Incremental Learning With Saliency Map for Moving Object Detection

Yanwei Pang, *Senior Member, IEEE*, Li Ye, Xuelong Li, *Fellow, IEEE*, and Jing Pan

***Abstract*— Moving object detection is a key to intelligent video analysis. On the one hand, what moves are not only interesting objects but also noise and cluttered background. On the other hand, moving objects without rich texture are prone to not be detected. Therefore, there are undesirable false alarms and missed alarms in the results of many algorithms of moving object detection. To reduce the false alarms and missed alarms, in this paper we propose to incorporate a saliency map into an incremental subspace analysis framework in which the saliency map makes the estimated background have less of a chance than the foreground (i.e., moving objects) to contain salient objects. The proposed objective function systematically takes into account the properties of sparsity, low rank, connectivity, and saliency. An alternative minimization algorithm is proposed to seek the optimal solutions. The experimental results on both the Perception Test Images Sequences data set and Wallflower data set demonstrate that the proposed method is effective in reducing false alarms and missed alarms.**

***Index Terms*— Motion, object detection, saliency map, subspace analysis.**

I. Introduction

OBJECT detection is the basis of intelligent video analysis. Generally, object recognition, action and behavior recognition, and tracking rely on the detected objects. In a sequence of images there are both moving and static objects. In this paper the focus is on detecting moving objects in a video.

Moving object detection is related to but different from class-specific object detection and general salient object detection. Pedestrian detection [1], [2], face detection, motion blur detection [3], [4], and hand detection are instances of class-specific object detection. The task of moving object detection is to detect semantically meaningful moving objects. Predefined classes of moving objects should be detected by a moving object detection algorithm. Moreover, other semantically meaningful objects should also be detected even though their classes are not predefined. Except for the semantically meaningful moving objects, the meaningless moving objects should not be detected. Examples of meaningless moving objects include water ripples, waving trees (leaves), shadows, noisy data, and the one caused by variations of illumination. However, the moving object detection algorithm relying merely on motion information is prone to incorrectly classify such meaningless moving objects as meaningful ones. The corresponding error is called false alarms. However, a salient object detection algorithm tends to correctly discard the meaningless objects. Hence, in this paper we propose to incorporate the output (i.e., saliency map) of a salient object algorithm into a subspace-analysis-based objective function so that the problem of false alarms can be alleviated.

It is noted that our method is also capable of alleviating the problem of missed alarms. Existing moving object detection algorithms tend to classify flat regions (i.e., textureless regions) inside an object and moving regions with a similar appearance (texture) as background and thus such regions may be missed. The state-of-the-art salient object detection algorithm can an output large value of saliency map at such regions. Utilizing the saliency map, our method has the ability to classify such regions as foreground.

In summary, we present an objective function that unifies the subspace analysis of background and saliency map. The objective function consists of four terms: 1) saliency map; 2) sparsity; 3) connectivity; and 4) low rank. An alternative minimization algorithm is proposed to find the optimal solution. The significant advantage compared with the previous subspace-based approaches is that saliency map is used to guide the results to have less false and missed alarms. The proposed method is named moving object detection with saliency map (MODSM). It is natural that an ideal saliency map [see the bottom of Fig. 1(a) and (b)] is desirable for the proposed method. However, even a relatively unsatisfying saliency map [see the bottom of Fig. 1(c) and (d)] can also play a positive role in the proposed MODSM method. Of course, a completely bad saliency map has a negative influence on moving object detection. Fortunately, great progress of salient object detection has been achieved [5]–[7] and their fruits can be borrowed for moving object detection.

Manuscript received September 17, 2016; revised November 11, 2016; accepted November 16, 2016. Date of publication November 18, 2016; date of current version March 5, 2018. This work was supported in part by the National Basic Research Program of China under Grant 2014CB340400, in part by the National Natural Science Foundation of China under Grant 61632081 and Grant 61271412, and in part by the Research Fund of Hainan Tropical Ocean University under Grant QYXB201501. This paper was recommended by Associate Editor A. Loui.

Y. Pang and L. Ye are with the School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China (e-mail: pyw@tju.edu.cn; liyetju@tju.edu.cn).

X. Li is with the Center for OPTical IMagery Analysis and Learning (OPTIMAL), State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an 710119, China (e-mail: xuelong_li@opt.ac.cn).

J. Pan is with the School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China, and also with the School of Electronic Engineering, Tianjin University of Technology and Education, Tianjin 300222, China (e-mail: jingpan23@gmail.com).

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

Digital Object Identifier 10.1109/TCSVT.2016.2630731

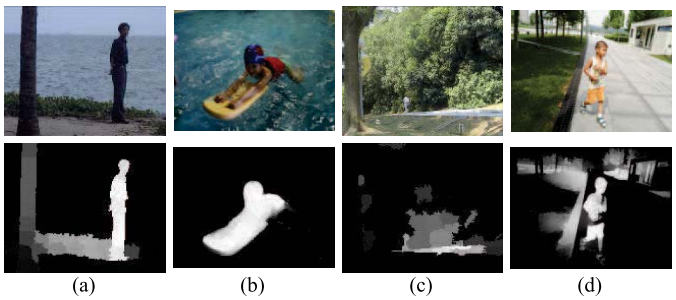


Fig. 1. Images (top) and their saliency maps (bottom).

It is noted that in our method the elements of the foreground vector to be optimized are constrained in the range of [0, 1] instead of in the binary set $\{0, 1\}$. Generally, the computational cost of optimizing the real foreground vector is much smaller than that of binary foreground vector. A time-consuming graph cut algorithm is usually required to solve the binary foreground vector (see [8], [9]). By contrast, there exists a close form for optimizing the real foreground because the corresponding objective function is a quadratic function. Moreover, setting the foreground vector between 0 and 1 instead of a binary value is able to result in better detection accuracy.

Several video saliency detection or segmentation methods that use the motion information to improve the results have been developed [10], [11]. Despite the initial success, their performance cannot arrive at the level of state-of-the-art low-rank-based and subspace-based methods [8], [12]–[19].

The rest of this paper is organized as follows. We review related work in Section II. The proposed method is given in Section III. The experimental results are provided in Section IV. We then conclude this paper in Section V.

II. Related Work

The methods of moving object detection can be divided into four categories [20]: 1) detecting followed by tracking [21]–[23] and subtracting frames [24]–[26]; 2) modeling background by a density function [14], [27]–[30]; 3) modeling background by subspace [13], [15], [19], [31], [32]; and 4) modeling background by a low-rank matrix [8]. The last two categories dominate the state-of-the-art methods and are closely related to our work. Note that moving object detection methods can also be divided into incremental methods and batch methods. Our method belongs to an incremental one.

A. Subtracting Frames

These kinds of methods detect moving objects based on the differences between adjacent frames [24]–[26]. However, these methods were proved not robust to illumination variations, changing background, camera motion, and noise.

B. Modeling Background by Density Function

This strategy assumes that the background is stationary and can be modeled by Gaussian, mixture of Gaussians, or Dirichlet process mixture models [14], [27]–[29]. The foreground (moving regions) can then be obtained by subtracting the current frame with the background model.

C. Modeling Background by Subspace

Instead of using a density function, subspace-based methods model the background as a linear combination of the bases of a subspace [13], [15], [19], [31], [32]. Because the subspace can be updated in an incremental (online) manner, its efficiency is much higher. These kinds of subspace-based algorithms need to impose constraints on the foreground in order to obtain valid solutions. Foreground sparsity is one of the widely used constraints, which implies that the area of moving objects is small relative to the background. Principal component pursuit (PCP) [33] is a classical subspace method for background modeling. Because of its close relationship to our method, we briefly describe it. Mathematically, let $\boldsymbol{O} \in \mathbb{R}^{n\times m}$ be the observation matrix containing m frames. Each column of $\boldsymbol{O}$ corresponds to a vectorized frame that has n pixels. Generally, $\boldsymbol{O}$ can be decomposed as $\boldsymbol{O} = \boldsymbol{B} + \boldsymbol{F}$, where $\boldsymbol{B} \in \mathbb{R}^{n\times m}$ is the low rank matrix (background) and $\boldsymbol{F} \in \mathbb{R}^{n\times m}$ is the sparse matrix (foreground). The PCP method can be formulated as the following minimization problem:

$$\begin{aligned} &\min \ \|\boldsymbol{B}\|_* + \lambda\|\boldsymbol{F}\|_1 \\ &\text{s.t. } \boldsymbol{B} + \boldsymbol{F} = O \end{aligned} \tag{1}$$

where the nuclear norm $\|\boldsymbol{B}\|_*$ is used to estimate the rank of $\boldsymbol{B}$ and the l_1 norm of $\boldsymbol{F}$ is used to measure the sparsity of the foreground $\boldsymbol{F}$. The constraint $\boldsymbol{B} + \boldsymbol{F} = \boldsymbol{O}$ makes that the minimization of rank of the background and the sparsity of the foreground are meaningful in the sense of the sum of the background and the foreground approaches to the observation. Without this constraint, traditional robust subspace methods can deal only with noise and outliers [34]–[37]. The method [38] improves PCP by taking the foreground connectivity (i.e., foreground structure) into account. RFDSA [13] takes smoothness and arbitrariness constraints into account.

However, RFDSA [13], PCP [33], and the method [38] are batch algorithms. Their detection speed cannot arrive at real-time level. Therefore, incremental (online) subspace methods are crucial for real-time detection [39]. He *et al.* [15] proposed an online subspace tracking algorithm called Grassmannian robust adaptive subspace tracking algorithm (GRASTA). Similar to PCP, GRASTA also explores l_1 norm for imposing sparsity on foreground. However, the GRASTA algorithm does not utilize any connectivity (also known as smoothness) property of foreground. The A Grassmannian online subspace updates with structured sparsity (GOSUS) algorithm [19] imposes a connectivity constraint on the objective function by grouping the pixels with a superpixel method and encouraging sparsity of the groups. Because of the large computational cost of the superpixel algorithm [40], GOSUS is not as efficient as GRASTA.

