Spatiotemporal Low-rank Modeling for Complex Scene Background Initialization

Sajid Javed, Arif Mahmood, Thierry Bouwmans, Soon Ki Jung

Spatiotemporal Low-rank Modeling for Complex Scene Background Initialization

Sajid Javed, Arif Mahmood, Thierry Bouwmans, and Soon Ki Jung, *Member, IEEE,*

***Abstract*—Background modeling constitutes the building block of many computer-vision tasks. Traditional schemes model the background as a low rank matrix with corrupted entries. These schemes operate in batch mode and do not scale well with the data size. Moreover, without enforcing spatiotemporal information in the low-rank component, and because of occlusions by foreground objects and redundancy in video data, the design of a background initialization method robust against outliers is very challenging. To overcome these limitations, this paper presents a spatiotemporal low-rank modeling method on dynamic video clips for estimating the robust background model. The proposed method encodes spatiotemporal constraints by regularizing spectral graphs. Initially a motion-compensated binary matrix is generated using optical flow information to remove redundant data and to create a set of dynamic frames from the input video sequence. Then two graphs are constructed, one between frames for temporal consistency and the other between features for spatial consistency, to encode the local structure for continuously promoting the intrinsic behavior of the low-rank model against outliers. These two terms are then incorporated in the iterative matrix completion framework for improved segmentation of background. Rigorous evaluation on severely occluded and dynamic background sequences, demonstrates the superior performance of the proposed method over state-of-the-art approaches.**

***Index Terms*—Background modeling, Matrix completion, Robust Principal Component Analysis, Spatiotemporal graph regularizations.**

I. Introduction

BACKGROUND modeling and initialization is a major step in many image processing and computer vision applications, such as moving object detection [1], video surveillance [2], video segmentation [3], and video inpainting [4]. This preprocessing phase involves extraction of a good quality background image from a given input observation matrix or video sequence containing outliers and missing data. A plethora of algorithms have been proposed for background modeling and initialization [5]–[9]. Interesting surveys on the segregation of background-foreground can be perused in [2], [10]–[12]. Background modeling becomes challenging in the presence of dynamic scenes, variations in lighting conditions, and occlusions by foreground objects. Therefore, background modeling remains an interesting and unresolved field of research [11].

Subspace learning methods such as Robust Principal Component Analysis (RPCA) [13] and Matrix Completion (MC) [14] based on low-rank modeling have been investigated for dimensionality reduction in high-dimensional space and have attracted a considerable amount of attention over the past few years. MC methods [15] try to decompose input video sequence into an intrinsic structure known as low-rank component, which corresponds to the model of background, from partial observations of its entries. Fig. 1 shows an MC-based successful reconstruction of the static model of background given a sequence of input images of the *Hall & Monitor* video taken from the *Scene Background Modeling and Initialization* (SBMI) dataset [16]. The sparse component constitutes irregular behavior of foreground objects. It can also be computed from low-rank formulation, for example, by solving the convex optimization framework of Candés *et al.* [13], [14].

Unfortunately, most conventional MC and RPCA-based matrix decomposition algorithms present some prevalent challenges for background modeling [2], [15]. Most of these methods are based on batch processing. In order to model low-rank matrix, given a few entries of an input video sequence, a number of stacked training video frames must be stored in memory prior to data processing. Also all frames have to be accessed in each iteration of the optimization process. As a result, these methods require a large amount of memory and are computationally inefficient. In real-life background modeling, it is more effective to quickly estimate low-rank matrix incrementally when a new frame arrives rather than to follow a batch strategy.

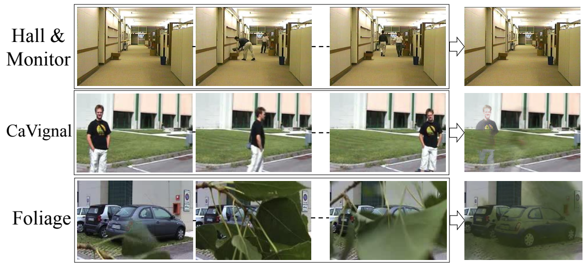
In many real world cases, the input video sequence consists of redundant data in the form of motionless frames. In which foreground objects remain static or move slowly within a specified time period. We observe that in these cases large number of outliers appear in the low-rank component. Superfluous data of this nature leads traditional MC approaches to poor performance. The *CaVignal* sequence in Fig. 1 clearly demonstrates the deficiencies of the recently proposed MC-based methods [1], [17], [18] for background model computation in the presence of superfluous data.

In some real world cases, clean background frames without any foreground object, are not available for training. In these cases either background and foreground coexist in each frame, or background image is heavily occluded by foreground objects. In such cases, background modeling becomes even more challenging. For example, the *Foliage* sequence in Fig. 1 shows the failure of existing MC methods [1], [19] to estimate background model because no prior knowledge about the pixels of background-foreground was available. Another pitfall of these methods is that they rely on the basic hypothesis

S. Javed and S. K. Jung are with Virtual Reality Laboratory, the School of Computer Science and Engineering, Kyungpook National University, 80 Daehak-ro, Buk-gu, Daegu, 41566, Republic of Korea. (e-mail: sajid@vr.knu.ac.kr, skjung@knu.ac.kr). (corresponding author: Soon Ki Jung)

A. Mahmood is with Department of Computer Science and Engineering, Qatar University, Qatar. (email: arif.mahmood@qu.edu.qa)

T. Bouwmans is with Universite de La Rochelle, Laboratoire Mathematiques, Image et Applications (MIA), 23 Avenue Albert Einstein, La Rochelle, BP 33060-17301, France. (email: tbouwman@univ-lr.fr)



(a) The set of input images [16] (b) Background model

Fig. 1. Typical example of the initialization of constant model background using state-of-the-art matrix completion frameworks [1], [19].

that background is static. This assumption is too restrictive in real-life scenarios. Many contemporary methods are unable to process situations in which background scene changes such as those caused by rippling water surfaces, swaying trees, and changing lighting conditions. For instance, the *fall* sequence taken from the change detection (CDnet14) dataset [20] in Fig. 2 shows failure of the background-foreground separation in a dynamic background scene.

Most existing RPCA [13], [19] and MC [1], [17] methods cannot efficiently handle these challenges. The study presented in this paper tries to overcome these limitations of the existing methods by proposing a robust *Spatiotemporal Low-rank Matrix Completion* (SLMC) algorithm which is based on multiple graph regularizations with dynamic frame extraction. The graphs are constructed with dynamic video frames that are computed from optical flow information. The proposed SLMC is an unsupervised algorithm for the estimation of background model and it efficiently handles the challenges highlighted in Fig. 1 in a single unified framework. It is assumed in SLMC that the images underlying background are linearly correlated and that the observation matrix composed of dynamic video frames can be approximated by the low-rank component through iterative matrix completion assisted by the optical flow information.

The proposed SLMC algorithm consists of three main stages: (i) detection of dynamic images to create an input dynamic sequence by discarding motionless video frames, (ii) computation of spatiotemporal graph Laplacians, and (iii) application of MC to incorporate the preceding two steps for the initialization of the background model which was occluded by foreground objects. First, a dense optical flow field is estimated between two consecutive frames and then the initial motion mask is generated, which facilitates the removal of superfluous data samples from the original sequence and creates a set comprising only the dynamic video clips. Second, spatiotemporal graph Laplacians are computed to encode the local similarity in dynamic sequence. Finally, iterative MC is applied on each column of the set of dynamic video clips with an initialized basis that guarantees fast convergence during the optimization procedure. Unlike existing approaches such as those investigated in [15], SLMC processes only one video frame from the set of dynamic sequence per time instance via stochastic optimization. The low-rank component estimated by the SLMC method is more robust and accurate than that obtained by previous MC approaches. It is because of the manifold information encoded in the graph Laplacian between the frames and pixels of the set of dynamic sequence.

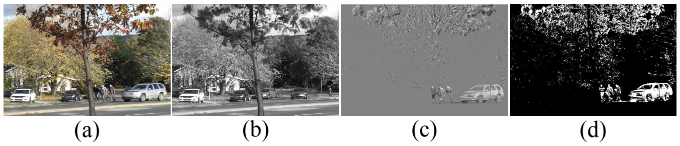


Fig. 2. Difficulties caused by dynamic background scenes using classical RPCA [13] for background-foreground separation. From left to right: (a) input, (b) low-rank component, (c) sparse component, and (d) mask of foreground objects, of *fall* sequence.

The proposed SLMC algorithm is then further extended to *Spatiotemporal Robust Principal Component Analysis* (SRPCA) with dynamic frames extraction. SRPCA efficiently separates the segments of foreground objects from dynamic background to overcome the challenges demonstrated by Fig. 2. In SLMC algorithm, a motion mask is used to label pixels obviously belonging to the foreground. While recovering background model, these pixels are considered as missing values or unobserved data. The observed background data still contains significant number of outliers which actually belong to the foreground. To effectively handle missing values and outliers, we propose the recovered background model to be smooth on the temporal as well as spatial manifolds. This has been ensured by incorporating a spatial and temporal graph regularization in the low-rank matrix completion objective function.

In contrast, the proposed SRPCA algorithm does not use optical flow based foreground pixel labeling process performed by the motion mask. However, the redundant data removal step is still used to remove motionless frames. The input to the SRPCA algorithm is the matrix containing a set of dynamic sequence, but there is no motion mask matrix input to SRPCA method. In SRPCA, both the background and the foreground components are optimized simultaneously while spatiotemporal smoothness constraint is only applied to the background model. The proposed SRPCA model may also be useful for extracting foreground objects from pre-recorded videos. However, because of the lack of computational resources it is very difficult to achieve a real-time processing for such task. Therefore, an important application of SRPCA method is offline video analysis for data mining purpose.

Finally, this study presents two types of very extensive qualitative analysis of the proposed SLMC and SRPCA algorithms: 1) For the task of reconstructing stationary model of background through SLMC on new challenging sequences of the SBMI dataset [16], and 2) detecting foreground objects from non-stationary scenes background through SRPCA on dynamic sequences of the I2R [21], Wallflower [22], and CDnet14 [20] datasets. In addition, for evaluating the quality of the static estimated background, a set of eight metrics described in [16] is used. A comparison of our method and earlier RPCA or MC and non-parametric learning methods is also presented to demonstrate the robustness of the proposed methods against outliers and the improvements in the modeling of background via graph regularization.

The rest of the paper is organized as follows. Related work is reviewed in Section II. The proposed method is described in Section III. The experimental results are discussed in Section IV. Finally, our conclusions and future research

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2016.2632302, IEEE Transactions on Circuits and Systems for Video Technology

directions are mentioned in Section V.

II. Related Work

In the past decade, excellent methods [1], [7], [8], [17] have been proposed to model the background image. These methods can be classified into several categories: such as subspace learning methods [1], [5], [6], [18], [19], a multiple features-based method [9], and non-parametric methods [7], [8]. More comprehensive and systematic reviews of all these methods were presented in [2], [10], [11], [23]. As the proposed method is based on subspace learning via the MC and RPCA framework we restrict the literature review to studies on RPCA and MC that process outliers with the integration of graph structure and motion information.

Oliver *et al.* [24] were the first to use PCA to model background using the eigenvalues of observation matrix. PCA provides a very robust subspace-learning model but is very sensitive to gross outliers. Several PCA enhancements are available in the literature [14], [25] that address the limitations of classical PCA with respect to outliers and noise, yielding the field of RPCA, also known as robust MC. For instance, the first study on RPCA and MC in [25], [26] concerned the recovery of low-rank matrix. Later, Candés *et al.* [13], [14] also used this recent notion of RPCA for the decomposition of input video sequence into its low-rank and sparse components. Under minimal assumptions, the technique, known as principal component pursuit, perfectly recovers the underlying low-dimensional subspace. Then, it was illustrated that this underlying subspace model can be exactly recovered if the number of observed entries is sufficiently large. Wang *et al.* [27] proposed a nonconvex relaxation method for MC tasks under contaminated non-Gaussian noise. This method is faster than the RPCA, since it includes a newly designed loss function and norm that can be solved using two optimization methods, called iterative soft and hard thresholding; however, it is limited to small-scale observation matrix. He *et al.* [19] also presented relatively stable online subspace tracking method named as GRASTA, which performed an iterative gradient descent on Grassmanian space for the recovery of the component of low-rank. These significant methods, both RPCA and MC, also provide a very elegant solution for the problem of initializing background. Interested readers can find more details in [15], [23].

A few studies such as those in [17], [18], recently considered prior knowledge of sparse observations (also known as motion information) for designing a stable MC framework for the estimation of a model of background. For example, Ye *et al.* [17] recently proposed the *Robust Motion-Assisted Matrix Restoration* (RMAMR) model for the segregation of background image from foreground objects. In this method, a dense motion field is incorporated in component low-rank and then mapped into a weighting matrix that indicates prior information about the pixels of background. Encouraging results are reported for many simple and dynamic scene of background. However, the method is based on a batch strategy and thus it is not suitable for large-scale data. Therefore, Javed *et al.* [28] recently proposed a new course to a fine iterative matrix decomposition framework with structural constraints, one on background and the second on foreground. New maximum norm (max-norm) constraints are applied on different superpixels, and as a result high performance is observed for the subtraction domain of background scene.

