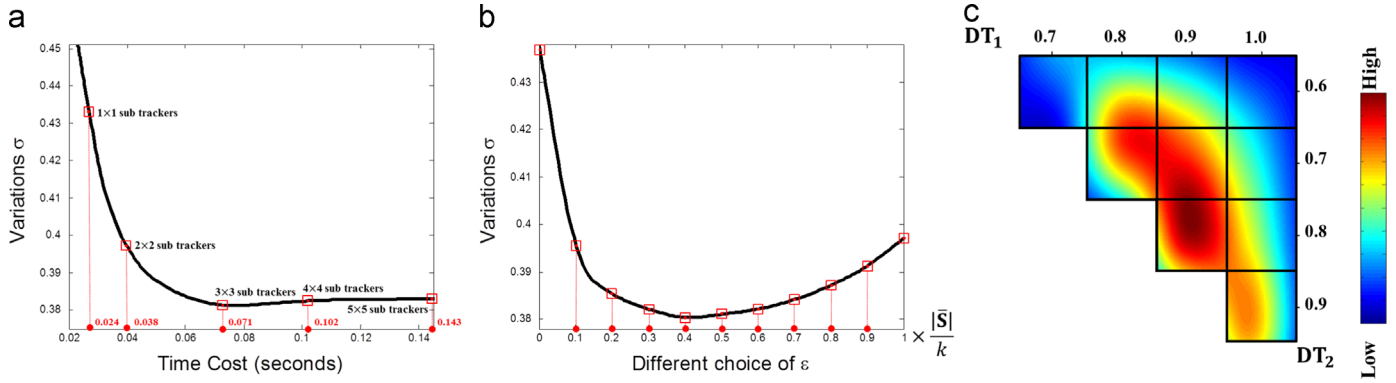

Fig. 15. Parameter selection analysis. (a) demonstrates the trade-off between the standard variation of tracked background frame batch and the time consumption of adopting different sub-tracker number, (b) shows the performance analysis for different choices of ε, (c) demonstrates the performance with different choices of DT hard thresholds.

time efficiency of our method. Meanwhile, we can perhaps make trade-off between efficiency and performance by appropriately reducing the RPCA low-rank revealing iteration times and/or introducing an incremental SVD method to expedite the convergence speed.

9. Conclusion and future work

In this paper, we have systematically presented a novel and versatile method to address a suite of research challenges

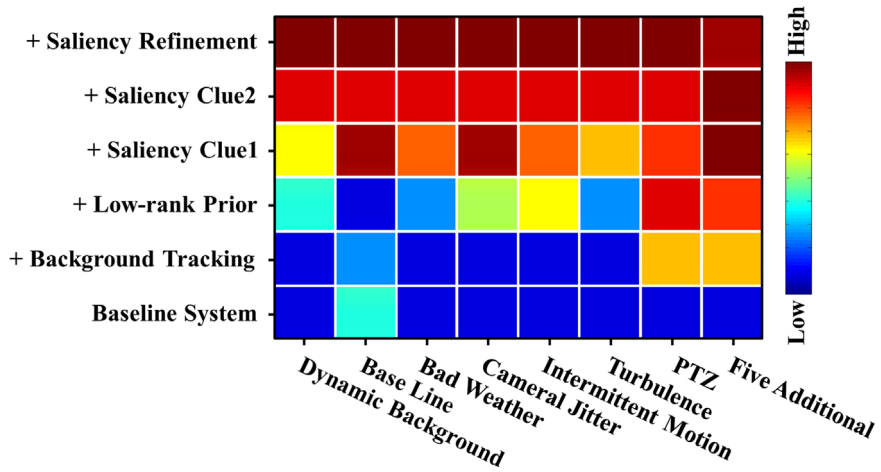

Fig. 16. The evaluation results for each component involved in our method (best viewed in color). The colors from blue to red indicate the averaged F-measure results from low to high. The horizontal axis lists the eight video categories, and the vertical axis indicates the different components involved in our method. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

Table 12
Time cost comparisons. The left column demonstrates the time performance of the state-of-the-art methods, and the right column demonstrates the time performance of each component involved in our method. All of these methods (components) run on a computer with Quad Core i7-3770 3.4 GHz CPU, 8GB RAM.