D. Modeling Background by Low-Rank Matrix

Low-rank modeling is effective in video representation [18]. A sequence of vectorized images is represented as a matrix

and the matrix is approximated by the sum of matrices of vectorized foreground, background, and noise [8]. It is rational to assume that the background matrix is low rank. Detecting contiguous outliers in the low-rank representation (DECOLOR) [8] is considered as one of the most successful low-rank-based algorithms. In DECOLOR, both foreground sparsity and contiguity (connectivity) are taken into account. It can be interpreted as a penalty regularized RPCA. However, the matrix computation can be started only if all the predefined numbers of successive images are available. Obviously, such a batch method is not suitable for real-time video analysis due to its low efficiency. ISC [9] blue (incremental, sparsity, and connectivity) and COROLA [17] are incremental versions of DECOLOR. ISC and COROLA transform a low-rank method to a subspace one. Our method differs from ISC in introducing saliency map into a unified objective function. In addition to the term of saliency map, the constraint on the foreground vector is different from the ISC method. The foreground vector in [9] is a binary vector with its element $x_i \in \{0, 1\}$, whereas in our method, the foreground vector (its negative vector is called the background vector in our method) is generalized to a vector with its element $0 \leq x_i \leq 1$. This difference makes our method much more robust than ISC (See Section IV-B). Because of the difference mentioned above, the optimization algorithm of the foreground vector (and its corresponding background vector) of our method is completely different from that of ISC. In ISC, the graph cut algorithm [41], [42] is used for optimization, whereas our method formulates it as a quadratic function and finding the optimal solution by computing its derivative. Our algorithm is more efficient than the graph cut algorithm.

The low-rank methods and subspace methods impose sparsity and connectivity (also known as smoothness) on foreground and impose low-rank or principal components on background. In addition to such properties, in this paper, we propose to impose saliency map on background. Because the background and foreground are nonoverlapping, exclusive, and complementary, imposing the saliency map on the background is equivalent to imposing the saliency map on the foreground.

III. Proposed Method

The proposed method belongs to an incremental-subspace-based moving object detection method. The main novelty of the proposed method lies in employing a saliency map to form a new objective function, resulting in fewer false and missed alarms.

A. Input and Output

The input of our algorithm is a sequence of frames (images) from $\boldsymbol{o}_s$ to $\boldsymbol{o}_t$. Denote $\boldsymbol{o} \in \mathrm{R}^{N\times 1}$ the current image and denote $\boldsymbol{o}_i$ the i-th pixel of $\boldsymbol{o}$. There are N pixels in an image. The goal is to find the locations of the moving objects (i.e., foreground) in the current image $\boldsymbol{o}$. The foreground locations are represented by a foreground indicator vector $\boldsymbol{f} \in \{0, 1\}^N$. The ith element f_i of $\boldsymbol{f}$ is equal to either zero or one

$$f_i = \begin{cases} 0 & \text{if pixel } i \text{ is classified as background} \\ 1 & \text{if pixel } i \text{ is classified as foreground.} \end{cases} \tag{2}$$

The foreground indicator vector $\boldsymbol{f}$ is obtained by binarizing the background vector $\boldsymbol{b} \in \mathrm{R}^{N\times 1}$ with a threshold t

$$f_i = \begin{cases} 0 & \text{if } b_i \geq t \\ 1 & \text{if } b_i < t \end{cases} \tag{3}$$

where $b_i \in [0, 1]$ is the ith element of $\boldsymbol{b}$. The possibility of pixel i being background increases with the value of b_i and the possibility of pixel i being foreground decreases with the increasing value of b_i. For the sake of completeness and clarity, we define a foreground vector $\bar{\boldsymbol{f}}$. The foreground indicator vector $\boldsymbol{f}$ is a binary version of the foreground vector $\bar{\boldsymbol{f}}$. The negative of the background vector $\boldsymbol{b}$ is identical to the foreground vector $\bar{\boldsymbol{f}}$ (i.e., $\bar{\boldsymbol{f}} = 1 - \boldsymbol{b}$).

B. Problem Formulation

As stated above [see (2) and (3)], the foreground indicator vector can be obtained by the binarizing background vector $\boldsymbol{b}$. The problem is how to compute $\boldsymbol{b}$ once a frame (image) $\boldsymbol{o}$ is given. In this paper, we formulate the problem of computing $\boldsymbol{b}$ as the following minimization problem:

$$\min_{\boldsymbol{b},\boldsymbol{U},\boldsymbol{v}} \sum_{i=1}^{N} \Big[\frac{1}{2} b_i (\boldsymbol{U}_i \boldsymbol{v} - \boldsymbol{o}_i)^2 + \beta(1 - b_i) - \alpha b_i (1 - s_i) \Big] + \lambda \|\boldsymbol{D}\boldsymbol{b}\|_1. \tag{4}$$

We first describe the meaning of each variable in (4) and then explain the roles of the four terms of (4). In (4), $\boldsymbol{U} \in \mathbb{R}^{N\times m}$ is a subspace matrix whose columns are orthonormal and m is the number of columns of $\boldsymbol{U}$, and $\boldsymbol{U}_i$ stands for the ith row of $\boldsymbol{U}$. The coefficient vector $\boldsymbol{v} \in \mathbb{R}^{m\times 1}$ is the low-dimensional representation of the image $\boldsymbol{o}$ in the subspace spanned by the rows of $\boldsymbol{U}$. $s_i \in [0, 1]$ is the ith element of the vector $\boldsymbol{s} \in \mathbb{R}^{N\times 1}$ of a saliency map obtained by some salient object detection algorithms such as those in [43]. The value of s_i reflects the confidence that the pixel i belongs to a salient object. The matrix $\boldsymbol{D} \in \mathbb{R}^{2N\times N}$ is a difference matrix, $\boldsymbol{D} = [\boldsymbol{D}_h, \boldsymbol{D}_v]^T$, where $\boldsymbol{D}_h$ and $\boldsymbol{D}_v$ [13] are forward finite-difference operators in the horizontal and vertical directions, respectively. The weights α, β, and λ are used for balancing the four terms of (4), which are to be discussed in the following paragraph.

The four terms $b_i(\boldsymbol{U}_i\boldsymbol{v} - \boldsymbol{o}_i)^2$, $(1 - b_i)$, $-b_i(1 - s_i)$, and $\|\boldsymbol{D}\boldsymbol{b}\|_1$ are called the background reconstruction term, foreground sparsity term, object saliency term, and connectivity term, respectively. The main novelty of the proposed method lies in the object saliency term.

1) Background Reconstruction Term: In the background reconstruction term $b_i(\boldsymbol{U}_i\boldsymbol{v} - \boldsymbol{o}_i)^2$, $\boldsymbol{U}_i\boldsymbol{v}$ is the reconstruction of the background [15], [19]. Therefore, $\boldsymbol{U}_i\boldsymbol{v} - \boldsymbol{o}_i$ measures how well $\boldsymbol{U}_i\boldsymbol{v}$ approaches $\boldsymbol{o}_i$. It is noted that the element b_i of the background vector makes the estimation focus on the background region.

2) Foreground Sparsity Term: It is well known that the foreground is small and sparse relative to the background. Consequently, minimizing the sum of the foreground term $(1 - b_i)$ makes the estimated foreground much sparser than the background.

3) Connectivity Term: It is reasonably assumed that if a pixel belongs to background (or foreground), then its neighbors also belong to background (or foreground). Therefore, minimizing the connectivity term $\|\boldsymbol{Db}\|_1$ makes the estimated background and foreground smooth as far as possible.

4) Object Saliency Term: Minimizing the term $-b_i(1-s_i)$ makes the estimated background $\boldsymbol{b}$ have less chance than the foreground to contain salient objects. Moving objects such as pedestrian, car, and dog are indeed salient objects in a video. Therefore, the proposed method is capable of making the estimated foreground to have high-level semantic objects and fewer false alarms. The saliency map term $-b_i(1-s_i)$ is the main novelty of this paper.

An empirical method for setting the weights α, β, and λ is given in Table III.

C. Problem Solution

We first discuss that there exists a solution to minimization problem (4) and then describe how to find the solution.

The minimization problem expressed as (4) consists of three unknown variables. We adopt an alternating minimization algorithm to seek the optimal variables $\boldsymbol{b}$, $\boldsymbol{U}$, and $\boldsymbol{v}$ in turn. In deriving how to seek each optimal variable, theoretical analysis is given to guarantee the existence of the solution.

Because the last term $||\boldsymbol{Db}||$ of (4) is the one norm of the multiplication of the difference matrix $\boldsymbol{D}$ and the background vector $\boldsymbol{b}$, it is difficult to directly seek the optimal solution of $\boldsymbol{b}$. To circumvent the difficulty, we let $\boldsymbol{w}=\boldsymbol{b}$ and $\boldsymbol{c}=\boldsymbol{Dw}$. Accordingly, the minimization problem can be equivalently expressed as

$$\min_{\boldsymbol{b},\boldsymbol{U},\boldsymbol{v}} \sum_{i=1}^{N}\Big[\frac{1}{2}b_i(\boldsymbol{U}_i\boldsymbol{v}-\boldsymbol{o}_i)^2+\beta(1-b_i) - \alpha b_i(1-s_i)\Big]+\lambda\|\boldsymbol{c}\|_1 \tag{5}$$

$$\text{s.t. } \boldsymbol{w}=\boldsymbol{b}, \quad \boldsymbol{c}=\boldsymbol{Dw}. \tag{6}$$

With the technique of Lagrangian multiplier, the constrained minimum problem expressed as (5) and (6) can be converted into the following unconstrained problem:

$$\min_{\boldsymbol{b},\boldsymbol{U},\boldsymbol{v},\boldsymbol{c},\boldsymbol{w},} \sum_{i=1}^{N}\Big[\frac{1}{2}b_i(\boldsymbol{U}_i\boldsymbol{v}-\boldsymbol{o}_i)^2+\beta(1-b_i) - \alpha b_i(1-s_i)\Big]+\lambda\|\boldsymbol{c}\|_1+\frac{\mu}{2}\|\boldsymbol{w}-\boldsymbol{b}\|_2^2 + \boldsymbol{x}^T(\boldsymbol{w}-\boldsymbol{b})+\frac{\mu}{2}\|\boldsymbol{c}-\boldsymbol{Dw}\|_2^2+\boldsymbol{y}^T(\boldsymbol{c}-\boldsymbol{Dw}). \tag{7}$$

The terms $\mu/2\|\boldsymbol{w}-\boldsymbol{b}\|_2^2$ and $\boldsymbol{x}^T(\boldsymbol{w}-\boldsymbol{b})$ in (7) are obtained by transforming the constraint $\boldsymbol{w}=\boldsymbol{b}$ in (6) into the unconstrained optimization function with the technique of the Lagrangian multiplier. It is noted that $\boldsymbol{x}^T(\boldsymbol{w}-\boldsymbol{b})$ is the term of the Lagarangian function with the vector $\boldsymbol{x}$ being the Largrangian multiplier. The term $\mu/2\|\boldsymbol{w}-\boldsymbol{b}\|_2^2$ is the penalty term (or called regularization term), which is used for guaranteeing that a meaning solution can be obtained. Similarly, the terms $\mu/2\|\boldsymbol{c}-\boldsymbol{Dw}\|_2^2$ and $\boldsymbol{y}^T(\boldsymbol{c}-\boldsymbol{Dw})$ in (7) are obtained by transforming the constraint $\boldsymbol{c}=\boldsymbol{Dw}$ in (6) into the unconstrained optimization function with the technique of the Lagrangian multiplier, where $\boldsymbol{y}^T(\boldsymbol{c}-\boldsymbol{Dw})$ and $\mu/2\|\boldsymbol{c}-\boldsymbol{Dw}\|_2^2$ are the term of the Lagrangian function and the penalty term, respectively.