Manifold learning [29]–[31] has also been assimilated in these approaches [13], [14] for promoting the robustness and structure of the recovered subspace of low-rank. For example, in [29] a graph Laplacian PCA (GLPCA) was proposed in which the principal components are leveraged by the graph structure. However, this graph regularized term is incorporated into classical PCA, which is very sensitive against data corruption. Therefore, Shahid *et al.* [31] recently proposed RPCA on a graph, in which graph regularization is incorporated in the RPCA framework. Encouraging results were presented in the case of recovery of low-rank matrix against gross corruption.

The methods that use either RPCA or MC for modeling low-rank component are all based on the traditional batch processing strategy for designing a structured low-rank matrix. Therefore, computational efficiency is sacrificed. An online or incremental method has also been reported in the literature [19]. However, for a set of incremental video frames, when a new frame is added, the optimization procedure has to be re-implemented on all available frames in this method. This is quite inefficient when input sequence is high dimensional. Moreover, none of these methods have been shown to be sufficiently accurate to produce a model of low-rank because the spatial and temporal constraints of this model are not exploited; therefore, the estimated component of low-rank is very sensitive against outliers and noise.

In this work, we overcome these limitations by presenting two algorithms for both the stationary and non-stationary learning of background image. We propose an iterative algorithm for batch learning. Unlike [19], when a new frame arrives, it is learned with the previous frame by exploiting spatial and temporal information. To the best of our knowledge, incorporating graph regularization terms (encoding data and feature similarity on low-rank) in an iterative algorithm of MC to enforce spatiotemporal coherence information is a novel approach that can be applied to model background.

III. Proposed Methodology

A. Method Overview

In this section we provide an overview of the proposed *Spatiotemporal Low-rank Matrix Completion* (SLMC) method. Our method consists of several components, which are described in the system diagram shown in Fig. 3. Initially, a dense optical flow is computed between each pair of consecutive video frames and motion-compensated binary mask is generated. This motion mask is further utilized to remove the motionless video frames from input video sequence and to prepare a set of dynamic frames, which only consists of those video clips that show dynamic scenes either because of the foreground or background variations. Then, two graphs are constructed to encode the spatiotemporal invariance of the scene background. Both of these graphs lie on two manifolds and ensure spatiotemporal smoothness. These two embedded manifolds, one among the video frames and the second among the spatial locations, are then incorporated in a unified iterative MC algorithm. The objective function is solved using matrix factorization based on alternating minimization strategy. Most of the existing methods [1], [5], [17] use batch processing while the proposed SLMC algorithm is made computationally efficient by using iterative processing approach. In the following we describe each step of the proposed SLMC algorithm in detail.

B. Notations

In this paper, we use following notations for matrices, vectors, and scalars: $\mathbf{x}_i \in \mathbb{R}^p$ denotes the *i*-th frame of a video sequence, which is represented as a column vector consisting of *p* pixels. $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n] \in \mathbb{R}^{p \times n}$ is a matrix representing all *n* frames of a sequence. Let **D** be the matrix consisting of only dynamic frames from the sequence **X**, $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_c] \in \mathbb{R}^{p \times c}$ where *c* is the number of dynamic frames. If no motionless frame exists, then *c* is equal to *n*. The recovered low-rank and sparse components are denoted by **L** and **S**, respectively. The main objective is to separate, the underlying matrix **L** corresponding to background model **B**, and component **S** belonging to foreground **F** from sequence **D**. Let **M** be the motion-compensated binary mask whose elements encode which pixel in matrix **D** belongs to **B**. Following norms of a matrix are used throughout this paper: $||\mathbf{X}||_1 = \sum_{i,j} |x_{i,j}|$ is the l_1-norm, and $x_{i,j}$ is an element of matrix **X**. $||\mathbf{X}||_F = \sqrt{\sum_{i,j} x_{i,j}^2}$ is the Frobenius norm. $||\mathbf{X}||_{max}$ is the max-norm. The general definition of $||\mathbf{X}||_{max}$ is given in (8). For the case of semi positive definite matrix, $||\mathbf{X}||_{max}$ is the maximum of the diagonal value of matrix **X**, $\max_j |x_{j,j}|$, as explained by Lee *et al.* [32]. $||\mathbf{X}||_{2,\infty} = \max_j (\sum_i x_{j,i}^2)^{\frac{1}{2}}$ is the maximum l_2 row norm of a matrix [32].

C. Mathematical Formulation

Given a sequence **D**, we require that the corresponding matrix **L** with singular vectors that are not spiky lie in a low-dimensional subspace by minimizing the following loss function under a convex optimization framework

$$\min_{\mathbf{L}} ||\mathbf{D} - \mathbf{L}||_F^2 + \lambda_1 ||\mathbf{L}||_{max}^2 \text{ s.t. } \mathbf{P}_\Omega(\mathbf{D}) = \mathbf{P}_\Omega(\mathbf{L}), \quad (1)$$

where Ω is the subset of the complete set of observed entries. To summarize the information available in **D** based on $\mathbf{P}_\Omega(\mathbf{D})$, the sampling operator $\mathbf{P}_\Omega$ is defined by

$$[\mathbf{P}_\Omega(\mathbf{D})]_{i,j} = \begin{cases} \mathbf{D}_{i,j}, & \text{if } (i,j) \in \Omega, \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

For ease of optimization, the nuclear norm is often applied on the matrix **L** in (1) to relax the matrix rank. Then the closed form solution of (1) can be obtained via Singular Value Decomposition (SVD). However, when the size of matrix **D** increases in particular in real-life applications, it is not possible to compute the principal components of such a huge matrix. Therefore, a variational form of nuclear norm was proposed in [33] and a maximum absolute value norm was used instead of nuclear norm regularization [28]. This form of max-norm is more robust than the nuclear norm in the presence of outliers [34], [35].

In the proposed method, we incorporate temporal smoothness constraint into (1) by encoding pairwise similarities among the video frames. In addition we also enforce the spatial graph regularization into (1) to incorporate the spatial smoothness among the background pixels. With these constraints, the proposed MC model is then re-formulated as

$$\begin{aligned} \min_{\mathbf{L}} ||\mathbf{M} \circ (\mathbf{D} - \mathbf{L})||_F^2 + \lambda_1 ||\mathbf{L}||_{max}^2 + \\ \gamma_1 tr(\mathbf{L}^\top \Phi_s \mathbf{L}) + \gamma_2 tr(\mathbf{L}^\top \Phi_t \mathbf{L}), \end{aligned} \quad (3)$$

where **M** constitutes the sampling operator $\mathbf{P}_\Omega(\mathbf{D})$ and "$\circ$" is element-wise multiplication. Φ_t is the Laplacian matrix of a temporal graph computed among the video frames. This is the first data manifold information that can be leveraged in the form of a discrete graph $\mathbf{G}_t$. Similarly $\Phi_s \in \mathbb{R}^{p \times p}$, is the Laplacian of a spatial graph $\mathbf{G}_s$ computed among the pixels. The regularization terms $tr(\mathbf{L}^\top \Phi_s \mathbf{L})$ and $tr(\mathbf{L}^\top \Phi_t \mathbf{L})$ in (3) are referred to as a spatiotemporal graph regularization of **L**. These terms encode a weighted penalization in the Laplacian basis. The regularization parameters γ_1, γ_2, and $\lambda_1 = 1/\sqrt{\max(p,c)}$ assign relative importance to each of the terms while optimising the objective function (3). Before solving (3), first we need to compute **D** and motion message **M**, and the graph Laplacian matrices, as described in the following sections.

D. Detection of Dynamic Video Frames

All pixels in a video exhibiting motion larger than a threshold definitely not belong to the background. The threshold is selected to be large enough so that the large motion should not result due to noise or estimation errors. We consider these pixels as missing data and try to estimate their values using the objective function (3) from the available pixels with small or no motion. The problem is still hard because many of the remaining pixels may still belong to the foreground and form outliers. The motion information is incorporated in our proposed framework by computing optical flow between each pair of consecutive frames. Most of the motion detection algorithms depend on optical flow to estimate motion between pixel values. Many studies on optical flow methods have been reported in the literature. The performance of optical flow based algorithms degrades at the boundaries of the moving objects, also known as motion boundaries. Therefore we recommend the dense optical flow method proposed in [36] because of its robustness against motion boundaries. These boundaries are the most important regions because inaccurate motion in the vicinity of the boundaries frequently produce incorrect results. This effect has also been reported in a prior work [37] since the erroneous motion vectors compromise incorrect motion models.

As discussed above, the matrix **X** comprises all the stacked column vector images. Assume that $\mathbf{x}_i$ and $\mathbf{x}_{i-1}$ are two consecutive frames at times *t* and $t-1$, respectively. The dense optical flow [36] is estimated between $\mathbf{x}_i$ and $\mathbf{x}_{i-1}$ to obtain the horizontal $\mathbf{V}^x$ and vertical $\mathbf{V}^y$ components of the motion field. Motion mask $\mathbf{M} \in \{0,1\} \in \mathbb{R}^{p \times n}$ of the entire video sequence, **X**, is then generated using the simple operation:

$$m_{i,j} = \begin{cases} 0, & \text{if } \sqrt{(v_{i,j}^x)^2 + (v_{i,j}^y)^2} \geq \tau, \\ 1, & \text{otherwise,} \end{cases} \quad (4)$$

where $m_{i,j}$, $v_{i,j}^x$, and $v_{i,j}^y$ are entries of **M**, $\mathbf{V}^x$, and $\mathbf{V}^y$ at the i^{th} rows and j^{th} columns, respectively. τ is the threshold

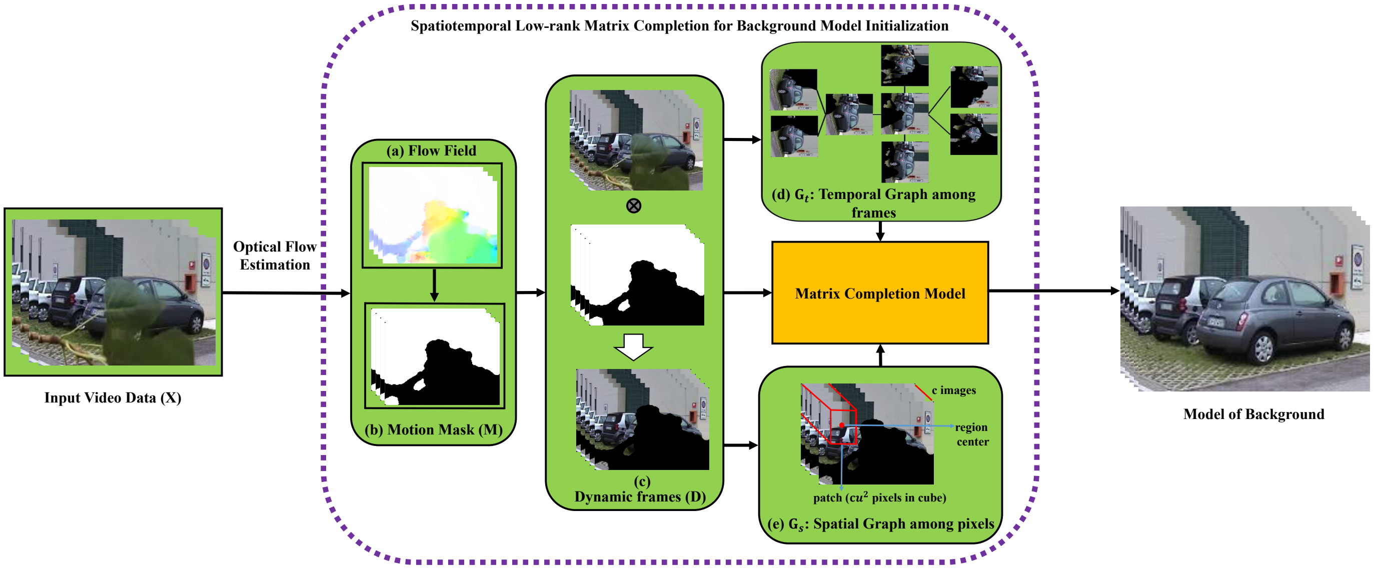

Spatiotemporal Low-rank Matrix Completion for Background Model Initialization
Input Video Data (X)
Optical Flow Estimation
(a) Flow Field
(b) Motion Mask (M)
⊗
(c) Dynamic frames (D)
(d) G$_t$: Temporal Graph among frames
Matrix Completion Model
c images
region center
patch (cu^2 pixels in cube)
(e) G$_s$: Spatial Graph among pixels
Model of Background

Fig. 3. Pictorial illustration of the proposed initialization of the model of **B**, which is heavily occluded by objects **F**. The input sequence **X** consists of two types of video frames: Motionless images, i.e., no variation in pixel values and a set of dynamic images where motion is observed in the pixel values. Steps (a) and (b) describe the output of the optical flow method and generate motion-compensated binary mask **M**. In step (c), the motionless frames are removed and motion regions are detected in **X** based on **M**. The partially observed matrix **D** only shows the set of dynamic images, where the motion is represented by the black region. In steps (d) and (e) two graphs are constructed and then finally, the proposed SLMC model is applied on **D** with spatiotemporal information to recover the model of **B**.

of motion magnitude, which is computed adaptively as the average of all the pixels in the motion field.

The next step is to prepare matrix **D** by eradicating the motionless frames in **X** using (4). It is empirically observed that the flow field is very small for motionless frames and slowly moving objects, i.e., when the pixel value does not deviate between two consecutive frames. Hence, the n^{th} frame in **X** is considered to be redundant or motionless, if all entries are 1 in the corresponding n^{th} column of **M**; otherwise, if some parts of the entry are 0, then the frame is considered as dynamic and it is augmented in matrix **D**. At this step, the size of matrix **M** is same as matrix **D** i.e., $\mathbf{M} \in \mathbb{R}^{p \times c}$, since we are only considering the motion mask of dynamic images. The steps described above are illustrated in Fig. 3 (a), (b), and (c). Using this technique, the dimension of matrix **X** may be reduced significantly depending on the number of stationary frames.