Method	Precision	Recall	F-measure
SuBSENSE14	0.043	Baseline System	0.1824
MOD13	0.213	+Background Tracking	0.1967
FTSG14	0.114	+Low-rank Prior	0.1978
CwisarDH14	0.138	+Saliency Clue 1	0.2193
ViBe11	0.028	+Saliency Clue 2	0.2434
KNN06	0.031	+Saliency Refinement	0.4868

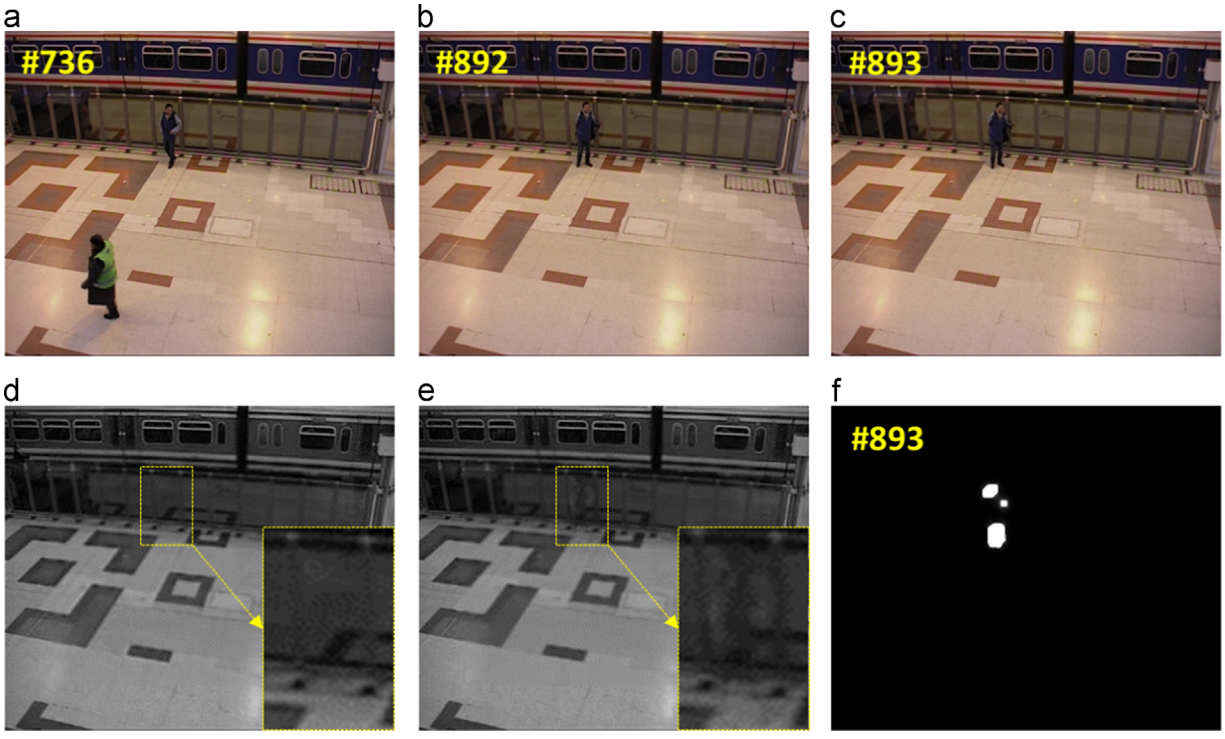

Fig. 17. Demonstration of our method's limitation. (a–c) show that the walking person stands still at the same position for over 160 video frames, (d) demonstrates the low-rank prior obtained before the #736 frame, and (e) demonstrates the low-rank prior after #892 frame. (f) is the #893 frame's salient motion detection result. Obviously, some parts of the standing-still person are not properly detected.

encountered in the motion tracking of non-stationary videos. The central idea of our method is to collectively incorporate the respective advantages of matching (i.e., background tracking) and modeling (i.e., low-rank information modeling for background) based methods into a unified low-rank analysis driven tracking-by-detection framework. Key novel technical elements include multiple local low-rank swarm based background trackers, a suite of new computing schemes involving divide-and-conquer strategy and low-rank semantic coherency analysis, low-rank prior biased salient motion detection based on aligned RPCA, and online updating of low-rank background prior assisted by multiple saliency clues, all of which contribute to the robust salient motion detection for long-term non-stationary videos with proved excellent performance. Consequently, our novel computational schemes promise to combat many obstinate problems induced by camera jitter, dynamic background, intermittent motion, and occasionally-occluded contextual interactions. Our comprehensive experiments and extensive comparisons with other state-of-the-art methods have demonstrated our method's superiorities in terms of robustness, accuracy, reliability, and versatility.