Optimization problem (7) is easier to be solved than original optimization problem (4). The alternating manner of solving (7) is given as follows:

1) b-Step: The goal is to seek the optimal $\boldsymbol{b}$ when $\boldsymbol{U}$, $\boldsymbol{v}$, $\boldsymbol{c}$, $\boldsymbol{w}$, $\boldsymbol{x}$, and $\boldsymbol{y}$ are fixed. In this situation, (7) is a quadratic function with respect to $\boldsymbol{b}$, which is differential because of the continuous nature of $\boldsymbol{b}$. Therefore, there is a unique solution to (7) in the sense of variable $\boldsymbol{b}$. Computing the derivative of the sum of the terms of (7) and letting the result be zero yield

$$b_i=\frac{\beta+\mu w_i+x_i-\frac{1}{2}(\boldsymbol{U}_i\boldsymbol{v}-\boldsymbol{o}_i)^2+\alpha(1-s_i)}{\mu}. \tag{8}$$

The influence of the saliency map s_i on the background b_i is intuitive: b_i decreases with increasing of s_i. Hence, the proposed method tends to let the estimated background not contain moving and salient objects; meanwhile, it tends to let the estimated foreground contain moving and salient objects.

2) c-Step: The goal is to seek the optimal $\boldsymbol{c}$ when $\boldsymbol{b}$, $\boldsymbol{U}$, $\boldsymbol{v}$, $\boldsymbol{w}$, $\boldsymbol{x}$, and $\boldsymbol{y}$ are fixed. Omitting irrelevant terms, it is reduced to the following traditional optimization problem:

$$\boldsymbol{c}=\arg\min_{\boldsymbol{c}}\lambda\|\boldsymbol{c}\|_1+\frac{\mu}{2}\|\boldsymbol{c}-\boldsymbol{Dw}\|_2^2+\boldsymbol{y}^T(\boldsymbol{c}-\boldsymbol{Dw}) \tag{9}$$

$$=\arg\min_{\boldsymbol{c}}\frac{\lambda}{\mu}\|\boldsymbol{c}\|_1+\frac{1}{2}\|\boldsymbol{c}-\boldsymbol{m}\|_2^2 \tag{10}$$

where

$$\boldsymbol{m}=\boldsymbol{Dw}-\frac{\boldsymbol{y}}{\mu}. \tag{11}$$

Equation (10) is a standard minimization problem [44] that is guaranteed to have unique solution. According to [13] and [44], the solution is given by

$$\boldsymbol{c}=S_{\frac{\lambda}{\mu}}\left(\boldsymbol{Dw}-\frac{\boldsymbol{y}}{\mu}\right) \tag{12}$$

with the soft-thresholding (shrinkage) operator $S_\varepsilon(x)$ being

$$S_\varepsilon(x)=\operatorname{sgn}(x)\max(|x|-\varepsilon,0)=\begin{cases}x-\varepsilon, & x>\varepsilon\\ x+\varepsilon, & x<-\varepsilon\\ 0 & \text{else.}\end{cases} \tag{13}$$

Equation (12) means that if the value of $\boldsymbol{Dw}-\boldsymbol{y}/\mu$ is larger than λ/μ, then the value of $\boldsymbol{c}$ is equal to $\boldsymbol{Dw}-\boldsymbol{y}/\mu-\lambda/\mu$. Otherwise, the value of $\boldsymbol{c}$ is equal to $\boldsymbol{Dw}-\boldsymbol{y}/\mu+\lambda/\mu$.

3) w-Step: The goal is to seek the optimal $\boldsymbol{w}$ when $\boldsymbol{b}$, $\boldsymbol{U}$, $\boldsymbol{v}$, $\boldsymbol{c}$, $\boldsymbol{x}$, and $\boldsymbol{y}$ are fixed. Omitting irrelevant terms, it is reduced to the following minimization problem:

$$\boldsymbol{w}=\arg\min_{\boldsymbol{w}}\frac{\mu}{2}\|\boldsymbol{w}-\boldsymbol{b}\|_2^2+\boldsymbol{x}^T(\boldsymbol{w}-\boldsymbol{b}) +\frac{\mu}{2}\|\boldsymbol{c}-\boldsymbol{Dw}\|_2^2+\boldsymbol{y}^T(\boldsymbol{c}-\boldsymbol{Dw}). \tag{14}$$

It can be seen that the objective function in (14) is a quadratic function of $\boldsymbol{w}$. Hence, the unique solution exists and can be obtained by computing the derivative of (14) and letting the result be zero. Specifically, the optimal $\boldsymbol{w}$ is calculated by

$$\boldsymbol{w}=(\boldsymbol{I}+\boldsymbol{D}^T\boldsymbol{D})^{-1}\left[\boldsymbol{D}^T\left(\boldsymbol{c}+\frac{\boldsymbol{y}}{\mu}\right)+\boldsymbol{b}-\frac{\boldsymbol{x}}{\mu}\right]. \tag{15}$$

4) x,y-Step: The goal is to seek the optimal $\boldsymbol{x}$ and $\boldsymbol{y}$ when $\boldsymbol{b}$, $\boldsymbol{U}$, $\boldsymbol{v}$, $\boldsymbol{c}$, and $\boldsymbol{w}$ are fixed. Computing the derivative of the sum of the terms of (7) with respect to $\boldsymbol{x}$ and $\boldsymbol{y}$ and then letting the result be zero yield the following updating rule:

$$\boldsymbol{x} \leftarrow \boldsymbol{x} + \mu(\boldsymbol{w} - \boldsymbol{b}) \tag{16}$$

$$\boldsymbol{y} \leftarrow \boldsymbol{y} + \mu(\boldsymbol{c} - \boldsymbol{D}\boldsymbol{w}). \tag{17}$$

It is noted that the coefficient μ is updated by

$$\mu \leftarrow a\mu \tag{18}$$

where a is a parameter and its empirical value is 1.25.

5) U-Step: The goal is to seek the optimal $\boldsymbol{U}$ when $\boldsymbol{b}$, $\boldsymbol{v}$, $\boldsymbol{c}$, $\boldsymbol{w}$, $\boldsymbol{x}$, and $\boldsymbol{y}$ are fixed. The problem of minimizing (7) with respect to $\boldsymbol{U}$ becomes

$$\boldsymbol{U} = \arg\min_{\boldsymbol{U}} \sum_i \frac{1}{2} b_i (\boldsymbol{U}_i \boldsymbol{v} - \boldsymbol{o}_i)^2, \quad \text{s.t. } \boldsymbol{U}\boldsymbol{U}^T = \boldsymbol{I} \tag{19}$$

where $\boldsymbol{I}$ is the identity matrix. It is known that orthogonal matrices representing linear subspaces of the Euclidean space can be represented as points on the Grassmann manifolds [45]. Therefore, subspace estimation can be equivalently formulated as an optimization problem on the Grassmann manifolds [45]. Defining

$$L_f \triangleq \frac{1}{2} \boldsymbol{X}_{\boldsymbol{b}} (\boldsymbol{U}\boldsymbol{v} - \boldsymbol{o}) \boldsymbol{v}^T \boldsymbol{U} = \sum_i \frac{1}{2} b_i (\boldsymbol{U}_i \boldsymbol{v} - \boldsymbol{o}_i)^2 \tag{20}$$

the optimization can be performed by using the gradient $\partial L_f / \partial \boldsymbol{U}$ on the Euclidean space and the gradient ∇L_f of the Grassmannian [46]. In (20), $\boldsymbol{X}_{\boldsymbol{b}} \in \mathbb{R}^{N \times N}$, which is a diagonal matrix generated by $\boldsymbol{b}$. The gradient of L_f is given by

$$\frac{\partial L_f}{\partial \boldsymbol{U}} = \boldsymbol{X}_{\boldsymbol{b}} (\boldsymbol{U}\boldsymbol{v} - \boldsymbol{o}) \boldsymbol{v}^T \tag{21}$$

and

$$\begin{aligned} \nabla L_f &= (\boldsymbol{I} - \boldsymbol{U}\boldsymbol{U}^T) \frac{\partial L_f}{\partial \boldsymbol{U}} \\ &= (\boldsymbol{I} - \boldsymbol{U}\boldsymbol{U}^T) \boldsymbol{X}_{\boldsymbol{b}} (\boldsymbol{U}\boldsymbol{v} - \boldsymbol{o}) \boldsymbol{v}^T \\ &= (\boldsymbol{I} - \boldsymbol{U}\boldsymbol{U}^T) \boldsymbol{r} \boldsymbol{v}^T \end{aligned} \tag{22}$$

where the residual vector $\boldsymbol{r}$ is defined as

$$\boldsymbol{r} \triangleq \boldsymbol{X}_{\boldsymbol{b}} (\boldsymbol{U}\boldsymbol{v} - \boldsymbol{o}. \tag{23}$$