E. Motion-Assisted Spatiotemporal Regularizations

As discussed above, the estimation of low rank **L** from existing approaches [1], [5], [19] is very sensitive against outliers and noise. We try to improve the quality of **L** by incorporating the data manifold information in the form of two graphs, one along the temporal dimension, across columns **D**, and the other along the spatial dimensions, across rows of **D**. The underlying assumption is that the low-dimensional embedding of the columns and rows of **D** lies on smooth manifolds both temporally and spatially [29], [38]. Let $\mathbf{G}_t = (\mathbf{V}_t, \mathbf{E}_t, \mathbf{A}_t)$ be the temporal graph with vertex $\mathbf{V}_t$ as the frames of matrix **D**, the set of pairwise edges $\mathbf{E}_t$ between $\mathbf{V}_t$, and the adjacency matrix $\mathbf{A}_t$, which encodes the weights and connectivity of the graph.

The importance of $\mathbf{G}_t$ is demonstrated by an example from the *Foliage* sequence shown in Fig. 3 (c) that are corrupted by **M**. The set of clean images of the *Foliage* sequence is matrix **L**, because they all belong to the same sequence. The underlying representation of **L** of the images has some redundancy, i.e., the similarity between different images of the same sequence is greater than that of any other sequence. Therefore, the key to recovering matrix **L** is to exploit the notion of similarity encoded in the data.

Similarly, the spatial graph $\mathbf{G}_s = (\mathbf{V}_s, \mathbf{E}_s, \mathbf{A}_s)$ can be constructed with the set $\mathbf{V}_s$ as the rows of matrix **D**. The pairwise relationship between the pixels is information that could alternatively be exploited to refine matrix **L** for the modeling of spatially consistent **B**. In the same sequence of images, different parts of the same image might also be related to each other. For example, different parts of the same image may feature *cars* in a scene, which also get repeated in the other images. Therefore, $\mathbf{G}_s$ between the rows of matrix **D** is beneficial for exploiting this new idea of similarity between the spatial features for improving the quality of **L**.

The frames and pixels connected with similar pixel values most likely are part of **B**. Therefore a segmentation that is spatially and temporally consistent with **B** can be obtained. For this purpose we find similarity between every two frames in the temporal direction and between pixel locations in the spatial dimensions. The graphs are then built using s-nearest neighbor strategy [39]. The first step consists of searching the closest neighbors for all the samples using Euclidean distances, where each node is connected to its s nearest neighbors. Let Δ be the matrix that contains all pairwise distances, $\Delta_{i,j}$ is the Euclidean distance between $(\mathbf{d}_i \ , \mathbf{d}_j) \in \mathbf{D}$

$$\Delta_{i,j} = \sqrt{\frac{||(\mathbf{m}_i \& \mathbf{m}_j) \circ (\mathbf{d}_i - \mathbf{d}_j)||_2^2}{||(\mathbf{m}_i \& \mathbf{m}_j)||_1}}, \tag{5}$$

where $\mathbf{m}_i \& \mathbf{m}_j$ is the AND operator applied on $\mathbf{m}_i$ and $\mathbf{m}_j$, $\mathbf{m}_i$ and $\mathbf{m}_j$ are column vectors of motion mask **M**. Thus, we consider only the observed pixels because $\mathbf{m}_i(k) \& \mathbf{m}_j(k)$

is 1 if both pixels $\mathbf{d}_i(k)$ and $\mathbf{d}_j(k)$ are observed, otherwise $\mathbf{m}_i(k)\&\mathbf{m}_j(k)$ is 0, if both or any one of them is missing entry. Then, the adjacency matrix $\mathbf{A}_t$ for the $\mathbf{G}_t$ can be computed using

$$\mathbf{A}_t(i,j) = exp\left(-\frac{\Delta_{i,j} - \omega_{min}}{\sigma^2}\right), \tag{6}$$

where ω_{min} is the minimum non-zero distance in Δ, and σ^2 is the smoothing factor in $\mathbf{G}_t$, which can be set equal to the average distance of the s-nearest neighbors. In general, if $\mathbf{d}_i$ is in the s-nearest neighbors of $\mathbf{d}_j$ then there is a link between two nodes $\{\mathbf{d}_i, \mathbf{d}_j\}$ and $\mathbf{A}_t(i,j)$ is > 0; otherwise, $\mathbf{A}_t(i,j) = 0$. Maximum value of $\mathbf{A}_t(i,j)$ will be 1. Finally, the normalized temporal graph Laplacian matrix that characterizes graph $\mathbf{G}_t$ is computed as

$$\Phi_t = \mathbf{W}^{-1/2}(\mathbf{W} - \mathbf{A})\mathbf{W}^{-1/2} = \mathbf{I} - \mathbf{W}^{-1/2}\mathbf{A}\mathbf{W}^{-1/2}, \tag{7}$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{W}$ is the degree matrix defined as $\mathbf{W} = diag(w_i)$, where $w_i = \sum_j \mathbf{A}_t(i,j)$. For ease of notation we ensure the size of Φ_t to be $p \times p$. In case the number of frames c are larger than p, we select $p < c$ frames randomly to compute Φ_t. If c is less than p padding is performed to ensure the size of Φ_t to be $p \times p$.

For the construction of Laplacian of spatial graph, $\Phi_s \in \mathbb{R}^{p \times p}$, it is more reasonable to enforce smoothness on the patch level of matrix $\mathbf{D}$ rather than on the pixel level. Indeed, comparing patches of the image allows one to use the local information of the image. As a first step, we vectorize the patches that correspond to the same position for all the images of the sequence. Let u^2 be the size of each square patch centered at the pixel under consideration. Then, we form p data samples of size cu^2, as mentioned in Fig. 3 (e). These transformed data samples are then input into the graph construction algorithm discussed above to get Φ_s. For larger datasets 'Fast Library for Approximate Nearest Neighbor' (FLANN) [39] can be used to compute Laplacian matrices more efficiently.

F. Modeling of Low-rank Matrix Completion

In this section, we present our iterative MC algorithm to solve (3) for the computation of matrix $\mathbf{L}$. This is achieved by preserving prior knowledge of matrix $\mathbf{D}$ in the form of $\mathbf{M}$ such that it is more suitable for the recovery of the stationary model of $\mathbf{B}$ occluded by objects $\mathbf{F}$.

Despite (3) is completely fitting our model. However, the main drawback of problem (3) is that it requires the computation of full (or partial) SVD of matrix $\mathbf{L}$ in every iterative cycle of the algorithm, which could become prohibitively expensive when the dimensions are large. We overcome this problem by adopting a proxy for the max-norm of rank matrix $\mathbf{L}$ defined by the following matrix factorization problem as:

$$||\mathbf{L}||_{\max} = \min_{\mathbf{U}\in\mathbb{R}^{p\times r}, \mathbf{V}\in\mathbb{R}^{r\times c}} \frac{1}{2}(||\mathbf{U}||_{2,\infty}.||\mathbf{V}||_{2,\infty}) \ s.t. \ \mathbf{L} = \mathbf{U}\mathbf{V}. \tag{8}$$

This variational form of the max-norm proxy has recently been used in standard max-norm minimization algorithms [32], [34], [35] that scale to very large matrix completion problems. In (8) $\mathbf{U}$ is termed the spatial basis, and $\mathbf{V}$ represents the temporal coefficients of $\mathbf{U}$ (also known as the principal directions and components) in r-dimensional linear space, and r is a rank that is upper bounded by the rank of $\mathbf{L}$. The product $\mathbf{U}\mathbf{V}$ is known as the approximation $\mathbf{L}$ of matrix $\mathbf{D}$. Taking this into account (8) can be substituted into (3) as:

$$\begin{aligned}\min_{\mathbf{U}\in\mathbb{R}^{p\times r}\mathbf{V}\in\mathbb{R}^{r\times c}} &||\mathbf{M} \circ (\mathbf{D} - \mathbf{U}\mathbf{V})||_F^2 + \frac{\lambda_1}{2}\{\ ||\mathbf{U}||_{2,\infty}^2.||\mathbf{V}||_{2,\infty}^2\} + \\ &\gamma_1 tr(\mathbf{U}^\top\mathbf{V}^\top\Phi_s\mathbf{U}\mathbf{V}) + \gamma_2 tr(\mathbf{U}^\top\mathbf{V}^\top\Phi_t\mathbf{U}\mathbf{V}).\end{aligned} \tag{9}$$

Since $||\mathbf{V}||_{2,\infty}^2 = 1$ as explained by Shen *et al.* [35]. (9) can be simplified as follows

$$\begin{aligned}\min_{\mathbf{U}\in\mathbb{R}^{p\times r}\mathbf{V}\in\mathbb{R}^{r\times c}} &||\mathbf{M} \circ (\mathbf{D} - \mathbf{U}\mathbf{V})||_F^2 + \frac{\lambda_1}{2}||\mathbf{U}||_{2,\infty}^2 + \\ &\gamma_1 tr(\mathbf{V}^\top\mathbf{U}^\top\Phi_s\mathbf{U}\mathbf{V}) + \gamma_2 tr(\mathbf{V}^\top\mathbf{U}^\top\Phi_t\mathbf{U}\mathbf{V}).\end{aligned} \tag{10}$$

As discussed above, (10) deals with the batch processing problems in which all video frames have to be available in a memory prior to any processing. In contrast, our goal here is to derive an iterative solution of (10) over spatiotemporal graph regularizations which only processes one frame per time instance. To do so, the iterative solution of (10) can be formulated as follows:

$$\begin{aligned}\min_{\mathbf{U}\in\mathbb{R}^{p\times r}, \mathbf{v}_t} \sum_{t=1}^{c} \Big(&||\mathbf{m}_t \circ (\mathbf{d}_t - \mathbf{U}\mathbf{v}_t)||_2^2 + \gamma_1(\mathbf{v}_t^\top\mathbf{U}^\top\Phi_s\mathbf{U}\mathbf{v}_t) \\ &+\gamma_2(\mathbf{v}_t^\top\mathbf{U}^\top\Phi_t\mathbf{U}\mathbf{v}_t)\Big) + \frac{\lambda_1}{2}||\mathbf{U}||_{2,\infty}^2,\end{aligned} \tag{11}$$

which can be solved via alternating minimization strategy, in which the cost function is minimized with respect to each individual optimization variable, whereas the other functions remain fixed [28], [35]. Thus, the estimated matrix $\mathbf{L}$ is more robust than in the traditional MC-based approaches.

1) Fixing $\mathbf{U}$ *in (11): Basis Initialization:* First, we initialize matrix $\mathbf{U}$ for solving $\mathbf{V}$. In earlier MC approaches [15], these bases are selected randomly and stored in a large matrix before any optimizations are performed. In this study, $\mathbf{U}$ was first initialized with a small number of images at the beginning of the video feed, but no fewer than r of matrix $\mathbf{L}$. In addition, the corresponding $\mathbf{M}$, spatiotemporal information Φ_s and Φ_t are also incorporated in this step, i.e., $\mathbf{U} = [(\widetilde{\Phi}_t[(\mathbf{m}_1, \mathbf{m}_2, ..., \mathbf{m}_r)\circ (\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_r)]^\top\Phi_s)^\top]$, where $\widetilde{\Phi}_t$ is $r \times r$ block of Φ_t. The size of $\mathbf{U}$ is $(p \times r)$, and r is manually selected to utilize a small number of images from matrix $\mathbf{D}$, with only r samples that are used to encode the graph structures. In this case, $\mathbf{U}$ is a very small sized batch and this operation is performed only once; therefore, the complexity of using it does not consume much memory.