Our ongoing research endeavors are concentrated on extending our key ideas to correctly perform the motion recognition in hand-hold videos, saliency motion guided object tracking, multi-view non-stationary video expression, etc.

Conflict of interest

None declared.

Acknowledgments

This research is supported in part by Innovation Foundation of BUAA for Ph.D. Graduates, National Natural Science Foundation of China (Nos. 61190120, 61190121, 61190125, 61300067, and 61532002) and National Science Foundation of USA (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.2015.09.033.

References

[1] C. Chen, S. Li, H. Qin, A. Hao, Real-time and robust object tracking in video via low-rank coherency analysis in feature space, Pattern Recognit. 48 (9) (2015) 2885–2905.
[2] 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.
[3] V. Navalpakkam, L. Itti, An integrated model of top-down and bottom-up attention for optimizing detection speed, in: IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, 2006, pp. 2049–2056.
[4] V. Rengarajan, A. Punnappurath, A. Rajagopalan, G. Seetharaman, Efficient change detection for very large motion blurred images, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2014, pp. 315–322.
[5] Y. Wang, P. Jodoin, F. Porikli, J. Konrad, Y. Benezeth, P. Ishwar, CDnet 2014: an expanded change detection benchmark dataset, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2014, pp. 393–400.
[6] P.L. StCharles, G.A. Bilodeau, R. Bergevin, SuBSENSE: a universal change detection method with local adaptive sensitivity[J]. Image Processing, IEEE Transactions on 24 (1) (2015) 359–373.
[7] R. Wang, F. Bunyak, G. Seetharaman, K. Palaniappan, Static and moving object detection using flux tensor with split Gaussian models, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2014, pp. 420–424.
[8] C. Wren, A. Ali, D. Trevor, A. Pentland, Pfinder: real-time tracking of the human body, IEEE Trans. Pattern Anal. Mach. Intell. 19 (7) (1997) 780–785.
[9] C. Stauffer, W. Grimson, Learning patterns of activity using real-time tracking, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 747–757.
[10] S. Varadarajan, P. Miller, H. Zhou, Spatial mixture of Gaussians for dynamic background modelling, in: IEEE Conference on Advanced Video and Signal Based Surveillance, 2013, pp. 63–68.
[11] J. Liu, J. Luo, M. Shah, Recognizing realistic actions from videos in the wild, in: IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 1996–2003.
[12] C. Xu, J. Liu, B. Kuipers, Moving object segmentation using motor signals, in: European Conference on Computer Vision, 2012, pp. 676–689.
[13] X. Zhou, C. Yang, W. Yu, Moving object detection by detecting contiguous outliers in the low-rank representation, IEEE Trans. Pattern Anal. Mach. Intell. 35 (3) (2013) 597–610.
[14] Z. Gao, L. Cheong, Y. Wang, Block-sparse RPCA for salient motion detection, IEEE Trans. Pattern Anal. Mach. Intell. 36 (10) (2014) 1975–1987.
[15] C. Chen, S. Li, H. Qin, A. Hao, Structure-sensitive saliency detection via multi-level rank analysis in intrinsic feature space, IEEE Trans. Image Process. 24 (8) (2015) 2303–2316.
[16] S. Huwer, H. Niemann, Adaptive change detection for real-time surveillance applications, in: IEEE International Workshop on Visual Surveillance, 2000, pp. 37–46.
[17] C. Stauffer, W.E.L. Grimson, Adaptive background mixture models for real-time tracking, in: IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, 1999, pp. 246–252.
[18] Z. Zivkovic, F. van der Heijden, Efficient adaptive density estimation per image pixel for the task of background subtraction, Pattern Recognit. Lett. 27 (7) (2006) 773–780.
[19] A. Elgammal, R. Duraiswami, D. Harwood, L. Davis, Background and foreground modeling using nonparametric kernel density estimation for visual surveillance, Proc. IEEE 90 (7) (2002) 1151–1163.
[20] X. Zhao, Y. Satoh, H. Takauji, S. Kaneko, K. Iwata, R. Ozaki, Object detection based on a robust and accurate statistical multi-point-pair model, Pattern Recognit. 44 (6) (2011) 1296–1311.
[21] G. Bilodeau, J. Jodoin, N. Saunier, Change detection in feature space using local binary similarity patterns, in: International Conference on Computer and Robot Vision, 2013, pp. 106–112.
[22] D. Liang, S. Kaneko, M. Hashimoto, K. Iwata, X. Zhao, Y. Satoh, Robust object detection in severe imaging conditions using co-occurrence background model, Int. J. Optomechatron. 8 (1) (2014) 14–29.
[23] Y. Benezeth, P.-M. Jodoin, B. Emile, H. Laurent, C. Rosenberger, Comparative study of background subtraction algorithms, J. Electron. Imaging 19 (3) (2010) 033003–033003-12.
[24] L. Maddalena, A. Petrosino, A self-organizing approach to background subtraction for visual surveillance applications, IEEE Trans. Image Process. 17 (7) (2008) 1168–1177.
[25] L. Maddalena, A. Petrosino, A fuzzy spatial coherence-based approach to background/foreground separation for moving object detection, Neural Comput. Appl. 19 (2) (2010) 179–186.
[26] L. Maddalena, A. Petrosino, The SOBS algorithm: what are the limits?, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops, IEEE, Providence, Rhode Island, USA, 2012, pp. 21–26.
[27] M.D. Gregorio, M. Giordano, A WiSARD-based approach to CDnet, in: Brazilian Congress on Computational Intelligence, 2013, pp. 172–177.
[28] M.D. Gregorio, M. Giordano, Change detection with weightless neural networks, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2014, pp. 409–413.
[29] P. Dollar, V. Rabaud, G. Cottrell, S. Belongie, Behavior recognition via sparse spatio-temporal features, in: IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005, pp. 65–72.
[30] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, L. Van Gool, A comparison of affine region detectors, Int. J. Comput. Vis. 65 (1–2) (2005) 43–72.
[31] G. Kim, C. Faloutsos, M. Hebert, Unsupervised modeling of object categories using link analysis techniques, in: IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.
[32] O. Barnich, M. Van Droogenbroeck, ViBe: a universal background subtraction algorithm for video sequences, IEEE Trans. Image Process. 20 (6) (2011) 1709–1724.
[33] K. Kim, D. Lee, I. Essa, Detecting regions of interest in dynamic scenes with camera motions, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 1258–1265.
[34] W. Kim, J. Han, Video saliency detection using contrast of spatiotemporal directional coherence, IEEE Signal Process. Lett. 21 (10) (2014) 1250–1254.
[35] Y. Fang, W. Lin, Z. Chen, C. Tsai, C. Lin, A video saliency detection model in compressed domain, IEEE Trans. Circuits Syst. Video Technol. 24 (1) (2014) 27–38.
[36] J. Wright, Y. Peng, Y. Ma, Robust principal component analysis: exact recovery of corrupted low-rank matrices by convex optimization, in: Advances in Neural Information Processing Systems, 2009, pp. 2080–2088.
[37] T. Zhou, D. Tao, Bilateral random projections, in: IEEE International Symposium on Information Theory Proceedings, 2011, pp. 1286–1290.
[38] J. Yan, M. Zhu, H. Liu, Y. Liu, Visual saliency detection via sparsity pursuit, IEEE Signal Process. Lett. 17 (8) (2010) 739–742.
[39] 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.