The solution on the Grassmannian manifolds is [15], [19]

$$\begin{aligned} \boldsymbol{U}\text{arrow} = \boldsymbol{U} &+ (\cos(\sigma\eta) - 1) \boldsymbol{U} \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|} \frac{\boldsymbol{v}^T}{\|\boldsymbol{v}\|} \\ &- \sin(\sigma\eta) \frac{\boldsymbol{r}}{\|\boldsymbol{r}\|} \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|}. \end{aligned} \tag{24}$$

6) v-Step: The low-dimensional representation ($\boldsymbol{v}$) of $\boldsymbol{o}$ can be simply calculated by

$$\boldsymbol{v} = \boldsymbol{U}^T \boldsymbol{o}. \tag{25}$$

Algorithm 1 summarizes the above steps. The initialization of the parameters α, β, μ, λ, and τ can be found from Table III, which is to be described in Section IV-B. The matrix U can be initialized by the result of any subspace method such as singular value decomposition (SVD). In our experiments, SVD is used for initializing the matrix $\boldsymbol{U}$. Specifically, we choose the first m ($m = 50$ in our experiments) frames to form a matrix $\boldsymbol{J} \in \mathbb{R}^{n \times m}$, where n is the number of pixels in an image (frame). The matrix $\boldsymbol{U}$ can be initialized by the singular vectors of SVD of the matrix $\boldsymbol{J}$ corresponding to the first few (five in our experiments) large singular values. The initial values of the vectors $\boldsymbol{b}$, $\boldsymbol{c}$, $\boldsymbol{x}$, and $\boldsymbol{y}$ can be zero or any random number. In our experiments, zero is used for initializing the vectors. A good initialization of the elements of $\boldsymbol{w}$ is 1.

Algorithm 1 Proposed Method of Moving Object Detection

Input:

A sequence of frames (images) from $\boldsymbol{o}_s$ to $\boldsymbol{o}_t$ and the current image is $\boldsymbol{o}$. Each image has N pixels.

Output:

Foreground indicator vector $\boldsymbol{f}$ corresponding to the current image $\boldsymbol{o}$.

Initialization
Initialize parameters α, β, μ, λ, and τ.
Initialize $\boldsymbol{U}$, $\boldsymbol{b}$, $\boldsymbol{c}$, $\boldsymbol{w}$, $\boldsymbol{x}$, and $\boldsymbol{y}$.
for $\boldsymbol{o} = \boldsymbol{o}_s$ to $\boldsymbol{o}_t$ **do**
  Applying some salient object detection algorithm on $\boldsymbol{o}$ and get the corresponding saliency map $\boldsymbol{s}$.
  $\boldsymbol{v} = \boldsymbol{U}^T \boldsymbol{o}$
  Iteration
    $\hat{\boldsymbol{b}} = \boldsymbol{b}$, $\hat{\boldsymbol{x}} = \boldsymbol{x}$, $\hat{\boldsymbol{y}} = \boldsymbol{y}$, $\hat{\boldsymbol{U}} = \boldsymbol{U}$
    $\boldsymbol{b} - Step$:$b_i = \frac{\beta + \mu w_i + x_i - \frac{1}{2}(\boldsymbol{U}_i \boldsymbol{v} - \boldsymbol{o}_i)^2 + \alpha(1 - s_i)}{\mu}$
    $\boldsymbol{c} - Step$:$\boldsymbol{c} = S_{\frac{\lambda}{\mu}}(\boldsymbol{D}\boldsymbol{w} - \frac{\boldsymbol{y}}{\mu})$
    $\boldsymbol{w} - Step$:$\boldsymbol{w} = (\boldsymbol{I} + \boldsymbol{D}^T\boldsymbol{D})^{-1} \left[\boldsymbol{D}^T \left(\boldsymbol{c} + \frac{\boldsymbol{y}}{\mu} \right) + \boldsymbol{b} - \frac{\boldsymbol{x}}{\mu} \right]$
    $\boldsymbol{x}, \boldsymbol{y} - Step$:Assign a small number 0.1 to μ. Update $\boldsymbol{x}$ and $\boldsymbol{y}$ by running the following formulas: $\boldsymbol{x} \leftarrow \hat{\boldsymbol{x}} + \mu(\boldsymbol{w} - \boldsymbol{b})$, $\boldsymbol{y} \leftarrow \hat{\boldsymbol{y}} + \mu(\boldsymbol{c} - \boldsymbol{D}\boldsymbol{w})$, $\mu \leftarrow 1.25\mu$.
  until $||\boldsymbol{b} - \hat{\boldsymbol{b}}||_2 < \tau ||\hat{\boldsymbol{b}}||_2$
  $\boldsymbol{U} - Step$: Assign a small number 5×10^{-3} to η [47]. Update $\boldsymbol{U}$ by running the following formulas: $\boldsymbol{r} = \boldsymbol{X}_{\boldsymbol{b}}(\boldsymbol{U}\boldsymbol{v} - \boldsymbol{o}, \boldsymbol{U} \leftarrow = \hat{\boldsymbol{U}} + (\cos(\sigma\eta) - 1)\hat{\boldsymbol{U}} \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|} \frac{\boldsymbol{v}^T}{\|\boldsymbol{v}\|} - \sin(\sigma\eta) \frac{\boldsymbol{r}}{\|\boldsymbol{r}\|} \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|}$.
  Compute foreground indicator vector, $\boldsymbol{f}$ is obtained by binarizing background vector:
  $f_i = \begin{cases} 0 \text{ if } b_i \geq t, \\ 1 \text{ if } b_i < t. \end{cases}$
end for

IV. Experimental Results

We describe intermediate results followed by a comparison with state-of-the-art methods on the Perception Test Images Sequences data set [29] and the Wallflower data set [50].

Though there are a lot of methods for obtaining saliency map, in our experiments, the saliency maps are obtained by the method developed in [43]. There are several reasons for us to choose the method in [43].

1) The method is accurate because of elegantly formulating the problem of salient object detection. Instead of heuristically integrating multiple low-level cues for salient

TABLE I
METHOD USED FOR INTERMEDIATE RESULTS

Method	Objective Function
BAL	$L_b = \min_{\mathbf{b},\mathbf{U},\mathbf{v}} \sum_{i=1}^{N} \left[\frac{1}{2} b_i (\mathbf{U}_i \mathbf{v} - o_i)^2 + \beta(1 - b_i)\right]$
ACO	$L_c = \min_{\mathbf{b},\mathbf{U},\mathbf{v}} \sum_{i=1}^{N} \left[\frac{1}{2} b_i (\mathbf{U}_i \mathbf{v} - o_i)^2 + \beta(1 - b_i)\right] + \lambda \|\mathbf{Db}\|_1$
ASMO	$L_s = \min_{\mathbf{b},\mathbf{U},\mathbf{v}} \sum_{i=1}^{N} \left[\frac{1}{2} b_i (\mathbf{U}_i \mathbf{v} - o_i)^2 + \beta(1 - b_i) - \alpha b_i (1 - s_i)\right]$
MODSM	$L = \min_{\mathbf{b},\mathbf{U},\mathbf{v}} \sum_{i=1}^{N} \left[\frac{1}{2} b_i (\mathbf{U}_i \mathbf{v} - o_i)^2 + \beta(1 - b_i) - \alpha b_i (1 - s_i)\right] + \lambda \|\mathbf{Db}\|_1$

object detection, the cost function of the method [43] is defined to directly and optimally achieve the goal of salient object detection.

2) The method is efficient because all the constraints are in linear form and thus the optimal saliency map can be solved by an efficient least-square optimization.
3) We have evaluated several methods of saliency map and we found that this method is effective indeed for the task of moving object detection in the proposed framework.

A. Intermediate Results

We give intermediate results to show the role of the saliency map term $-b_i(1 - s_i)$ and the connectivity term $\|\boldsymbol{Db}\|_1$.

For notation simplicity, in Table I, we list the objective functions of four methods (configurations) baseline (BAL), add connectivity only (ACO), add saliency map only (ASMO), and our MODSM method. The objective function of our method is the weighted sum of the objective functions of BAL, ACO, and ASMO. In our method, sparsity, low rank, connectivity, and saliency map are taken into account. Specifically, the objective function of the proposed method is

$$L = \min_{\boldsymbol{b},\boldsymbol{U},\boldsymbol{v}} \sum_{i=1}^{N} \left[\frac{1}{2} b_i (\boldsymbol{U}_i \boldsymbol{v} - \boldsymbol{o}_i)^2 + \beta(1 - b_i) - \alpha b_i (1 - s_i)\right] + \lambda \|\boldsymbol{Db}\|_1. \quad (26)$$

BAL is the method whose objective function L_b (27) consists of the first two terms of L (26)

$$L_b = \min_{\boldsymbol{b},\boldsymbol{U},\boldsymbol{v}} \sum_{i=1}^{N} \left[\frac{1}{2} b_i (\boldsymbol{U}_i \boldsymbol{v} - \boldsymbol{o}_i)^2 + \beta(1 - b_i)\right]. \quad (27)$$

In addition to the reconstruction term, the BAL method merely makes use of the sparsity term $\beta(1-b_i)$. The objective function of the BAL is similar to but slightly different from that of GRASTA [15].

Compared with L_b (27), the objective function L_c (28) of ACO has an additional connectivity term $\lambda\|\boldsymbol{Db}\|_1$

$$L_c = \min_{\boldsymbol{b},\boldsymbol{U},\boldsymbol{v}} \sum_{i=1}^{N} \left[\frac{1}{2} b_i (\boldsymbol{U}_i \boldsymbol{v} - \boldsymbol{o}_i)^2 + \beta(1 - b_i)\right] + \lambda \|\boldsymbol{Db}\|_1. \quad (28)$$

The objective function of BAL is similar to but different from that of ISC [9]. See [9] for the difference.