2) Solving for $\mathbf{V}$*:* Since the iterative method processes each column of matrix $\mathbf{D}$, $\mathbf{V}$ accumulates all of the coefficient vectors $\mathbf{v}$ for each image. In this step, $\mathbf{v}_t$ for the current image is computed using the pre-defined matrix $\mathbf{U}$ above by projecting each new sample. To that end, we keep only the terms which depend on $\mathbf{v}_t$. This stage requires a small-scale convex optimization problem as:

$$\min_{\mathbf{v}_t}||\mathbf{m}_t \circ (\mathbf{d}_t - \mathbf{U}\mathbf{v}_t)||_2^2 + \gamma_1(\mathbf{v}_t^\top\mathbf{U}^\top\Phi_s\mathbf{U}\mathbf{v}_t) + \gamma_2(\mathbf{v}_t^\top\mathbf{U}^\top\Phi_t\mathbf{U}\mathbf{v}_t). \tag{12}$$

Using a fixed $\mathbf{U}$ in (12), it constitutes a least-squares problem, which can be solved using

$$\begin{aligned}\mathbf{v}_t &= \rho_1^{-1}\mathbf{U}^\top(\mathbf{m}_t \circ \mathbf{d}_t), \text{and} \\ \rho_1 &= \mathbf{U}^\top(\widetilde{\mathbf{M}}_t + \gamma_1\Phi_s + \gamma_2\Phi_t)\mathbf{U},\end{aligned} \tag{13}$$

where $\widetilde{\mathbf{M}}_t$ is $p \times p$ matrix having m on its diagonal [40]. After some new images have been revealed or projected, $\mathbf{U}$

is a full r matrix. At this point, the previously computed $\mathbf{v}_t$ is stored in another temporary vector, e.g., $\mathbf{v}_t$, and then, in the next iteration, we can update these coefficients using the prediction function as:

$$\mathbf{v}_t = \begin{cases}\mathbf{v}_t, & \text{if } ||\mathbf{v}_t||_2 \leq 1,\\ \text{update step} & \text{otherwise,}\end{cases} \tag{14}$$

and this update is performed by incorporating a prior $\mathbf{v}_t$ on (12). This step also requires another small optimization problem [35]:

$$\mathbf{v}_t = \underset{\epsilon,\epsilon>0,\mathbf{v}_t}{\text{argmin}} \min_{||\mathbf{v}_t||_2=1} \frac{1}{2}||\mathbf{m}_t \circ (\mathbf{d}_t - \mathbf{U}\mathbf{v}_t)||_2^2 + \gamma_1(\mathbf{v}^\top\mathbf{U}^\top\Phi_s\mathbf{U}\mathbf{v}) + \gamma_2(\mathbf{v}^\top\mathbf{U}^\top\Phi_t\mathbf{U}\mathbf{v}) + \frac{\epsilon}{2}(||\mathbf{v}_t||_2^2 - 1). \tag{15}$$

The closed form solution of (15) can be obtained by taking a derivative with respect to $\mathbf{v}_t$ and setting it to 0 constitutes another least-squares solution as:

$$\mathbf{v}_t = \rho_2^{-1}\mathbf{U}^\top(\mathbf{m}_t \circ \mathbf{d}_t), \text{and} \quad \rho_2 = [\epsilon\mathbf{I} + \mathbf{U}^\top(\widetilde{\mathbf{M}}_t + \gamma_1\Phi_s + \gamma_2\Phi_t)\mathbf{U}], \tag{16}$$

where $\mathbf{I}$ is the identity matrix. The positive dual variable ϵ is introduced to solve (16) if l_2 norm of $\mathbf{v}_t$ is greater than 1. The lower bound ϵ_1 on the optimal ϵ is a constant α at this step, since $||\mathbf{v}_t||_2 \leq 1$, whereas the upper bound ϵ_2 needs to be searched for an optimal solution. Initially, we set $\epsilon_2 = r$, and then, if $||\mathbf{v}_t||_2 > 1$, it is iteratively updated at each iteration as $\epsilon_2 \leftarrow 2\epsilon_2$ until $||\mathbf{v}_t||_2 \leq 1$ is satisfied. Finally, ϵ is computed as $\frac{1}{2}(\epsilon_1 + \epsilon_2)$ to solve (16).

3) Learning $\mathbf{U}$*:* In this step, the main model is learnt for slowly changing model of $\mathbf{B}$. In general, $\mathbf{U}$ is adaptively updated whenever a new frame approaches through minimizing previously computed $\mathbf{v}_t$. This is achieved by first defining two new auxiliary matrix variables $\mathbf{Q} \in \mathbb{R}^{r\times r}$ and $\mathbf{R} \in \mathbb{R}^{p\times r}$, which retain the information about pre-computed $\mathbf{U}$ and $\mathbf{v}_t$, as:

$$\mathbf{Q}_t \leftarrow \mathbf{Q}_{t-1} + \mathbf{v}_t(\mathbf{v}_t)^\top, \quad \mathbf{R}_t \leftarrow \widehat{\mathbf{M}} \circ (\mathbf{R}_{t-1}) + \widehat{\mathbf{M}} \circ [(\mathbf{U}_{t-1})\mathbf{v}_t(\mathbf{v}_t)^\top], \tag{17}$$

where $\widehat{\mathbf{M}}$ is $p \times r$ block selected from $\mathbf{M}$. In contrast to the method in [28], we update $\mathbf{R}$ only for those pixels that belong to $\mathbf{B}$. In (17), all the information in $\mathbf{U}$ and $\mathbf{v}_t$ is updated in the current image. Then, the subgradient $\widetilde{\mathbf{U}}$ of $\frac{1}{2}||\mathbf{U}||_{2,\infty}^2$ is computed by $\widetilde{\mathbf{U}} = \frac{1}{2}\partial||\mathbf{U}||_{2,\infty}^2$. Finally each column $\mathbf{u}_j$ of $\mathbf{U}$ is then updated using the block-coordinate descent method as:

$$\mathbf{u}_j \leftarrow \rho_3^{-1}[\mathbf{u}_j - \frac{1}{\mathbf{Q}_{jj}}(\mathbf{U}\mathbf{q}_j - \mathbf{r}_j + \lambda_1\widetilde{\mathbf{u}}_j)], \text{and} \quad \rho_3 = (\widetilde{\mathbf{M}}_t + \gamma_1\Phi_s + \gamma_2\Phi_t), \tag{18}$$

where $\widetilde{\mathbf{u}}_j$ is the jth column of $\widetilde{\mathbf{U}}$, which is basically the maximum of the l_2 row norm of $\mathbf{U}$. The solution converges to the optimal solution asymptotically as compared to its batch counterpart, as proved in [33], [35], only if r is given and basis $\mathbf{U}$ is estimated as above. Furthermore, this $\mathbf{U}$ is updated column-wise and therefore it is independent from the number of samples. Hence, it solves the computational issues that arise when enforcing such hard constraints on matrix $\mathbf{L}$. Finally, matrix $\mathbf{D}$ is then recovered by component $\mathbf{L}$ that is the product of $\mathbf{U}$ and its $\mathbf{V}$, which changes sequentially at a time instance t. Alg. 1 presents the details of SLMC.

Indeed, image $\mathbf{B}$ is then recovered by computing the average values in the columns of $\mathbf{L}$, excluding first those r images, resulting in a vector $\mathbf{b} \in \mathbb{R}^{p\times 1}$, which is then reshaped into a matrix $\mathbf{B} \in \mathbb{R}^{m_1\times m_2}$ having width m_1 and height m_2 as mentioned in the final step in Alg. 1. Fig. 4 (d) shows the estimated model of $\mathbf{B}$ obtained by SLMC.

Algorithm 1 SLMC for static model of **B** occluded by **F**.

```
 1: procedure SLMC ($\mathbf{X} \in \mathbb{R}^{p\times n}, \lambda_1, \gamma_1, \gamma_2, \alpha, r, \eta$) ▷ Inputs
 2:     Compute $\mathbf{M}$ using (4)
 3:     $\mathbf{D} \leftarrow [\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3, ..., \mathbf{d}_c], \mathbf{L} \in \mathbb{R}^{p\times c}$    ▷ Initialize input
 4:     Compute $\Phi_s$ and $\Phi_t$ using (4), (5), (6), and (7)
 5:     $\mathbf{U} \in \mathbb{R}^{p\times r}, \mathbf{V} \in \mathbb{R}^{r\times c}, \mathbf{v} \in \mathbb{R}^{r}$    ▷ Dual variables
 6:     $\mathbf{Q} \in \mathbb{R}^{r\times r}, \mathbf{R} \in \mathbb{R}^{p\times r}, \mathbf{e} \in \mathbb{R}^{p\times 1}$    ▷ Auxilliary matrices
 7:     while $not\ converge$ do
 8:         Fix U and solve $\mathbf{V}(:,t) \leftarrow \mathbf{v}_t$ using (12) and (13)
 9:         Update $\mathbf{v}_t$ using (14), (15), and 16)
10:         Solve Q and R using (17)
11:         Update $\mathbf{U}(:,j) \leftarrow \mathbf{u}_j$ using (18)
12:         Compute $\mathbf{y} \leftarrow$max$[(\mathbf{d} - \mathbf{U}\mathbf{v}) - \lambda_1, 0]$
13:         Compute $\mathbf{e} \leftarrow \mathbf{y}$+min$[(\mathbf{d} - \mathbf{U}\mathbf{v}) + \lambda_1, 0]$
14:         if $\frac{max(||\mathbf{e}||_2, ||\mathbf{v}||_2)}{p} < \eta$    ▷ Convergence
15:             break
16:         else
17:             repeat step 8 to 15 until convergence
18:     end while
19:     return U, V
20:     Compute the model B
21:     $\mathbf{L} \leftarrow \mathbf{U}\mathbf{V}$
22:     $\mathbf{B} \in \mathbb{R}^{m_1\times m_2} \leftarrow \mathbf{b} \leftarrow mean[\mathbf{L}(:, 1 + r : end)]$
23: end procedure
```

G. Extension to SRPCA

The solution of (3), is obtained after the convergence of the iterative procedure. The recovered matrix $\mathbf{L}$ represents the $\mathbf{B}$ of the entire sequence, but it is modeled by using only the entries corresponding to $\mathbf{B}$ from $\mathbf{M}$ and the average of all columns in matrix $\mathbf{L}$. The component $\mathbf{S}$ that belongs to objects $\mathbf{F}$ cannot be modeled explicitly using (3) since the observation of matrix $\mathbf{S}$, which contains outliers of the objects $\mathbf{F}$, is already utilized by (3) using $\mathbf{M}$. Matrix $\mathbf{S}$ is fully optimized by considering the case of a dynamic sequence, in which $\mathbf{B}$ scene changes continuously at each frame. SRPCA aims to decompose matrix $\mathbf{D}$ into non-stationary matrices $\mathbf{L}$ and $\mathbf{S}$ by converting (3) into the following constrained problem as:

$$\min_{\mathbf{L},\mathbf{S}}||\mathbf{D} - \mathbf{L} - \mathbf{S}||_F^2 + \lambda_1||\mathbf{L}||_{max}^2 + \lambda_2||\mathbf{S}||_1 + \gamma_1 tr(\mathbf{L}^\top\Phi_s\mathbf{L}) + \gamma_2 tr(\mathbf{L}^\top\Phi_t\mathbf{L}), \tag{19}$$

where the l_1-norm on matrix $\mathbf{S}$ imposes the sparsity constraints on the pixels of objects $\mathbf{F}$. Incorporation of spatiotemporal graphs regularization in matrix $\mathbf{L}$ enhances the robustness of the proposed component $\mathbf{S}$ against noise and dynamic pixels. Thus, a spatiotemporally coherent mask of $\mathbf{F}$ can be obtained, thereby reducing many false positives as shown in Fig. 5 (d) to (e). (19) can also be solved using an alternative minimization technique. The entire algorithm that is used to solve (19) is summarized in Alg. 2. The shrink$(\cdot)$ in step (10) of Alg. 2 is known as the soft-thresholding function defined as shrink$(\mathbf{S}) =$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2016.2632302, IEEE Transactions on Circuits and Systems for Video Technology

Algorithm 2 SRPCA for the dynamic separation of **B**-**F**.

```
Input: $\mathbf{D} \in \mathbb{R}^{p\times c}, \Phi_s \in \mathbb{R}^{p\times p}, \Phi_t \in \mathbb{R}^{p\times p}, \mathbf{U} \in \mathbb{R}^{p\times r}, \mathbf{V} \in \mathbb{R}^{r\times c}, \mathbf{v} \in \mathbb{R}^{r}, \mathbf{Q} \in \mathbb{R}^{r\times r}, \mathbf{R} \in \mathbb{R}^{p\times r}, \mathbf{e} \in \mathbb{R}^{p\times 1}, \lambda_1, \lambda_2, \gamma_1, \gamma_2, \alpha, r$
Output: L, S
Set S=0.
repeat
    Initialize $\mathbf{U} = (\widetilde{\Phi}_t[\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_r]^\top \Phi_s)^\top$,
    Minimize V with respect to U, removing m from (12), (13) changes to
        $\mathbf{v}_t = \rho_4^{-1}\mathbf{U}^\top(\mathbf{d}_t - \mathbf{s}_t)$, and
        $\rho_4 = [\mathbf{U}^\top(\gamma_1\Phi_s + \gamma_2\Phi_t)\mathbf{U}]$          (20)
    Update V by ignoring M in (14), (15), (16)
    Learn U with respect to v by ignoring M in (17) and (18), (18) changes to
        $\mathbf{u}_j \leftarrow \rho_5^{-1}[\mathbf{u}_j - \frac{1}{\mathbf{Q}_{jj}}(\mathbf{U}\mathbf{q}_j - \mathbf{r}_j + \lambda_1\widetilde{\mathbf{k}}_j)]$, and
        $\rho_5 = (\gamma_1\Phi_s + \gamma_2\Phi_t)$,          (21)
    Compute $\mathbf{L} \leftarrow \mathbf{UV}$
    Update S as $\mathbf{S} \leftarrow \text{shrink}[\mathbf{d} - \mathbf{U}\mathbf{v}]$.
until convergence
```

$\text{sign}(\mathbf{S})\max(\text{abs}(\mathbf{S}), 0)$. Fig. 5 (c) and (e) are the results of **F** detection using the RPCA [13] and proposed SRPCA methods.

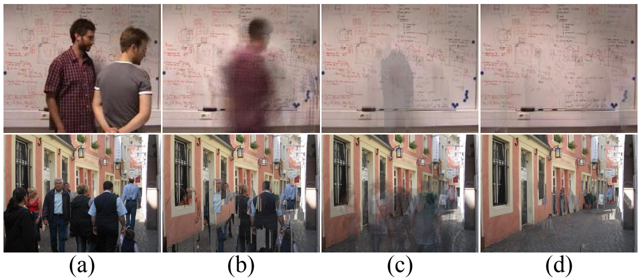

(a) (b) (c) (d)

Fig. 4. Testing of spatiotemporal model to show robustness in the initialization of the stationary model of **B** against severe occlusions and a cluttered scene. From left to right: (a) two video frames, (b) static model **B** obtained by [28], (c) results by MC without spatiotemporal regularizations, and (d) results by the proposed SLMC. From top to bottom: 2 frames captured from the *Board* and *Toscana* sequences of the SBMI dataset [16].