[40] T. Zhang, B. Ghanem, S. Liu, N. Ahuja, Robust visual tracking via structured multi-task sparse learning, Int. J. Comput. Vis. 101 (2) (2013) 367–383.
[41] N. Goyette, P. Jodoin, F. Porikli, J. Konrad, P. Ishwar, Changedetection. net: a new change detection benchmark dataset, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2012, pp. 1–8.
[42] S. Roweis, Em algorithms for PCA and SPCA, in: Neural Information Processing Systems, 1998, pp. 626–632.
[43] Y. Peng, A. Ganesh, J. Wright, W. Xu, Y. Ma, RASL: robust alignment by sparse and low-rank decomposition for linearly correlated images, IEEE Trans. Pattern Anal. Mach. Intell. 34 (11) (2012) 2233–2246.
[44] A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, IEEE Trans. Pattern Anal. Mach. Intell. 2 (1) (2009) 183–202.
[45] J. Cai, J. Emmanuel, Z. Shen, A singular value thresholding algorithm for matrix completion, SIAM J. Optim. 20 (4) (2010) 1956–1982.
[46] A. Radhakrishna, S. Appu, S. Kevin, L. Aurelien, F. Pascal, S. Sabine, Slic Superpixels, in: EPFL Technical Report, 2010.
[47] Y. Wu, J. Lim, M. Yang, Online object tracking: a benchmark, in: IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 2411–2418.

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. degrees in Computer Science at Beihang University. His research interests are on virtual reality, computer simulation, computer graphics, geometric modeling, image processing, and computer vision.