TABLE II
F1-SCORES OF THE BAL METHOD AND THE METHODS THAT ADD DIFFERENT TERMS (SEE TABLE I). THE PROPOSED METHOD IS THE METHOD OF ADDING BOTH CONNECTIVITY AND SALIENCY MAP TERMS TO BAL

Method	*WS*	*Cur*	*Fou*	*Hal*	*SM*	*Lob*	*Esc*	*BS*	*Cam*	*mean*
BAL	.8717	.8477	.2952	.6695	.6978	.5607	.3864	.6531	.1281	.5678
ACO	.8835	.8942	.8230	.6833	.6912	.4609	.7410	.6799	.5500	.7118
ASMO	.8907	.8112	.6220	.6583	.6765	.4812	.5521	.6844	.4424	.6466
MODSM	.9404	.9098	.8205	.6859	.7362	.5762	.7553	.7280	.7876	.7711

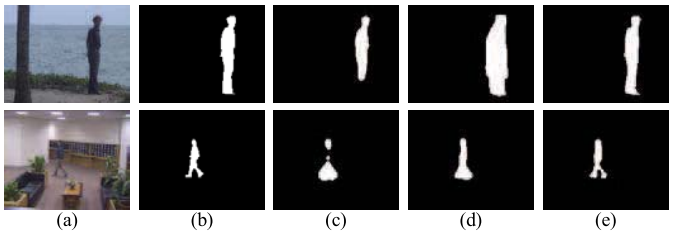


Fig. 2. Two examples of the influence of adding connectivity and saliency map to the objective function. (a) Input image. (b) Ground truth. (c) BAL. (d) ACO. (e) MODSM.

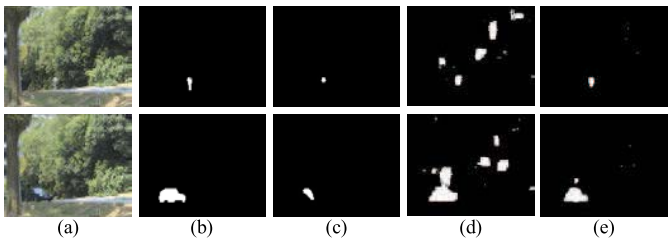


Fig. 3. Other two examples of the influence of adding connectivity and saliency map to the objective function. (a) Input image. (b) Ground truth. (c) BAL. (d) ACO. (e) MODSM.

The objective function L_s of ASMO is

$$L_s = \min_{\boldsymbol{b},\boldsymbol{U},\boldsymbol{v}} \sum_{i=1}^{N} \left[\frac{1}{2} b_i (\boldsymbol{U}_i \boldsymbol{v} - \boldsymbol{o}_i)^2 + \beta(1 - b_i) - \alpha b_i (1 - s_i)\right]. \quad (29)$$

One can see from Table II the contribution of different terms. The average F1-scores of BAL, ACO, ASMO, and our MODSM are 56.78%, 71.18%, 64.66%, and 77.11%, respectively. Both ACO and ASMO are capable of improving the detection result of BAL, showing the importance of independently introducing the connectivity term and the saliency map term. Adding the saliency map term gives 64.66%−56.78% = 7.88% improvement over BAL. By incorporating both the saliency map term and the connectivity term, the proposed method gives 77.11%−56.78% = 20.33% improvement over BAL.

Several frames of Perception Test Image Sequences [29] are used for analyzing the intermediate results. Some examples are shown in Figs. 2 and 3. Fig. 2(a) shows two input frames with water surface background for the top one and indoor environment for the bottom one. The ground truths of the moving objects are given in Fig. 2(b). Fig. 2(c) shows the detected results of BAL from which one can see that the

Fig. 4. (b) and (d) Saliency maps of (a) and (c), respectively.

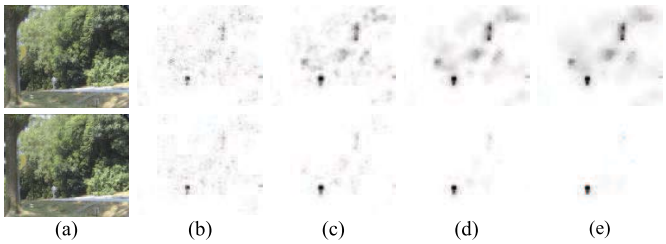


Fig. 5. Input image and the background vector obtained in different iterations of ACO (top) and MODSM (i.e., adding both saliency map and connectivity to BAL) (bottom). (a) Input image. (b) iteration #1. (c) iteration #3. (d) iteration #6. (e) iteration #10.

detected object is smaller than the ground truth. The top of Fig. 2(c) shows that the feet and some portions of the shanks are missed by the BAL method. The bottom of Fig. 2(c) shows that the middle of the person is missed by the BAL method.

As can be seen from Fig. 2(d), with the help of connectivity term, the ACO is able to detect the missed parts [see the feet and legs on the top of Fig. 2(c) and the middle part of the person on the bottom of Fig. 2(c)] of the persons. However, one can also see that there are many false alarms in Fig. 2(d). False alarms are the by-product of add connectivity. Fig. 2(e) is the result of the proposed method (i.e., by adding both saliency map and connectivity to BAL). Obviously, introducing the saliency map successfully discards the false alarms existing in Fig. 2(d).

The results given in Fig. 3(e) demonstrate that adding saliency map into the objective function is capable of suppressing many false alarms when the size of moving objects [a person on the top of Fig. 3(a) and a car on the bottom of Fig. 3(a)] is small, whereas the background is large, complex, and dynamic. Fig. 3(d) shows that adding connectivity into the objective function not only enlarges the area of the objects detected by BAL but also incorrectly classifies moving leafs and shadows as semantic objects. Adding saliency map [Fig. 3(e)] plays a role of overcoming the drawbacks of adding connectivity.

The saliency maps of the top and bottom of Fig. 3(a) are shown in Fig. 4(b) and (d), respectively. Though the saliency maps are not ideal, they provide a useful clue for the proposed method (i.e., by adding both saliency map and connectivity to BAL).

The proposed algorithm (i.e., adding both saliency map and connectivity to BAL) (see Algorithm 1) and the ACO algorithm update the background vector $\boldsymbol{b}$ iteratively. Figs. 5 and 6 show how the background vector $\boldsymbol{b}$ varies with iterations. Fig. 5(a) shows the input image identical to the top of Fig. 3(a). The top and bottom of Fig. 5 correspond to the iteration results

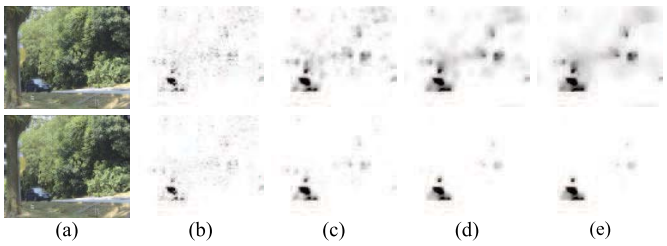


Fig. 6. Input image and the background vector b obtained in different iterations of ACO (top) and MODSM (i.e., adding both saliency map and connectivity to BAL) (bottom). (a) Input image. (b) iteration #1. (c) iteration #3. (d) iteration #6. (e) iteration #10.

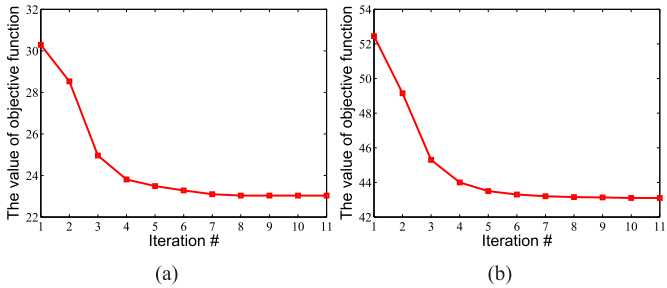


Fig. 7. Convergence of the proposed method (i.e., adding both saliency map and connectivity to the BAL). (a) For the input image shown in Fig. 5(a). (b) For the input image shown in Fig. 6(a).

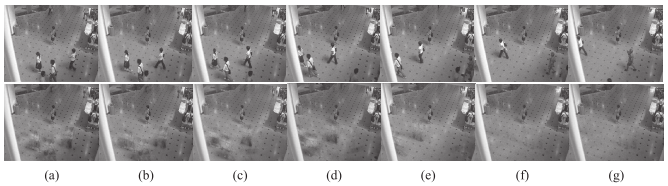


Fig. 8. Reconstructed background updates from the first frame to the 34th frame. (a) $t = 1$. (b) $t = 4$. (c) $t = 9$. (d) $t = 15$. (e) $t = 20$. (f) $t = 27$. (g) $t = 34$.

of the ACO algorithm and the proposed MODSM algorithm, respectively. One can see from the top of Fig. 5 that the background vector obtained by ACO contains more regions of waving leafs as the iteration proceeds. However, one can see from the bottom of Fig. 5 that the background vector obtained by the proposed method excludes more regions of waving leafs as the iteration proceeds and hence the foreground vector focuses on the true meaningful moving person.

Similar to Fig. 5, the bottom of Fig. 6 also demonstrates that adding the saliency map into the objective function makes the estimated background vector iteratively exclude the influence of moving leafs.

Fig. 7 shows the convergence property of the proposed algorithm. Generally, the value of the objective function L decreases drastically at the first five iterations and becomes stable after iteration # 8. To further show the convergence of the background of the first few frames, the bottom of Fig. 8 shows the reconstructed backgrounds corresponding to the first, fourth, ninth, 15th, 20th, 27th, and 34th frame of the *Shopping Mall* (*SM*) video. We can find that the reconstructed background improves gradually as more frames are available.

TABLE III
PARAMETERS OF THE MODSM METHOD

m	β	λ	α	μ	τ
5	$\beta = \max(\frac{1}{2}\beta, 4.5\hat{\sigma}^2)$	5β	$\min\left(\left\lceil \frac{s_m}{s_m - s_M} \right\rceil \hat{\sigma} s_m, 6.5\beta\right)$	0.1	10^{-6}

For example, the reconstructed background corresponding to the first frame [see the bottom of Fig. 8(a)] contains obvious artifacts caused by moving objects. However, when the model updates to 20th frame or 34th frame, the reconstructed backgrounds [see the bottom of Fig. 8(e) and (g)] are mainly dominated by true backgrounds and the influence of the moving objects is negligible.

B. Comparison With the State-of-the-Art Methods on Perception Test Images Sequences

The Perception Test Images Sequences data set [29] is also used for comparison with the state-of-the-art methods. The data set consists of nine videos captured in a variety of indoor and outdoor environments, including offices, campuses, sidewalks, and other private and public sites.