IV. Experimental Evaluations

We test both of the proposed algorithms by conducting extensive evaluations on challenging datasets including new SBMI [16], I2R [21], Wallflower [22], and CDnet14 [20]. First, we test the proposed SLMC algorithm for the task of reconstructing a constant background model **B**. Then, we perform experiments on highly dynamic scenes of **B** to detect objects **F** using SRPCA. Both quantitative and qualitative results are reported, followed by a description of the implementation and computational complexities.

A. Evaluations of SLMC on SBMI Dataset

The SBMI dataset[1] [16] consists of 14 different videos recorded indoors and outdoors. Note that many state-of-the-art approaches [1], [7], [9], [19] were tested on very simple scenes of **B**, while SBMI dataset comprises of complex scenes in which **B** is largely occluded by objects **F**. Therefore this dataset permits a rigorous comparison of initialization techniques suitable for constructing a static model of **B**. However real world videos may contain more than one states of the static background scenes. For instance, the *lobby* sequence (see Fig. 7 (1^{st} row) and Table III) contains multiple static scenes of background with all lights on and a few lights off and the task is to estimate foreground-free image for each static state.

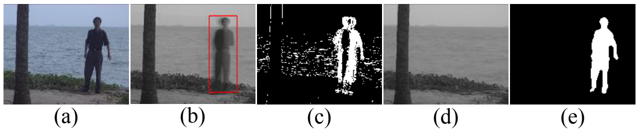

(a) (b) (c) (d) (e)

Fig. 5. Testing of dynamic segmentation of **B** and **F**. From left to right: (a) one input frame, (b)-(c) model **B** and mask of objects **F** of classical RPCA algorithm [13] (red rectangle in (b) shows ghost appearance in recovered image of **B**), (d)-(e) **B** and **F** segmentation results of non-stationary scene by the SRPCA.

TABLE I
DETAILS OF THE 14 SEQUENCES OF THE SBMI DATASET USED IN OUR EXPERIMENTS..

Seq. Name	Size×No. of frames	Challenges	Size of D
Board	$[200, 164] \times 228$	**B** is less visible	$[200, 164] \times 228$
Candela	$[352, 288] \times 350$	Intermittent object motion	$[352, 288] \times 156$
CAVIAR1	$[384, 256] \times 610$	Slowly moving people	$[384, 256] \times 414$
CAVIAR2	$[384, 256] \times 460$	Slowly moving and stop people	$[384, 256] \times 366$
CaVignal	$[200, 136] \times 258$	Intermittent object motion	$[200, 136] \times 116$
Foliage	$[200, 144] \times 395$	Severe occlusions	$[200, 144] \times 386$
Hall & Monitor	$[352, 240] \times 296$	Slowly moving people	$[352, 240] \times 250$
HighwayI	$[320, 240] \times 439$	Moving cars	$[320, 240] \times 439$
HighwayII	$[320, 240] \times 499$	Moving cars	$[320, 240] \times 490$
HumanBody2	$[320, 240] \times 740$	Moving people with dominant **F**	$[320, 240] \times 740$
IBMtest2	$[320, 240] \times 90$	Short sequence of moving people	$[320, 240] \times 90$
People and Foliage	$[320, 240] \times 349$	**B** is more occluded by **F**	$[320, 240] \times 338$
Snellen	$[144, 144] \times 320$	Severe occlusions **B** is not visible	$[144, 144] \times 316$
Toscana	$[800, 600] \times 6$	Short sequence of cluttered **B** scene	$[800, 600] \times 6$

Both SLMC and SRPCA can be employed to efficiently estimate background model for each of these states. The size of matrix **D** may or may not be different in this case. To employ SLMC, the final step in Algo. 1 may be ignored and the best estimate of **B** may be directly found from **L**. Since SLMC is designed to cope with occluded scenes of **B**, it is more suitable to handle such sequences.

Table I provides more details of the SBMI dataset. In Fig. 6 1st row shows input images and 2nd row shows the only one available ground truth image of **B** for each SBMI sequence. We group these sequences into following three categories.

(**i**) ***B** is heavily occluded by **F** objects*: Five videos including *Board*, *Snellen*, *Foliage*, *People & Foliage*, and *Toscana* depict situations in which **B** is either little or not visible and is largely occluded by **F** objects about 90% of the frames. For example, in *Board* two people appear and one person starts dancing to occlude **B**. Similarly, moving leaves occupy most of the **B** area most of the time in *Snellen*, *Foliage*, and *People & Foliage* sequences. In these sequences, only 2% clean **B** frames are available. In SLMC motionless frames are removed and **D** only contains dynamic frames. Size of **D** differs from **X** which contains the original video sequence sizes described in Table I. Moreover, *Toscana* is a very short sequence consisting of only six frames corresponding to a crowded scene.

(**ii**) *Intermittent object motion*: Three SBMI sequences including *Candela*, *CaVignal*, and *Hall & Monitor* belong to this category. For instance, in *Candela* sequence, one man enters the indoor scene carrying a small bag and then sits on the sofa

[1] http://sbmi2015.na.icar.cnr.it/

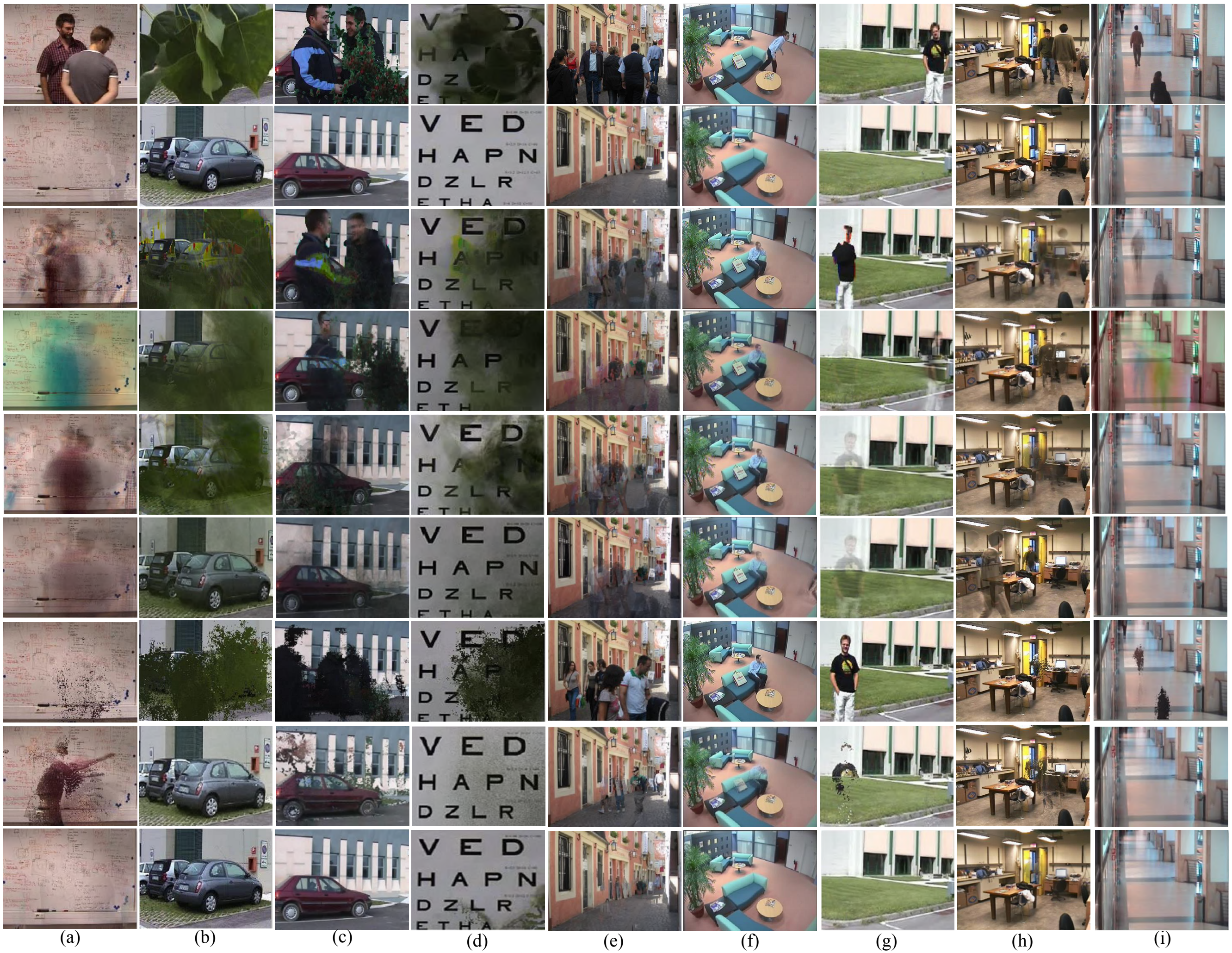


Fig. 6. Qualitative results of the proposed method. From left to right: 9 typical video frames of, (a) *Board*, (b) *Foliage*, (c) *People and Foliage*, (d) *Snellen*, (e) *Toscana*, (f) *Candela*, (g) *CaVignal*, (h) *HumanBody2*, and (i) *Caviar1*. From top to bottom: 1st row: input images, 2nd row: true background images **G**, 3rd row: results by the DECOLOR [1], 4th row: the GoDec [6], 5th row: the GRASTA [19], 6th row: the RMAMR [17], 7th row: the IMBS [8], 8th row: the SOBS [7], 9th row: and the proposed SLMC. The first *5* rows from (a) to (e) belong to a very difficult sequences, where **B** is severely occluded and cluttered by objects **F**, whereas (f) to (g) and (h) to (i) demonstrate the *Intermittent Object Motion* and *Bootstrap* conditions.

in about 50% of the frames of the entire sequence. Then, he abandons the bag and leaves the sofa. In this case, the number of frames in **D** is also different from that in sequence **X**, as indicated in Table I. The same situation is also observed in the *Hall & Monitor* sequence, in which some training images that are available at the beginning become redundant. The sequence named *CaVignal* is even more challenging. Indeed, the only man appearing in the sequence stands motionless on the left of the scene for the first 60% of the sequence, see Fig. 1. Then starts moving slowly towards the right, before suddenly stopping on the right, see Fig. 6 (7th column). Approximately 50% of the redundant frames are extracted from this sequence, see Table I.

(iii) *Bootstrap Sequences*: The six videos including *CAVIAR1*, *CAVIAR2*, *HighwayI*, *HighwayII*, *HumanBody2*, and *IBMtest2* are related to the bootstrap situation in which clean background frames do not exist neither in the beginning nor in the middle of the sequence. However in each frame the number of background pixels are larger than the foreground pixels. For instance, in *HighwayI* and *HighwayII* sequences, the highway is always crowded with cars which keep on moving throughout the sequence. In each frame **B** is revealed for at least 50% of the pixels.

1) Qualitative Results and Comparison: We compare visual quality of the reconstructed **B** with several state-of-the-art approaches including 40 existing methods based on subspace learning and multiple features, as well as non-parametric methods for stable recovery of **B**. The implementation of these methods is publicly available in Background Subtraction (BGS) [3] and Low-rank and Sparse (LRS) [41] libraries.

Because of space limitations, we present comparison of qualitative results of the proposed SLMC algorithm with six most noteworthy methods, including *DEtecting Contiguous Outliers in the LOw-rank representation* (DECOLOR) [1], *Go Decomposition* (GoDec) [6], *Grassmanian Robust Adaptive Subspace Tracking Algorithm* (GRASTA) [19], and RMAMR [17]. A non-parametric method *Independent Multimodal Background Subtraction* (IMBS) [8] and a neural network-based method *Self-Organizing approach to Background Subtraction* (SOBS) [7] are also used for comparison. The visual comparison results over nine sequences are illustrated in Fig. 6.

2) Quantitative Evaluations and Analysis: We quantitatively compare the quality of the results of the proposed

TABLE II
QUANTITATIVE ANALYSIS ON SBMI DATASET USING 6 ACCURACY MEASURES. SEE FIG. 6 FOR VISUAL COMPARISON. THE BEST MODEL OF **B** CORRESPONDS TO LOWER VALUES OF AGE, PEPS, PCEPS AND HIGHER VALUES OF MS-SSIM, PSNR, AND CQM.