The weather conditions when collecting the data cover sunny, cloudy, and rainy weather. The videos with static background are named *Bootstrap* (*BS*), *SM*, and *Hall* (*Hal*). The videos with dynamic background are called *Fountain* (*Fou*), *Escalator* (*Esc*), *Water Surface* (*WS*), *Curtain* (*Cur*), and *Campus* (*Cam*). The *Lobby* (*Lob*) video is captured when there are drastic variations in illumination. The sizes (widths and heights) of the frames include [160, 130], [160, 128], [176, 144], [160, 120], [160, 128], and [320, 256].

We compare the proposed MODSM algorithm with DECOLOR [8], ISC [9], RFDSA [13], Dirichlet process Gaussian mixture model (DP-GMM) [14], GRASTA [15], PCP [33], GMM [48], and SOBS [49]. PCP, GRASTA, and RFDSA are the state-of-the-art subspace-based algorithms. DP-GMM is the state-of-the-art density-based algorithm and DECOLOR is the state-of-the-art low-rank-based algorithm. DP-GMM and GRASTA are incremental algorithms, whereas PCP, DECOLOR, and RFDSA are batch algorithms. As stated in Section II, ISC can be considered as an incremental version of DECOLOR. Note that GRASTA randomly samples a fraction of pixels in an image for subspace modeling and object detecting. Its detection accuracy increases with the fraction. To reduce randomness and get its best accuracy, 100% pixels are used in our experiments.

The parameters (7) of the MODSM method are given in Table III, where m is the number of columns (basis vectors) of the matrix $\boldsymbol{U}$. The parameter β balances the sparsity term and other terms. In the training stage, β is updated frame by frame according to $\beta = \max(\frac{1}{2}\beta, 4.5\hat{\sigma}^2)$. For the first frame, β is set to be the variance of the first frame. The schemes of setting β and λ are the same as those in [8]. In Table III, $\hat{\sigma}^2$ is given by

$$\hat{\sigma}^2 = \frac{1}{2|\Omega|} \sum_{\boldsymbol{O} \in \Omega} ||\boldsymbol{U}\boldsymbol{v} - \boldsymbol{o}||_2^2 \tag{30}$$

TABLE IV
F1-SCORES OF DIFFERENT METHODS ON PERCEPTION TEST IMAGES SEQUENCES

Method	*WS*	*Cur*	*Fou*	*Hal*	*SM*	*Lob*	*Esc*	*BS*	*Cam*	*mean*
GMM	.7948	.7580	.6854	.3335	.5363	.6519	.1388	.3838	.0757	.4842
SOBS	.8247	.8178	.6554	.5943	.6677	.6489	.5770	.6019	.6960	.6760
DP-GMM	.9090	.8203	.7049	.5484	.6522	.5794	.5055	.6024	.7567	.6754
PCP	.4137	.6193	.5679	.5917	.7234	.6989	.6728	.6582	.3406	.5874
DECOLOR	.8866	.8255	**.8598**	.6424	.6525	.6149	.6994	.5869	**.8096**	.7308
GRASTRA	.7310	.6591	.3786	.5817	.7142	.5550	.4697	.6146	.2504	.5505
ISC	.7176	.2919	.7112	.6560	**.7487**	.5715	.6751	.6787	.2897	.5933
RFDSA	.8796	.8976	.7544	.6673	7407	**.8029**	.6353	.6841	.6779	.7489
MODSM	**.9404**	**.9098**	.8205	**.6859**	.7362	.5762	**.7553**	**.7280**	.7876	**.7711**

where Ω and $|\Omega|$ are the set and the number of training images, respectively. s_m is the ratio of the number of pixels whose saliency is larger than the mean of the saliency maps of training images

$$s_m = \frac{\sum_S \sum_{i=1}^{N} I(s_i - s_M)}{N|\Omega|} \tag{31}$$

with

$$s_M = \frac{\sum_S \sum_{i=1}^{N} s_i}{N|\Omega|} \tag{32}$$

and

$$I(x) = \begin{cases} 1 & x > 0 \\ 0 & x \leq 0. \end{cases} \tag{33}$$

Note that the $\lceil x \rceil$ in Table III stands for the floor function of x.

We have evaluated many formulas for setting α and the experimental results show that the formula in Table III is the best.

Table III gives a general rule for parameter setting. However, the detection performance can be significantly improved if video-specific parameters are utilized.

The F_1-score, the harmonic mean of precision and recall, is used for objective evaluation

$$F_1 = 2\frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}. \tag{34}$$

The F_1-score results of the different methods are given in Table IV. Among the nine videos, the proposed MODSM, RFDSA, DECOLOR, and ISC get the best performance on five (i.e., *WS*, *Cur*, *Hal*, *Esc*, and *BS*), one (i.e., *Lob*), two (i.e., *Fou* and *Cam*), and one (i.e., *SM*) different videos, respectively. The average F_1-score of the proposed MODSM is the largest. However, our method does not work well for the *Lob* video. The main reason is that the performance of the method [43] of creating saliency map on the *Lob* video degraded significantly. If the *Lob* video was excluded, the average F_1-score of MODSM grows from 0.7711 to 0.7955, whereas that of RFDSA decreases from 0.7489 to 0.7421. It is expected that the performance of MODSM increases with the performance of saliency map. Table IV also shows that if proper prior information (i.e., connectivity, saliency map, and sparsity) is employed, then the incremental algorithm MODSM can outperform the batch algorithms DECOLOR and RFDSA.

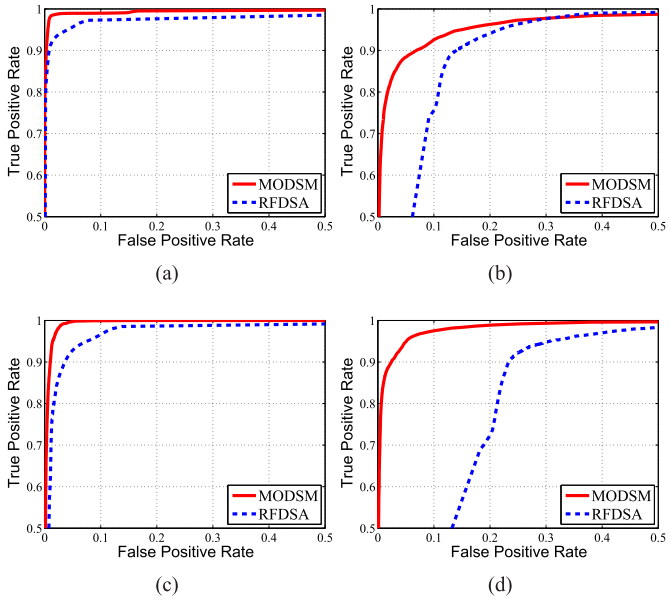


Fig. 9. ROC curves on the *WS*, *Esc*, *Fou*, and *Cam* videos.

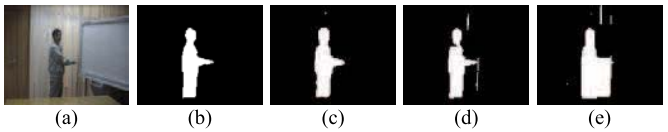


Fig. 10. Detected objects for a frame of the *Cur* video. (a) Input image. (b) Ground truth. (c) MODSM. (d) RFDSA. (e) DECOLOR.

Moreover, the proposed MODSM is remarkably superior to the incremental algorithm ISC. The average F1-score of ISC [9] is 59.33%, whereas the average F1-score of our method is as large as 77.11%. The proposed MODSM method gives an average of 17.78% over the ISC method. The benefit of our method comes not only from introducing saliency map into a unified objective function but also from employing continuous foreground vector instead of a binary one.

The ROC curves of the MODSM and RFDSA on the *WS*, *Esc*, *Fou*, and *Cam* videos are shown in Fig. 9, where the superiority of the MODSM can be observed. Take the *Fou* video as an example. The true positive rates (i.e., recall) of MODSM and RFDSA are, respectively, 0.99 and 0.935 when the false positive rate is 0.05. Note that the DOCOLOR and ISC methods cannot generate the ROC curves because of their binary values of the estimated foreground and background.

Several specific results of MODSM, RFDSA, and DECOLOR are visualized in Figs. 10–13(a)–(e) as the current input frame, ground truth of the moving objects, and the detected results of MODSM, RFDSA, and DECOLOR, respectively.

Fig. 10(a) shows a frame of the *Cur* video. Fig. 10(d) shows that RFDSA incorrectly regards the variation caused by motion of the curtain as moving objects and RFDSA results in incomplete neck of the person. Fig. 10(e) shows that DECOLOR gives rise to even more false alarms. Investigating Fig. 10(b) and (c), one can find that the result of MODSM is much closer to that of the ground truth.

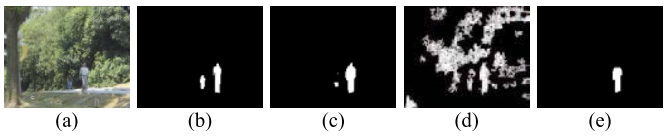


Fig. 11. Detected objects for a frame of the *Cam* video. (a) Input image. (b) Ground truth. (c) MODSM. (d) RFDSA. (e) DECOLOR.

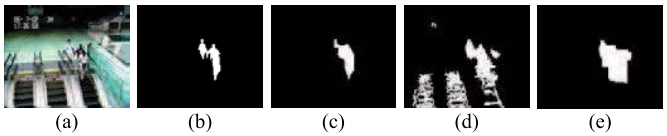


Fig. 12. Detected objects for a frame of the *Esc* video. (a) Input image. (b) Ground truth. (c) MODSM. (d) RFDSA. (e) DECOLOR.

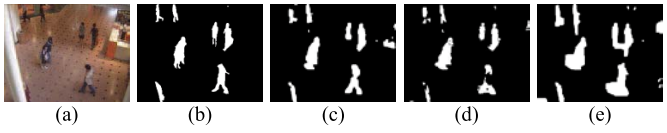


Fig. 13. Detected objects for a frame of the *SM* video. (a) Input image. (b) Ground truth. (c) MODSM. (d) RFDSA. (e) DECOLOR.

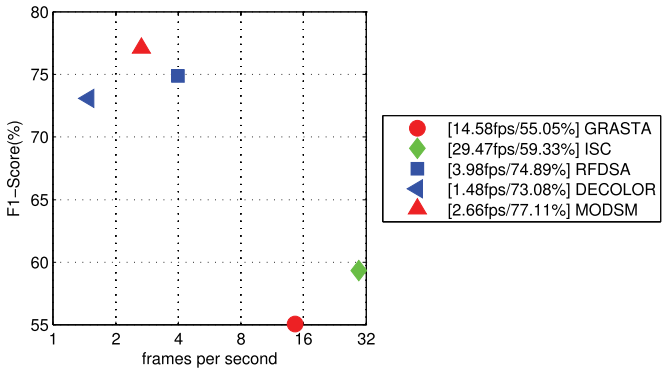


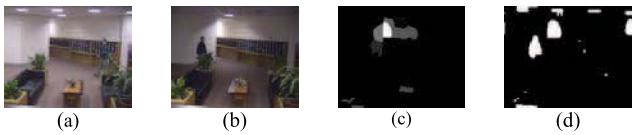
Fig. 14. Average F1-score versus the average FPSs on Perception Test Images Sequences.

Fig. 11(a) shows a frame of the *Cam* video. Fig. 11(d) shows that RFDSA incorrectly classifies many waving leafs as meaningful moving objects. Fig. 11(e) tells that DECOLOR cannot detect the left small person and the head of the right large person is also mistakenly classified as background. Fig. 11(c) shows that the proposed method is powerful for classifying the waving leafs as background and detecting both of the persons.

Fig. 12(a) shows a frame of the *Esc* video. Fig. 12(d) shows that RFDSA classifies the moving escalator as semantically meaningful moving objects. Because of using the information of saliency map, the proposed MODSM [Fig. 12(c)] avoids the errors of RFDSA. Fig. 12(e) shows that DECOLOR has almost not missed alarms but has many false alarms. The result [Fig. 12(c)] of MODSM is the best among the three methods.

Fig. 13(a) shows a frame of the *SM* video. It can be seen that MODSM is comparable and even slightly better than RFDSA and DECOLOR.

Fig. 14 shows the average F1-score versus average frames per second (FPSs) of GRASTA, DECOLOR, RFDSA, ISC, and the proposed MODSM method. Three conclusions can be drawn from Fig. 14.



(a) (b) (c) (d)

Fig. 15. Detected result for a frame of the *Lob* video. (a) Previous frame. (b) Current frame. (c) Saliency map. (d) Detected result.

TABLE V

F1-SCORES ON THE *WS* VIDEO AND THE *Cam* VIDEO VERSUS THE SALIENCY MAPS WITH DIFFERENT VARIANCES OF THE GAUSSIAN NOISE

Variance	*0*	*0.05*	*0.1*	*0.15*	*0.2*	*0.25*	*0.3*	*0.35*
WS	.9404	.9378	.9382	.9315	.9276	.9255	.9204	.9178
Cam	.7876	.7758	.7750	.7701	.7612	.7584	.7442	.7403

1) The proposed MODSM method is faster than DECOLOR and is comparable to RFDSA.
2) The proposed MODSM method is able to obtain the best F1-score.
3) The proposed MODSM method is capable of getting a good tradeoff between the F1-score and the computational speed.

Nevertheless, there is space to improve the efficiency of the proposed method. One way is to speed up the computation of the saliency map.

As can be seen from Table IV, the proposed method MODSM results in unsatisfying results on the *Lob* video. Fig. 15 attempts to explain the reasons. On the one hand, switching from light ON [Fig. 15(a)] to light OFF [Fig. 15(b)] gives rise to large variation, which is difficult for the basis vectors $\boldsymbol{U}$ to capture. On the other hand, the saliency map is not satisfying on the regions of the moving object (person). In this case, introducing the bad saliency map [Fig. 15(c)] has a negative influence on the task of moving object detection. The research progress of salient object detection is helpful for improving the performance of the propose method.

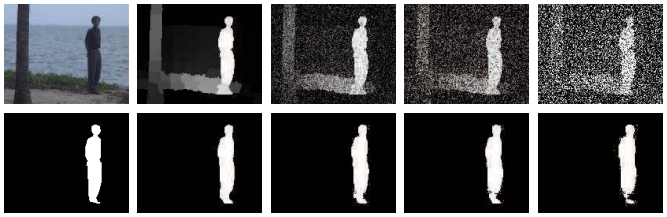
We also investigate how the quality of the saliency map influences the detection accuracy (F1-score). The Gaussian noise with different variances and the salt-and-pepper noise with different density are added into the saliency maps of the *WS* video and the *Cam* video, respectively. The detection results corresponding to the Gaussian noise are given in Table V and the detection results corresponding to the salt-and-pepper noise are given in Table VI. It is observed from Table V that the loss of F1-score is negligible when the variance of the Gaussian noise increases to 0.05, 0.1, and 0.15. The decrease in F1-score is relatively large only when the variance is very large. The phenomenon can also be seen from Table VI.

Fig. 16 visualizes the detection results when the Gaussian noise with variances 0, 0.05, 0.15, and 0.35 is added to the saliency map. As can be seen from Fig. 16, the person can be detected even when the saliency map contains significant noise. Of course, when the noise is too heavy that the saliency map is severely corrupted and is completely immersed in the noise, the detection accuracy will drop inevitably.

TABLE VI

F1-SCORES ON THE *WS* VIDEO AND THE *Cam* VIDEO VERSUS THE SALIENCY MAPS WITH DIFFERENT DENSITIES OF THE SALT-AND-PEPPER NOISE

Density	*0*	*0.05*	*0.1*	*0.15*	*0.2*	*0.25*	*0.3*	*0.35*
WS	.9404	.9392	.9379	.9341	.9323	.9267	.9223	.9145
Cam	.7876	.7832	.7746	.7705	.7647	.7502	.7462	.7389



(a) (b) (c) (d) (e)

Fig. 16. Detection result on the *WS* video when the Gaussian noise with different variances is added to the saliency map. (a) Input image. (b) var = 0. (c) var = 0.05. (d) var = 0.15. (e) var = 0.35.

TABLE VII

F1-SCORES OF DIFFERENT METHODS ON THE WALLFLOWER DATA SET

Method	*CF*	*FA*	*LS*	*BS*	*TOD*	*WT*	*mean*
DECOLOR	.4009	0	**.8199**	05869	**.8519**	.9402	.6004
RFDSA	.9162	.5800	.3897	.6841	.1605	.5090	.5399
GRASTRA	.2494	.3091	.2267	.6146	.1205	.4685	.3314
ISC	.6969	.4091	.2679	.6787	.1261	.4685	.4412
MODSM	**.9618**	**.6148**	**.7824**	.7280	.3966	**.9502**	.7389

C. Comparison With the State-of-the-Art Methods on the Wallflower Data Set

In this section, experimental results are given on the Wallflower data set [50]. The Wallflower data set consists of seven different test sequences. The scenarios of the data set are *Moved Object*, *Time of Day* (*TOD*), *Light Switch* (*LS*), *Waving Trees* (*WT*), *Camouflage* (*CF*), *BS*, and *Foreground Aperture* (*FA*).

The proposed method is compared with two representative incremental methods (i.e., ISC [9] and GRASTA [15]) and two representative batch methods (i.e., DECOLOR [8] and RFDSA [13]). The source codes of the four methods are publicly available.

The strategy of setting parameters for the proposed MODSM method is the same as Table III.

Table VII gives the F1-scores of DECOLOR, RFDSA, GRASTA, ISC, and the proposed MODSM method. The average scores of DECOLOR, RFDSA, GRASTA, ISC, and our method MODSM are 60.04%, 53.99%, 33.14%, 44.12%, and 73.89%, respectively. The proposed method is the best among the five methods. Specially, our method gives an average 13.85%, 19.90%, 40.75%, 29.77% improvement over DECOLOR, RFDSA, GRASTA, and ISC, respectively. Therefore, the superiority of the proposed method is very significant. The results demonstrate the important role of introducing saliency map into the objective function.

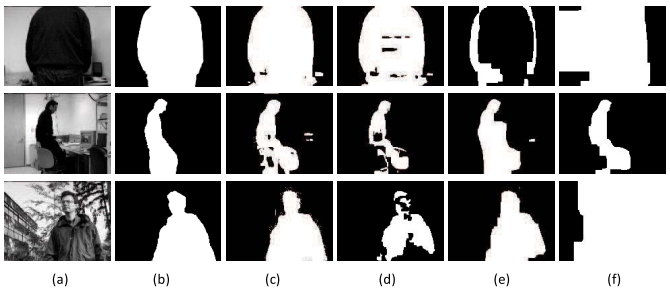


Fig. 17. Detection results for frames of the *CF* video, *LS* video, and *WT* video of the Wallflower data set. (a) Input image. (b) Ground truth. (c) MODSM. (d) RFDSA. (e) DECOLOR. (f) ISC.

In addition to the above quantitative comparison, we show in Fig. 17 several detection examples. The input images shown on the top, middle, and the bottom of Fig. 17(a) are sampled from the *CF* video, *LS* video, and *WT* video, respectively. The ground truth is shown in Fig. 17(b). Fig. 17(c)–(f) shows the detection results of MODSM, RFDSA, DECOLOR, and ISC, respectively. Among the five methods, the results of our MODSM are the best in the sense of approximating the ground truths.

The experimental results on both the Perception Test Images Sequences data set and the Wallflower data set demonstrate the effectiveness of the proposed method.

V. Conclusion

In this paper, we have presented a moving object detection method. The method makes use of a saliency map by incorporating it into a unified objective function for which the properties of sparsity, low rank, connectivity, and saliency are integrated. The manner of using a saliency map yields a smaller number of false alarms and missed alarms. Our future work will apply the idea of using a saliency map to other state-of-the-art incremental and batch methods of moving object detection. Moreover, we will investigate other state-of-the-art methods of generating a saliency map.

References

[1] J. Cao, Y. Pang, and X. Li, "Pedestrian detection inspired by appearance constancy and shape symmetry," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 1316–1324.