Methods	**Evaluation Measures**	**Board**	**Candela**	**CAVIAR1**	**CAVIAR2**	**CaVignal**	**Foliage**	**Hall&Monitor**	**HighwayI**	**HighwayII**	**HumanBody2**	**IBMtest2**	**People&Foliage**	**Snellen**	**Toscana**	Average
DECOLOR [1]	**AGE**	25.93	3.88	7.60	1.07	18.20	51.27	3.11	21.42	9.95	12.29	7.11	50.14	65.98	14.74	20.91
	pEPs	0.37	0.03	0.08	0.00	0.10	0.71	0.00	0.26	0.08	0.14	0.05	0.72	0.87	0.26	0.26
	pCEPs	0.28	0.02	0.07	0.00	0.07	0.59	0.00	0.24	0.05	0.09	0.03	0.61	0.80	0.18	0.22
	MS-SSIM	0.65	0.94	0.93	0.99	0.40	0.38	0.98	0.40	0.71	0.87	0.93	0.35	0.47	0.83	0.73
	PSNR	16.21	27.58	24.46	42.02	17.02	11.86	34.09	15.80	19.12	20.99	27.57	11.31	10.73	20.96	21.41
	CQM	31.27	40.26	44.09	57.52	29.35	25.51	45.49	22.13	27.49	31.70	37.74	21.38	24.83	32.82	33.68
GoDec [6]	**AGE**	22587	9.42	14.28	21.09	17.92	39.28	3.86	2.40	2.89	9.47	23.86	29.32	59.61	11.52	19.17
	EPs	11100	4792	14039	55809	4251	17631	2869	350	607	6457	29459	36625	16289	68240	19180
	pEPs	0.33	0.04	0.14	0.56	0.15	0.61	0.03	0.00	0.00	0.08	0.08	0.47	0.78	0.14	0.24
	CEPs	8377	3265	11980	44745	3094	14586	1708	70.03	41.78	3504	16599	28900	14778	47530	14227
	pCEPs	0.25	0.03	0.12	44745	0.11	0.50	0.02	0.00	0.04	0.21	28900	0.37	0.71	0.09	0.21
	MS-SSIM	0.81	0.94	0.95	0.45	0.81	0.65	0.94	0.98	0.98	0.86	0.52	0.79	0.53	0.91	0.83
	PSNR	18.52	25.17	23.70	0.96	18.63	15.26	29.18	36.33	32.31	21.00	17.48	17.04	11.26	23.50	22.17
	CQM	43.98	38.48	34.01	31.41	31.69	28.76	43.08	58.72	38.93	34.40	28.62	28.16	25.10	36.46	35.84
GRASTA [19]	**AGE**	28.00	3.88	1.80	5.22	11.16	25.85	3.33	4.13	3.25	9.70	2.94	41.04	43.98	12.77	14.14
	pEPs	0.40	0.03	0.00	0.00	0.10	0.59	0.02	0.01	0.00	0.09	0.00	0.82	0.86	0.21	0.22
	pCEPs	0.32	0.02	0.00	0.00	0.07	0.45	0.01	0.00	0.00	0.06	0.00	0.75	0.79	0.16	0.19
	MS-SSIM	0.69	0.94	0.96	0.95	0.93	0.89	0.95	0.96	0.96	0.92	0.97	0.84	0.85	0.86	0.91
	PSNR	15.73	27.58	38.43	41.22	24.80	19.06	30.21	32.56	31.09	22.88	36.17	14.86	14.03	22.03	26.58
	CQM	32.02	40.26	49.82	41.35	39.61	33.63	40.73	57.59	45.68	35.86	44.36	26.46	36.91	32.72	40.58
RMAMR [17]	**AGE**	28.42	5.05	1.55	1.78	12.00	12.51	2.04	2.76	2.70	12.34	4.46	38.82	20.21	18.67	14.22
	pEPs	0.54	0.03	0.00	0.00	0.14	0.63	0.00	0.00	0.00	0.15	0.03	0.83	0.89	0.31	0.25
	pCEPs	0.43	0.02	0.00	0.00	0.09	0.47	0.00	0.00	0.00	0.09	0.02	0.77	0.83	0.19	0.21
	MS-SSIM	0.79	0.92	0.96	0.97	0.90	0.89	0.99	0.97	0.99	0.84	0.92	0.85	0.88	0.66	0.89
	PSNR	17.20	27.41	41.22	50.32	24.31	18.41	37.97	35.88	35.66	19.77	29.13	15.14	17.22	19.53	27.62
	CQM	41.75	40.13	55.64	57.74	39.80	33.23	46.32	58.62	46.20	35.47	41.20	27.59	40.24	30.53	42.26
SOBS [7]	**AGE**	24.90	3.68	1.68	2.66	4.09	**3.82**	2.44	1.22	0.65	8.78	7.2	15.10	16.88	7.86	7.21
	pEPs	0.31	0.02	0.00	0.00	3.19	0.55	0.98	0.00	0.00	0.07	0.01	10.02	37.35	0.07	3.97
	pCEPs	0.22	0.01	0.00	0.00	1.60	**0.00**	0.32	0.00	0.00	0.03	0.00	5.01	24.37	0.04	2.23
	MS-SSIM	0.56	0.94	0.84	0.81	0.87	**0.99**	0.96	0.99	**0.99**	0.86	0.93	0.75	0.93	0.90	0.88
	PSNR	16.69	26.65	38.37	46.32	21.85	**31.77**	30.93	42.68	44.63	22.21	29.18	16.61	**21.25**	23.55	27.50
	CQM	30.35	39.44	49.55	60.32	42.26	**39.13**	43.18	65.57	54.37	34.94	39.72	35.36	**44.74**	27.50	**43.90**
IMBS [8]	**AGE**	9.08	5.09	5.92	3.70	3.75	19.49	1.57	1.92	3.24	5.17	5.09	13.62	17.50	20.97	15.49
	pEPs	0.06	0.04	0.04	0.00	0.03	0.14	0.00	0.00	0.00	0.02	0.03	0.10	0.18	0.29	0.28
	pCEPs	0.00	0.02	0.02	0.00	0.01	0.03	0.00	0.00	0.00	0.00	0.02	0.01	0.13	0.24	0.22
	MS-SSIM	0.66	0.92	0.90	0.98	0.90	0.80	0.99	0.98	0.98	0.92	0.91	0.97	**0.95**	0.69	0.87
	PSNR	21.66	25.93	22.14	34.32	24.29	18.17	38.86	39.56	35.12	25.45	26.86	**23.90**	21.01	16.91	22.29
	CQM	32.96	36.27	34.68	45.20	38.93	31.90	48.36	54.89	38.31	37.13	37.33	31.85	41.60	24.21	40.16
SLMC	**AGE**	**3.39**	**0.43**	**0.00**	**0.00**	**0.00**	12.02	**0.00**	**1.21**	**0.00**	**2.36**	**1.57**	**4.29**	**7.00**	**4.71**	**2.85**
	pEPs	**0.01**	**0.00**	**0.00**	**0.00**	**0.00**	**0.07**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.04**	**0.12**	**0.04**	**0.02**
	pCEPs	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.02**	**0.00**
	MS-SSIM	**0.87**	**0.98**	**0.99**	**0.99**	**0.99**	0.94	**0.99**	**0.99**	0.95	**0.96**	**0.97**	**0.94**	0.76	**0.91**	**0.94**
	PSNR	**28.45**	**44.57**	**88.76**	**68.89**	**50.68**	25.63	**70.69**	**66.78**	**61.44**	**28.63**	**52.13**	**29.42**	20.29	**24.33**	**47.19**
	CQM	**47.27**	**61.33**	**68.99**	**61.23**	**68.36**	33.36	**80.33**	**71.42**	**66.55**	**42.36**	**66.55**	**39.68**	27.89	**41.05**	**54.24**

algorithm with existing methods using six different criteria as suggested by the authors of SBMI [16]. These criteria are summarized as follows:

- *Average of the Gray-level Error (AGE)* is ℓ_1 norm of the difference of ground truth image **G** and the estimated background.
- *pEPs* is the percentage of error pixels with respect to the total number of pixels in the image.
- *pCEPs* is the percentage of clustered error pixels with respect to the total number of pixels in the image.
- *MS-SSIM* is the multi-scale structural similarity index measure which estimates the perceived visual distortion.
- *PSNR* is the peak signal-to-noise ratio is

$$PSNR = 10log_{10}\frac{(g-1)^2}{MSE}, \tag{22}$$

  where MSE is the *Mean Squared Error* between **G** and **B**, and g is number of gray levels.
- *CQM* is the color image quality measure which estimates the quality of a color between the **G** and **B**.

The goal is to minimize the AGE, pEPs, and pCEPs values for more accurate recovery of **B** while MS-SSIM, PSNR, and CQM should be maximized. For fair comparison, we used the optimal set of parameters for each method as suggested by the original authors. Table II presents the results of the performance of SLMC using the above mentioned criteria and comparison with other existing approaches. For sequences that belong to the intermittent object motion category, the majority of the methods failed to compensate for persistent clutter. For instance, in the beginning of the frames in the *CaVignal* sequence. In contrast, some methods such as SOBS and GoDec, were unable to effectively process the slowly moving person in *CaVignal*, as well as the remaining subject who stay in the scene longer for the next 20% of video frames. Similarly, this situation is also observed in the case of the *Candela* sequence. Indeed, no method could effectively process persistent outliers (in the form of a man and a small bag) in the final estimation of **B**. These methods follow a less frequent update strategy that does not enable them to process these cases appropriately. In contrast, SLMC gives the most promising results for the *Candela* and *CaVignal* sequences, and a comparable performance for the *Hall & Monitor* video. The qualitative analysis of the accuracy results in terms of the values of pEPs and pCEPs are presented in Table II. In addition, the AGE values were quite low in the case of reduced objects **F** as compared to the region containing the entire image in *Candela* and *Hall & Monitor*, whereas, this figure inclines in the *CaVignal* sequence. Overall, SLMC provides the most accurate and stable model of **B**, as opposed to RMAMR, SOBS, and IMBS in this category, all of which were found to show noticeable outliers.

We now consider the more challenging sequences, in which **B** is less visible than objects **F**, although moving leaves

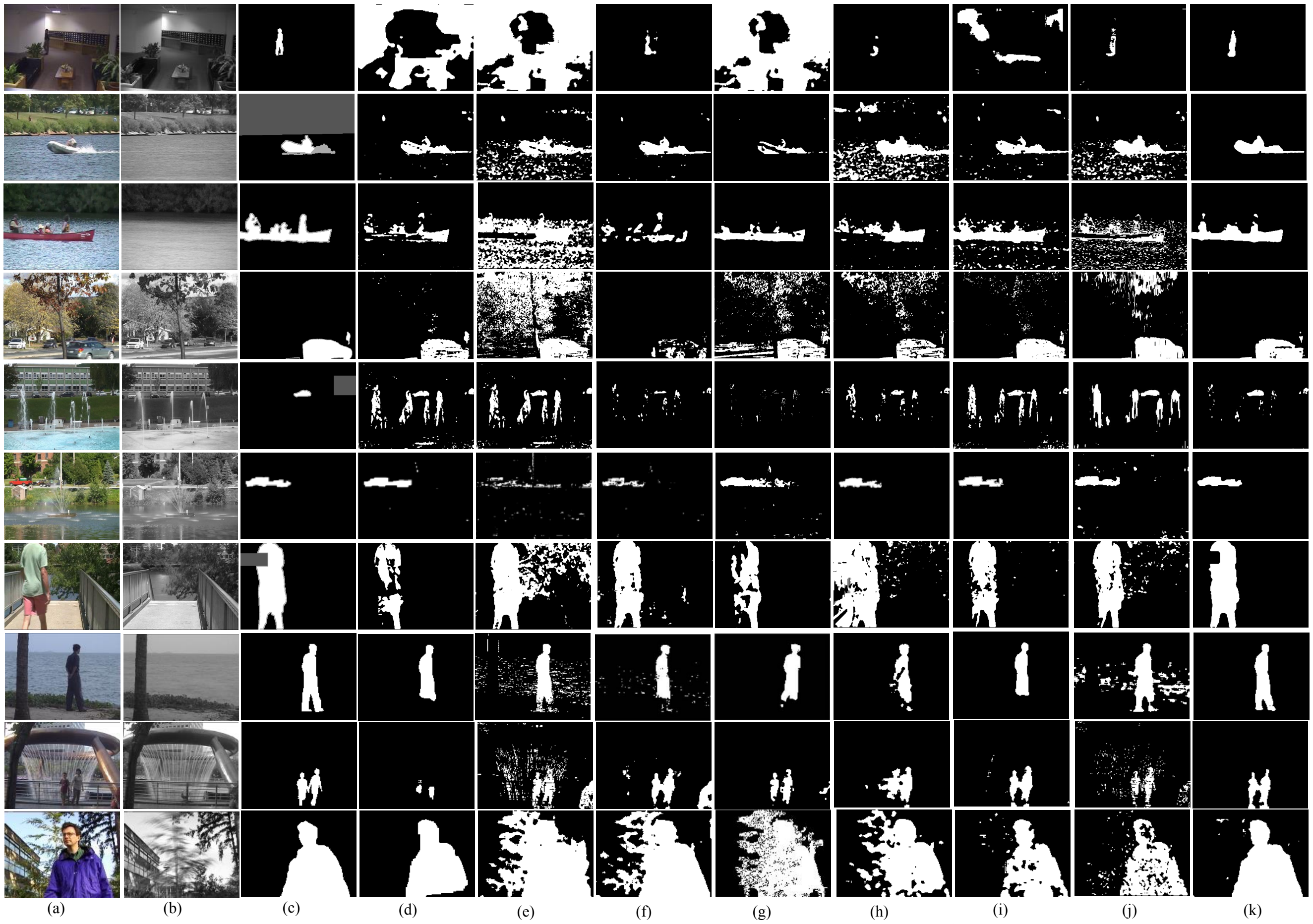


Fig. 7. Visual results of the proposed method. From left to right: (a) 10 input images, (b) estimated images of **B** by SRPCA, (c) ground truth images, and the segmentation results of objects **F** obtained by (d) DECOLOR [1], (e) GoDec [6], (f) GRASTA [19], (g) TVRPCA [5], (h) SOBS [7], (i) RMAMR [17], (j) SLMC, and (k) SRPCA. From top to bottom: input images of 10 sequences, i.e., (1) *lobby*, (2) *boats*, (3) *canoe*, (4) *fall*, (5) *fountain1*, (6) *fountain2*, (7) *overpass*, (8) *watersurface*, (9) *fountain*, (10) *waving trees*.

or artificial plants occupy almost the entire region of **B** in more than half of the video frames in the *Foliage*, *People & Foliage*, and *Snellen* sequences. Only SLMC, RMAMR, and SOBS show an encouraging performance, as evident from the comparatively low values of AGE, pEPs, and pCEPs. However, a significant discrepancy is seen in the PSNR, MS-SSIM, and CQM values. Indeed, the remaining algorithms, for example, IMBS, GRASTA, and DECOLOR, produce a greenish halo clutter as an outlier of **F** in the final model of **B**. Thus, these methods achieve poor accuracy. For the *Board* sequence, a large variation is seen in the values of AGE and pCEPs among the top performers. Only SLMC and IMBS produce a good model **B** in the presence of the dancing man. This observation is clearly shown in the noteworthy values of AGE, pEPs, and pCEPs obtained for these methods. All the other methods produce a large outlier in terms of AGE and other values for this sequence, and hence, the quality of **B** produced by these methods is poor. The dancing man, who occupies a large portion of **B** scene, leads every strategy to include the contribution of the isolated man in the final model of **B**. In addition, for a cluttered scene, such as *Toscana*, only SLMC is capable of achieving noteworthy statistics. Indeed, all the methods fail to appropriately process a crowded scene followed by the weak update mechanism. Table II demonstrates that SLMC outperformed the other approaches. Improved accuracy of SLMC is because of the spatiotemporal consistency constraints.

For the bootstrapping cases, Table II shows that SLMC provides promising performance for these videos. In addition, most of the compared methods provided a good estimation of the model of **B**. This is because **B** scene is either more dominant than **F** or some training data is available at some location in the sequence.

B. Evaluations of the Proposed SRPCA Algorithm on Dynamic Sequences

The proposed SRPCA algorithm is tested on ten challenging videos selected from three different datasets (see Table III for details). Ground truth for **B** for these sequences is not available. However, ground truth **G** of foreground objects is available which we compare with the estimated **F**. Table III indicates that the size of **D** is equal to the size of **X** (entire video sequence), because the pixel values are changing continuously except for the *lobby* sequence.

In this experiment, we compared SRPCA with seven algorithms including DECOLOR [1], GoDec [6], RMAMR [17], GRASTA [19], SOBS [7], and *Total Variation regularized RPCA* (TVRPCA) [5]. In addition, SRPCA is also compared with the proposed SLMC whose **F** mask is obtained by taking the absolute difference between input image and estimated

TABLE III
DESCRIPTION OF THE 10 SEQUENCES TAKEN FROM THE 3 DATASETS.

Seq. Name	Size×No. of frames	Challenges	Size of D
Lobby (I2R [21])	$[160, 128] \times 1,415$	Light being switched on/off in a lobby	$[160, 128] \times 843$
Boats (CDnet14 [20])	$[320, 240] \times 3,000$	Shimmering water	$[320, 240] \times 7,999$
Canoe (CDnet14 [20])	$[320, 240] \times 1,190$	Slowly moving canoe and rippling water	$[320, 240] \times 1,190$
Fall (CDnet14 [20])	$[720, 480] \times 4,000$	Dynamic **B** with swaying trees	$[720, 480] \times 4,000$
Fountain1 (CDnet14 [20])	$[432, 288] \times 1,184$	Fountains with fast moving car	$[432, 288] \times 1,184$
Fountain2 (CDnet14 [20])	$[432, 288] \times 1,499$	Fountain with water fall	$[432, 288] \times 1,499$
Overpass (CDnet14 [20])	$[320, 240] \times 3,000$	Waving trees with slowly moving person	$[320, 240] \times 3,000$
Water Surface (I2R [21])	$[160, 128] \times 633$	Rippling water and **F** lingering object	$[160, 128] \times 633$
Fountain (I2R [21])	$[160, 128] \times 524$	Fountain	$[160, 128] \times 524$
Waving Trees (Wallflower [22])	$[160, 120] \times 286$	Swaying of tree	$[160, 120] \times 286$

model of background computed by Alg. 1. A best threshold is also selected empirically for estimated **F** segment in case of SLMC. Fig. 7 displays the binary masks of objects **F** obtained for some of the relevant frames relating to the ten sequences and their comparison with other methods. In addition, the quantitative performance of the discrimination between **F** and **B** is evaluated in terms of the F_1 score.

$$F_1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}, F_1 \in \mathbb{R}(0,1). \tag{23}$$

Results are shown in Table IV for all *7* algorithms that were compared. In this table we show average *Precision*, *Recall*, and F_1 score over 10 video clips.

For complex dynamics as presented by the *lobby* sequence, in which **B** scene undergoes an illumination change midway through the sequence of frames, GRASTA, SLMC, and proposed SRPCA approaches produce good results in terms of the average F_1 score. A closer visual inspection of **F** for the *lobby* video (see Fig. 7 (1st row) from (d) to (j)) shows that all 7 of the compared algorithms fail to discriminate the segmentation of **B**-**F** while the light is being switched off in the lobby. For instance, the DECOLOR, GoDec, TVRPCA, and RMAMR algorithms generate very noisy segments of **F** as seen in Fig. 7 (d), (e), (g), and (i), respectively. In contrast, the proposed SRPCA scheme accurately adjusts the dynamic illumination changes into the estimated model of **B**. The spatiotemporal constraints that are incorporated into the proposed method improve the segmentation of **F** as seen in the obtained values of *Precision*, *Recall*, and F_1 score.

In the case of most complex dynamic scenarios of **B**, the results in Table IV show that the SRPCA algorithm attains, on average, the highest *Precision*, *Recall*, and F_1 scores, outperforming all comparative algorithms. The first challenging aspect to consider is the detection of **F** objects under variations in **B** if objects **F** remain in front of an extremely dynamical region of **B**. As a result, a noisy mask of **F** appears inside the regions being detected. This effect was thoroughly inspected in the *fall*, *overpass*, and *waving trees* videos mentioned in Fig. 7 (4th, 7th, and last row) (d) to (i). Only the SRPCA method succeeded in achieving performance of more than 90% in terms of F_1 score.

Another important point is if **B** contains highly dynamical regions that may be detected as elements of **F**, as is the case with *Fountain1*, *Fountain2*, and *Fountain* videos. As shown in Fig. 7 (d) to (j), 7 of the methods that are compared all mistakenly detect the fountains as **F** objects in these sequences. In contrast, SRPCA and RMAMR methods can successfully detect objects under these circumstances to discriminate between **F** and **B** because these schemes include motion assistance for pixel variability (see Fig. 7 (i) and (k)). Furthermore, the spatiotemporal continuity introduced in the proposed algorithm allows the inclusion of highly dynamical information of **B**.

The third challenging situation arises when object **F** moves slowly within a highly dynamical region of **B** as in the case of the *Boats*, *Canoe*, and *Water Surface* sequences (see Fig. 7 (d) to (k) (2nd, 3rd, and 8th row)). Here, all 7 of the compared methods perform the worst discrimination since the pixels of object **F** are also encoded into the model **B** as shown in Fig. 7 (d) to (j). Likewise, DECOLOR, GRASTA, and SOBS algorithms are unable to correctly discriminate the slowly changing scene involving the moving canoe in the model of **B**, leading to false segmentations. Because of motion assistance in RMAMR, it outperforms all the other methods except SRPCA. However, in the case of *Water Surface*, some regions of **F** object are not detected and are persistently miss-classified as **B** by all the methods that were compared. In contrast, SRPCA correctly identifies these slowly moving objects. Unlike SLMC, the first important aspect in SRPCA is that both **B** and **F** components are optimized simultaneously. Secondly, the availability of spatiotemporal information and the ability to detect dynamic video frames enables SRPCA to adapt the model to dynamic changes in **B** scene, thereby improving the segmentation task. Overall, the results we attained demonstrate that SRPCA allows for improved discrimination between **F** and **B** compared to all other methods.

C. Implementation Details and Computational Time

Execution time of all algorithms was compared on a machine with 3.0 GHz Intel core i5 processor and 4GB RAM. Solution of the proposed models (3) and (19) require a set of 11 parameters including $r, \tau, \lambda_1, \lambda_2, \alpha, \gamma_1, \gamma_2, \eta, \sigma, u^2$, and *s*. The rank r was set to 10 in order to rapidly update the model of **B** and this was followed by a block-coordinate descent method. τ is a threshold for redundant frame decision, which is estimated automatically as discussed in Section III (D). λ_1, λ_2, and α are regularization parameters, which were all set according to $1/\sqrt{\max(p,c)}$ as suggested by Candés *et al.* [13]. After setting λ_1, the parameters γ_1 and γ_2 are set to 10 as used by [30], [31]. The purpose is to make the manifold terms dominate in the objective function. The constant η was used as stopping criterion to enable Alg. 1 to converge and it was set to 10^{-6}. The remaining three parameters were related to the construction of $\mathbf{G}_t$ and $\mathbf{G}_s$. The parameter σ controls the smoothness on the graphs and was set to 0.05, which was effective in all the experiments; it can also be adapted as the average distance of the connected samples. Provided that σ is not large, the parameter does not affect the final quality of **B**. For the construction of $\mathbf{G}_s$ on image patches, we used $u^2 = 25$. The number of nearest neighbors *s* is set 10 for both graphs. In addition, we used the FLANN [39] libraries for more efficient computation of the graphs. Both graphs were constructed using the open source toolbox called GSPBox[2] [42]. More tuned values of these parameters may have generated even better

[2] Available for public use: https://lts2.epfl.ch/gsp/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2016.2632302, IEEE Transactions on Circuits and Systems for Video Technology

TABLE IV
COMPARISON OF RECALL, PRECISION, AND F_1 SCORE ON DYNAMIC VIDEOS (SEE FIG. 7 AND TABLE III.)

Methods	lobby			boats			canoe			fall			fountain1			fountain2			overpass			watersurface			fountain			waving trees		
	Re	Pre	F_1	Re	Pre	F_1	Re	Pre	F_1	Re	Pre	F_1	Re	Pre	F_1	Re	Pre	F_1	Re	Pre	F_1	Re	Pre	F_1	Re	Pre	F_1	Re	Pre	F_1
DECOLOR [1]	0.17	**0.99**	0.29	0.36	0.12	0.19	0.70	0.85	0.73	0.76	0.53	0.61	**0.96**	0.01	0.02	**1.00**	0.47	0.64	0.82	0.72	0.81	0.72	0.98	0.83	0.20	**0.95**	0.33	0.87	0.89	0.88
GoDec [6]	0.07	0.72	0.12	0.34	0.13	0.18	0.64	0.65	0.42	0.37	0.45	0.60	0.60	0.01	0.11	0.72	0.55	0.38	0.66	0.73	0.66	0.85	0.62	0.72	0.68	0.28	0.40	0.98	0.53	0.68
RMAMR [17]	0.15	0.99	0.27	0.38	0.23	0.78	0.78	0.87	0.81	0.72	0.80	0.75	0.74	0.42	0.51	0.97	0.95	**0.96**	0.78	0.87	0.82	0.62	**0.99**	0.76	**0.81**	0.74	**0.77**	0.79	0.87	0.83
GRASTA [19]	0.80	0.81	0.80	**0.94**	0.50	0.66	0.30	0.95	0.46	0.27	0.87	0.42	0.40	0.18	0.24	0.75	0.77	0.75	0.81	0.93	0.87	0.85	0.86	0.85	0.8	0.48	0.60	0.97	0.59	0.74
TVRPCA [5]	**0.91**	0.30	0.45	0.51	0.53	0.52	0.54	**0.99**	0.70	0.82	0.33	0.48	0.40	0.01	0.12	0.56	**0.98**	0.72	0.65	0.95	0.77	0.86	0.84	0.84	0.70	0.76	0.73	0.94	0.52	0.67
SOBS [7]	0.72	0.33	0.46	0.27	0.11	0.15	0.54	0.70	0.64	0.74	0.62	0.61	0.30	0.61	0.32	0.90	0.82	0.88	0.64	0.78	0.68	0.65	0.89	0.75	0.80	0.59	0.68	0.84	0.64	0.73
SLMC	0.79	0.76	0.77	0.44	0.47	0.46	0.40	0.28	0.33	**0.91**	0.34	0.50	0.52	0.52	0.52	0.72	0.57	0.64	0.61	0.69	0.64	**0.90**	0.52	0.66	0.73	0.39	0.51	0.75	0.85	0.79
SRPCA	0.86	0.80	**0.83**	0.72	**0.95**	**0.82**	**0.89**	**0.99**	**0.94**	0.85	**0.99**	**0.92**	0.86	**0.78**	**0.82**	0.84	0.97	0.90	**0.87**	**0.97**	**0.92**	0.89	0.97	**0.93**	0.78	0.75	0.76	**0.99**	**0.95**	**0.97**

results, however we have emphasized on generalization of the proposed algorithm over unseen datasets.