[2] J. Cao, Y. Pang, and X. Li, "Pedestrian detection inspired by appearance constancy and shape symmetry," *IEEE Trans. Image Process.*, vol. 25, no. 12, pp. 5538–5551, Dec. 2016.

[3] Y. Pang, H. Zhu, X. Li, and X. Li, "Classifying discriminative features for blur detection," *IEEE Trans. Cybern.*, vol. 46, no. 10, pp. 2220–2227, Oct. 2016.

[4] Y. Pang, H. Zhu, X. Li, and J. Pan, "Motion blur detection with an indicator function for surveillance machines," *IEEE Trans. Ind. Electron.*, vol. 63, no. 9, pp. 5592–5601, Sep. 2016.

[5] H. Li, H. Lu, Z. Lin, X. Shen, and B. Price, "Inner and inter label propagation: Salient object detection in the wild," *IEEE Trans. Image Process.*, vol. 24, no. 10, pp. 3176–3186, Oct. 2015.

[6] J. Lei *et al.*, "A universal framework for salient object detection," *IEEE Trans. Multimedia*, vol. 18, no. 9, pp. 1783–1795, Sep. 2016.

[7] W. Wang, J. Shen, X. Li, and F. Porikli, "Robust video object cosegmentation," *IEEE Trans. Image Process.*, vol. 24, no. 10, pp. 3137–3148, Oct. 2015.

[8] X. Zhou, C. Yang, and W. Yu, "Moving object detection by detecting contiguous outliers in the low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 3, pp. 597–610, Mar. 2013.

[9] J. Pan, X. Li, X. Li, and Y. Pang, "Incrementally detecting moving objects in video with sparsity and connectivity," *Cognit. Comput.*, vol. 8, no. 3, pp. 420–428, 2016.

[10] W. Wang, J. Shen, and L. Shao, "Consistent video saliency using local gradient flow optimization and global refinement," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4185–4196, Nov. 2015.

[11] H. Fu, D. Xu, B. Zhang, S. Lin, and R. K. Ward, "Object-based multiple foreground video co-segmentation via multi-state selection graph," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3415–3424, Nov. 2015.

[12] C. Li, X. Wang, L. Zhang, J. Tang, H. Wu, and L. Lin, "WELD: Weighted low-rank decomposition for robust grayscale-thermal foreground detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. PP, no. 99, p. 1, 2016, doi: 10.1109/TCSVT.2016.2556586.

[13] X. Guo, X. Wang, L. Yang, X. Cao, and Y. Ma, "Robust foreground detection using smoothness and arbitrariness constraints," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 535–550.

[14] T. S. F. Haines and T. Xiang, "Background subtraction with Dirichlet process mixture models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 4, pp. 670–683, Apr. 2014.

[15] J. He, L. Balzano, and A. Szlam, "Incremental gradient on the Grassmannian for online foreground and background separation in subsampled video," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1568–1575.

[16] X. Ye, J. Yang, X. Sun, K. Li, C. Hou, and Y. Wang, "Foreground–background separation from video clips via motion-assisted matrix restoration," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 11, pp. 1721–1734, Nov. 2015.

[17] M. Shakeri and H. Zhang, "COROLA: A sequential solution to moving object detection using low-rank approximation," *Comput. Vis. Image Understand.*, vol. 146, pp. 27–39, 2016.

[18] X. Zhou, C. Yang, H. Zhao, and W. Yu, "Low-rank modeling and its applications in image analysis," *ACM Comput. Surveys*, vol. 47, no. 2, 2015, Art. no. 36.

[19] J. Xu, V. K. Ithapu, L. Mukherjee, J. M. Rehg, and V. Singh, "GOSUS: Grassmannian online subspace updates with structured-sparsity," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 3376–3383.

[20] T. Bouwmans, A. Sobral, S. Javed, S. K. Jung, and E.-H. Zahzah, "Decomposition into low-rank plus additive matrices for background/foreground separation: A review for a comparative evaluation with a large-scale dataset," *Comput. Sci. Rev.*, to be published, doi: 10.1016/j.cosrev.2016.11.001.

[21] P. Dollár, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1532–1545, Aug. 2014.

[22] M. Isard and A. Blake, "CONDENSATION—Conditional density propagation for visual tracking," *Int. J. Comput. Vis.*, vol. 29, no. 1, pp. 5–28, 1998.

[23] Y. Pang, K. Zhang, Y. Yuan, and K. Wang, "Distributed object detection with linear SVMs," *IEEE Trans. Cybern.*, vol. 44, no. 11, pp. 2122–2133, Nov. 2014.

[24] Y. Lin, Y. Tong, Y. Cao, Y. Zhou, and S. Wang, "Visual-attention based background modeling for detecting infrequently moving objects," *IEEE Trans. Circuits Syst. Video Technol.*, vol. PP, no. 99, p. 1, 2016, doi: 10.1109/TCSVT.2016.2527258.

[25] A. Neri, S. Colonnese, G. Russo, and P. Talone, "Automatic moving object and background separation," *Signal Process.*, vol. 66, pp. 219–232, Apr. 1998.

[26] K. Wang, L. Xu, Y. Fang, and J. Li, "One-against-all frame differences based hand detection for human and mobile interaction," *Neurocomputing*, vol. 120, pp. 185–191, Nov. 2013.

[27] T. Huynh-The, O. Banos, S. Lee, B. H. Kang, E.-S. Kim, and T. Le-Tien, "NIC: A robust background extraction algorithm for foreground detection in dynamic scenes," *IEEE Trans. Circuits Syst. Video Technol.*, vol. PP, no. 99, p. 1, 2016, doi: 110.1109/TCSVT.2016.2543118.

[28] I. Haritaoglu, D. Harwood, and L. S. Davis, "W_4: Real-time surveillance of people and their activities," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 809–830, Aug. 2000.

[29] L. Li, W. Huang, I. Y.-H. Gu, and Q. Tian, "Statistical modeling of complex backgrounds for foreground object detection," *IEEE Trans. Image Process.*, vol. 13, no. 11, pp. 1459–1472, Nov. 2004.

[30] J. Lei, S. Li, C. Zhu, M.-T. Sun, and C. Hou, "Depth coding based on depth-texture motion and structure similarities," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 2, pp. 275–286, Feb. 2015.

[31] Y. Pang, S. Wang, and Y. Yuan, "Learning regularized LDA by clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 12, pp. 2191–2201, Dec. 2014.

[32] Y. Yuan, Y. Pang, J. Pan, and X. Li, "Scene segmentation based on IPCA for visual surveillance," *Neurocomputing*, vol. 72, nos. 10–12, pp. 2450–2454, 2009.
[33] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *J. ACM*, vol. 58, no. 3, 2011, Art. no. 11.
[34] P. Favaro, R. Vidal, and A. Ravichandran, "A closed form solution to robust subspace estimation and clustering," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 1801–1807.
[35] C. Qiu and N. Vaswani, "ReProCS: A missing link between recursive robust PCA and recursive sparse recovery in large but correlated noise," *CoRR*, vol. abs/1106.3286, 2011.
[36] F. De la Torre and M. J. Black, "A framework for robust subspace learning," *Int. J. Comput. Vis.*, vol. 54, no. 1, pp. 117–142, Aug. 2003.
[37] R. Vidal, Y. Ma, and S. Sastry, "Generalized principal component analysis (GPCA)," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 12, pp. 1945–1959, Dec. 2005.
[38] B. Xin, Y. Tian, Y. Wang, and W. Gao, "Background subtraction via generalized fused lasso foreground modeling," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 4676–4684.
[39] L. Balzano, R. Nowak, and B. Recht, "Online identification and tracking of subspaces from highly incomplete information," in *Proc. Allerton Conf. Commun.*, 2010, pp. 704–711.
[40] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.
[41] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222–1239, Nov. 2001.
[42] V. Kolmogorov and R. Zabin, "What energy functions can be minimized via graph cuts?" *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 147–159, Feb. 2004.
[43] W. Zhu, S. Liang, Y. Wei, and J. Sun, "Saliency optimization from robust background detection," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2814–2821.
[44] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.
[45] S. Mittal and P. Meer, "Conjugate gradient on Grassmann manifolds for robust subspace estimation," *Image Vis. Comput.*, vol. 30, nos. 6–7, pp. 417–427, 2012.
[46] A. Edelman, T. A. Arias, and S. T. Smith, "The geometry of algorithms with orthogonality constraints," *SIAM J. Matrix Anal. Appl.*, vol. 20, no. 2, pp. 303–353, 1998.
[47] D. Zhang and L. Balzano, "Global convergence of a Grassmannian gradient descent algorithm for subspace estimation," in *Proc. 19th Int. Conf. Artif. Intell. Statist.*, 2016, pp. 1–16.
[48] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 1999, p. 252.
[49] L. Maddalena and A. Petrosino, "A self-organizing approach to background subtraction for visual surveillance applications," *IEEE Trans. Image Process.*, vol. 17, no. 7, pp. 1168–1177, Jul. 2008.
[50] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers, "Wallflower: Principles and practice of background maintenance," in *Proc. IEEE Int. Conf. Comput. Vis.*, Sep. 1999, pp. 255–261.

Yanwei Pang (M'07–SM'09) received the Ph.D. degree in electronic engineering from University of Science and Technology of China, Hefei, China, in 2004.

He is a Professor with Tianjin University, Tianjin, China. His research interests include deep learning, object detection, pattern recognition, and image processing, in which he has authored more than 100 scientific papers including 27 IEEE TRANSACTION papers.

Li Ye received the B.S. degree in electronic engineering from Tianjin University, Tianjin, China, in 2014. She is currently working toward the M.S. degree at the same university.

Her research interests include object detection and image recognition.

Xuelong Li (M'02–SM'07–F'12) is a Full Professor with the Center for Optical Imagery Analysis and Learning, State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an, China.

Jing Pan received the B.S. degree in mechanical engineering from North China Institute of Technology (now North University of China), Taiyuan, China, in 2002 and the M.S. degree in precision instrument and mechanism from University of Science and Technology of China, Hefei, China, in 2007. She is currently working toward the Ph.D. degree with Tianjin University, Tianjin, China.

She is a Lecturer with the School of Electronic Engineering, Tianjin University of Technology and Education, Tianjin. Her research interests include computer vision and pattern recognition.