The time complexities were also investigated during our experiments. For fixed values of $s = 10$ and $u^2 = 25$, the complexity of $\mathbf{G}_t$ is $O(pc(\log(c)))$ and that of $\mathbf{G}_s$ is $O(cp\log(p))$ [31]. In addition, the proposed MC method is an iterative approach and its complexity is $O(pr^2)$. In contrast to earlier RPCA batch approaches, the proposed method processes only one frame per time instance and updates $\mathbf{B}$ when a new sample arrives. Taking its complexity as $O(pr^2)$, SLMC is independent from the number of video samples, but proportional to r. As the method hardly takes 5 iterations per frame, it is linear to the sample size and almost linear to the ambient dimensions; therefore, it is much more efficient and outperforms earlier methods [1], [6], [17], [19]. Thus, the overall time complexity, including graph constructions, of Alg. 1 above is $O(p(c\log(c)+r^2+c\log(p)))$ and the memory required by SLMC is also reduced to $O(pr)$, since it is not dependent on video samples.

To compare the overall computational time including graphs construction step of SLMC and SRPCA, we first selected a very short sequence named *IBMtest2* (see Table 1) from SMBI dataset. We then created a batch of 90 frames with image resolution of 240×320. For fair comparison with the above mentioned approaches, the time is recorded in seconds. Fig. 8 presents the performance in terms of computational time and it is noticed that both SLMC and SRPCA achieve the most promising results as compared to the previous algorithms. Since both $\mathbf{B}$ and $\mathbf{F}$ components are simultaneously optimized, the time is effected in case of SRPCA but it is still attractive for a surveillance video processing. All these experimental investigations reveal that the proposed SLMC and SRPCA methods show a very nice potential for the robust estimation of model $\mathbf{B}$ and detection of foreground objects in terms of a very good accuracy and speed.

V. CONCLUSION

In this paper, two fast algorithms including SLMC and SRPCA are presented for estimation of stationary as well as dynamic background models. The proposed algorithms are based on iterative processing and hence solve some of the problems associated with traditional batch processing methods. In SLMC, first redundant samples are removed from the input matrix to alleviate the difficulty of supporting outliers. SLMC provides an efficient and more reliable mechanism to recover the background component, even in the presence of missing entries, using matrix completion together with max-norm constraints. Moreover, the model of matrix $\mathbf{L}$ is well maintained by exploiting the idea of similarity in the form of graph Laplacian regularizations. The key aspect of SLMC is its capacity to generate an accurate model of $\mathbf{B}$ even if it is occluded by $\mathbf{F}$ objects. Large-scale experimental evaluations on different datasets using six different criteria demonstrated that the proposed scheme achieved promising performance as compared to the existing methods. Online construction of graphs and background model for scenes recorded by moving and pan tilt zoom cameras remain an open challenge. We plan to investigate the possibility of extending the proposed method to scenes that are more crowded and that are recorded using a moving camera by further extending the notion of data similarity using coarse to fine strategy.

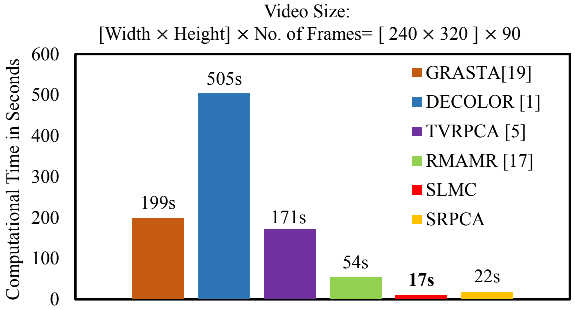


Fig. 8. Comparison of computational time in seconds for SBMI sequence.

ACKNOWLEDGMENTS

The authors would like to thank Nauman Shahid [31] (Signal Processing Laboratory, Ecole Polytechnique Federale de Lausanne (EPFL), Switzerland) for his collaboration and positive feedback with discussion on the graph part. This research is supported by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the ITRC (Information Technology Research Center) (IITP-2016-H8601-16-1002) supervised by the IITP (Institute for Information & communications Technology Promotion).

REFERENCES

[1] 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, 2013.

[2] T. Bouwmans and E. H. Zahzah, “Robust PCA via Principal Component Pursuit: A review for a comparative evaluation in video surveillance,” *Comp. Vis. Image Und.*, vol. 122, pp. 22–34, 2014.

[3] A. Sobral and T. Bouwmans, “A library framework for algorithms evaluation in foreground/background segmentation,” in *Background Modeling and Foreground Detection for Video Surveillance*, 2014.

[4] M. Ebdelli, O. Le Meur, and C. Guillemot, “Video inpainting with short-term windows: application to object removal and error concealment,” *IEEE Trans. Image Process.*, vol. 24, no. 10, pp. 3034–3047, 2015.

[5] X. Cao, L. Yang, and X. Guo, “Total Variation Regularized RPCA for Irregularly Moving object Detection Under Dynamic Background,” *IEEE Trans. Cybernetics*, vol. 46, no. 4, pp. 1014–1027, 2016.

[6] T. Zhou and D. Tao, “Godec: Randomized low-rank & sparse matrix decomposition in noisy case,” in *in Proc. Int. Conf. Mach. Learn.*, 2011.

[7] L. Maddalena and A. Petrosino, "The SOBS algorithm: what are the limits?" in *IEEE Int. Conf. Comp. Vis. Pattern Recog. Workshops*, 2012.

[8] D. Bloisi and L. Iocchi, "Independent multimodal background subtraction," in *Int. Conf. Comp. Mod. Objects Presented in Images, Fund., Meth. and App.*, 2012.

[9] J. Yao and J.-M. Odobez, "Multi-layer background subtraction based on color and texture," in *IEEE Int. Comp. Vis. Pattern Recog.* IEEE, 2007.

[10] T. Bouwmans, F. El Baf, B. Vachon *et al.*, "Statistical background modeling for foreground detection: A survey," *Hand. Pattern Recog. Comp. Vis.*, 2010.

[11] T. Bouwmans, "Traditional and recent approaches in background modeling for foreground detection: An overview," *Comp. Sci. Rev.*, vol. 11, pp. 31–66, 2014.

[12] H. Han, J. Zhu, S. Liao, Z. Lei, and S. Z. Li, "Moving object Detection Revisited: Speed and Robustness," *IEEE Trans. Circuits Syst. Vid. Tech.*, vol. 25, no. 6, pp. 910–921, 2015.

[13] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust Principal Component Analysis?" *J. of ACM*, vol. 58, no. 3, p. 11, 2011.

[14] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Found. Comp. Math.*, vol. 9, no. 6, pp. 717–772, 2009.

[15] A. Sobral, T. Bouwmans, and E.-h. Zahzah, "Comparison of matrix completion algorithms for background initialization in videos," in *Int. Conf. Image Anal. Process. Work.* Springer, 2015.

[16] L. Maddalena and A. Petrosino, "Towards benchmarking scene background initialization," in *Proc. ICIAP*. Springer, 2015.

[17] 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. Vid. Tech.*, vol. 25, no. 11, pp. 1721–1734, Nov 2015.

[18] H. Mansour and A. Vetro, "Video background subtraction using semi-supervised robust matrix completion," in *in Proc. IEEE Int. Conf. Acous. Sp. Sig. Process.* IEEE, 2014.

[19] J. He, L. Balzano, and A. Szlam, "Incremental gradient on the Grassmannian for online foreground and background separation in subsampled video," in *IEEE Int. Comp. Vis. Pattern Recog.* IEEE, 2012.

[20] Y. Wang, P.-M. Jodoin, F. Porikli, J. Konrad, Y. Benezeth, and P. Ishwar, "Cdnet 2014: An expanded change detection benchmark dataset," in *in Proc. IEEE Conf. Comp. Vis. Pattern Recog. Work.* IEEE, 2014.

[21] 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, 2004.

[22] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers, "Wallflower: Principles and practice of background maintenance," in *IEEE ICCV*, 1999.

[23] 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," *arXiv preprint arXiv:1511.01245*, 2015.

[24] N. M. Oliver, B. Rosario, and A. P. Pentland, "A Bayesian computer vision system for modeling human interactions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 831–843, 2000.

[25] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma, "Robust Principal Component Analysis: Exact recovery of corrupted low-rank matrices via convex optimization," in *Adv. Neural Inf. Process. Syst.*, 2009.

[26] Z. Lin, A. Ganesh, J. Wright, L. Wu, M. Chen, and Y. Ma, "Fast convex optimization algorithms for exact recovery of a corrupted low-rank matrix," in *Comp. Adv. Mul. Sens. Adapt. Process.* Citeseer, 2009.

[27] S. Wang, D. Liu, and Z. Zhang, "Nonconvex relaxation approaches to robust matrix recovery," in *Int. J. Conf. Art. Intell.* AAAI Press, 2013.

[28] S. Javed, S. Oh, A. Sobral, T. Bouwmans, and S. Jung, "Background subtraction via superpixel-based online matrix decomposition with structured foreground constraints," in *IEEE Int. Conf. Comp. Vis. W.*, 2015.

[29] B. Jiang, C. Ding, B. Luo, and J. Tang, "Graph-Laplacian PCA: Closed-form solution and robustness," in *IEEE CVPR*, 2013.

[30] M. Yin, J. Gao, Z. Lin, Q. Shi, and Y. Guo, "Dual Graph Regularized Latent Low-rank Representation for Subspace Clustering," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 4918–4933, 2015.

[31] N. Shahid, V. Kalofolias, X. Bresson, M. Bronstein, and P. Vandergheynst, "Robust Principal Component Analysis on Graphs," in *IEEE Int. Conf. Comp. Vis.*, 2015.

[32] J. D. Lee, B. Recht, N. Srebro, J. Tropp, and R. R. Salakhutdinov, "Practical large-scale optimization for max-norm regularization," in *Adv. Neural Inf. Process. Syst.*, 2010.

[33] J. Feng, H. Xu, and S. Yan, "Online Robust PCA via stochastic optimization," in *Adv. Neural Inf. Process. Syst.*, 2013.

[34] A. Jalali and N. Srebro, "Clustering using max-norm constrained optimization," in *Int. Conf. Mach. Learn.* Omnipress, 2012.

[35] J. Shen, H. Xu, and P. Li, "Online optimization for Max-Norm Regularization," in *Adv. Neural Inf. Process. Syst.*, 2014.

[36] C. Liu, "Beyond pixels: exploring new representations and applications for motion analysis," Ph.D. dissertation, Citeseer, 2009.

[37] S. Denman, C. Fookes, and S. Sridharan, "Improved simultaneous computation of motion detection and optical flow for object tracking," in *IEEE Int. Conf. Dig. Image Comp.:Tech. App.* IEEE, 2009.

[38] T. Jin, J. Yu, J. You, K. Zeng, C. Li, and Z. Yu, "Low-rank matrix factorization with multiple hypergraph regularizer," *Pattern Recog.*, vol. 48, no. 3, pp. 1011–1022, 2015.

[39] M. Muja and D. G. Lowe, "Scalable nearest neighbor algorithms for high dimensional data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 11, pp. 2227–2240, 2014.

[40] S. Chouvardas, M. A. Abdullah, L. Claude, and M. Draief, "Robust Online Matrix Completion on Graphs," *arXiv preprint 1605.04192*, 2016.

[41] A. Sobral, T. Bouwmans, and E.-h. Zahzah, "LRS Library: Low-rank and Sparse tools for Background Modeling and Subtraction in Videos," in *Robust Low-Rank and Sparse Matrix Decomposition: App. Image Vid. Process.*, 2016.

[42] N. Perraudin, J. Paratte, D. Shuman, V. Kalofolias, P. Vandergheynst, and D. K. Hammond, "Gspbox: A toolbox for signal processing on graphs," *arXiv preprint arXiv:1408.5781*, 2014.

Sajid Javed received his BSc (Hons) degree in computer science from University of Hertfordshire, Hatfield, United Kingdom, in 2010. He has been a combined masters and doctoral candidate in the Virtual Reality Lab at Kyungpook National University, Republic of Korea, since 2012. His active research areas are background modeling and foreground object detection, robust principal component analysis, matrix completion, and subspace clustering. He has an active collaboration with MIA Lab in France.

Arif Mahmood received his Masters and the Ph.D. degrees in Computer Science from Lahore University of Management Sciences, Lahore, Pakistan in 2003 and 2011 respectively. Currently he is Postdoc researcher in the Department of Computer Science and Engineering, Qatar University, Doha. His current research direction is action detection and person segmentation in crowded environments. Before this he worked as Research Assistant Professor with the School of Mathematics and Statistics, the University of the Western Australia. He worked on characterizing structure of complex networks using Machine Learning techniques. Before that he was Research Assistant Professor with the School of Computer Science and Software Engineering, UWA and worked on hyper-spectral object recognition and action recognition using depth images. His major research interests are in Computer Vision and Pattern Recognition. More specifically he has performed research in data clustering, classification, action and object recognition using image sets.

Thierry Bouwmans is an associate professor at the University of La Rochelle, France. His research interests include detection of moving objects. He is the creator and administrator of the background subtraction web site. He served as the lead guest editor in two editorial works: (1) Special issue in MVA on background modeling for foreground detection in real world dynamic scenes, (2) Handbook on Background Modeling and Foreground Detection in CRC Press.

Soon Ki Jung is a professor in the School of Computer Science and Engineering at Kyungpook National University, Republic of Korea. He received his MS and PhD degrees in computer science from Korea Advanced Institute of Science and Technology (KAIST), Korea, in 1992 and 1997, respectively. He has been a visiting professor at University of Southern California, USA, in 2009. He has been an active executive board member of Human Computer Interaction, Computer Graphics, and Multimedia societies in Korea. Since 2007, he has also served as executive board member of IDIS Inc. His research areas include a broad range of computer vision, computer graphics, and virtual reality topics.