Moving Object Detection on RGB-D Videos Using Graph Regularized Spatiotemporal RPCA

Conference Paper · September 2017

DOI: 10.1007/978-3-319-70742-6_22

4 authors:

Sajid Javed
KUCARS UAE
70 PUBLICATIONS **1,815** CITATIONS
SEE PROFILE

Thierry Bouwmans
La Rochelle Université
147 PUBLICATIONS **5,968** CITATIONS
SEE PROFILE

Maryam Sultana
Mohamed bin Zayed University of Artificial Intelligence
17 PUBLICATIONS **329** CITATIONS
SEE PROFILE

Soon Ki Jung
Kyungpook National University
191 PUBLICATIONS **2,073** CITATIONS
SEE PROFILE

Moving Object Detection on RGB-D Videos Using Graph Regularized Spatiotemporal RPCA

Sajid Javed[1], Thierry Bouwmans[2], Maryam Shah[1], and Soon Ki Jung[1]

[1]School of Computer Science and Engineering, Kyungpook National University, 80 Daehak-ro, Buk-gu,Daegu, 702-701, Republic of Korea
{***sajid@vr.knu.ac.kr***},{***maryam@vr.knu.ac.kr***},{***skjung@knu.ac.kr***}
[2]Laboratoire MIA (Mathematiques, Image et Applications)- Université de La Rochelle, 17000, France, {***thierry.bouwmans@univ-lr.fr***}

Abstract. The detection of moving objects is the fundamental step in surveillance video analysis and meaningful toward object tracking and other higher level computer vision tasks. Nevertheless, many state-of-the-art methods are still limited in accurately detecting the moving objects because of complex background scenes such as illumination condition, color saturation, and shadows etc. *Robust Principal Component Analysis* (RPCA) models have shown potential for moving object detection, where input data matrix is decomposed into a low-rank matrix representing the background image and a sparse component identifying moving objects. Three deficiencies, however, still exist. First, one has to first restructure/transform the multi-way input video sequence into a data matrix. Such a preprocessing step usually leads to the information loss and would cause performance degradation. Second, RPCA models usually perform the matrix decomposition using batch processing. Therefore, it is very difficult for previous models to achieve real-time processing for disentangling moving objects from input scene. Third, previous models usually process color or intensity features for matrix decomposition. The lack of additional features also degrades the overall performance specially when color of background and foreground pixels is saturated. To address these problems, we evaluate the performance of *Online Spatiotemporal Robust Principal Component Analysis* (OS-RPCA) algorithm [1] for moving objects detection using RGB-D videos. OS-RPCA is a graph regualrized algorithm which preserves the low-rank background spatiotemporal information in the form of dual spectral graphs. One graph is constructed among the columns of data matrix and second graph is constructed among the spatial locations of the data matrix. A novel objective function is designed which encodes the spatiotemporal graph regularized constraints. The objective function is solved using online optimization scheme. We evaluate OS-RPCA algorithm for moving object detection on new RGB-D dataset, and show competitive results as compared to 8 state-of-the-art methods.

1 Introduction

The segmentation of moving objects from video sequence is the first step in surveillance video analysis [2] and higher level computer vision and image processing tasks, such as anomaly detection [3], salient object detection [4], image rain streak removals [5], video

inpainting [6], and visual object tracking [7]. This pre-processing step consists of disentangling moving objects known as foreground from the static or dynamic scene called background component. Moving object detection becomes more challenging when the video sequences comprising strong and mild illumination changes, foreground objects that are very close in color to the background, and foreground or background objects that are too close to/far from the sensor [8]. Moreover, sequences containing foreground objects in all their frames called bootstrapping, shadows caused by foreground objects, and sequences with scenarios known for causing ghosting artifacts in the detected motion, i.e., abandoned foreground objects or removed foreground objects, have remained a challenging scenarios [9, 10].

Many methods have been reported to cope with these aforementioned complex background scenes in the literature [9, 11–20]. Moving object segmentation from these videos is still a challenging problem [8, 9, 21, 22]. Subspace learning methods, such as Robust Principal Component Analysis (RPCA) [23], has gained significant attention in the past few years for background-foreground separation [21, 22]. In [23], J . Wright *et al.* presented the theory of RPCA for the extraction of redundant and grossly corrupted information from input data matrix. Background-Foreground modeling is considered as a matrix decomposition problem in [23, 24] as:

$$\min_{\mathbf{B},\mathbf{F}} ||\mathbf{B}||_* + \lambda||\mathbf{F}||_1, \text{ such that}, \mathbf{X} = \mathbf{B} + \mathbf{F}, \tag{1}$$

where $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n] \in \mathbb{R}^{p\times n}$ is the input video sequence of *n* frames, and each $\mathbf{x}_i \in \mathbb{R}^p$ denotes i-th frame. RPCA-based matrix decomposition model defined by (1) assumes that the sequence $\mathbf{X}$ can be represented as a combination of highly redundant information part (e.g., visually consistent background regions) and a grossly corrupted sparse component (e.g., distinctive foreground object regions). The redundant information part usually lies in a low dimensional feature subspace, which can be approximated by a low-rank feature matrix called background component $\mathbf{B}$. In contrast, the foreground component $\mathbf{F}$ deviating from the low-rank subspace can be viewed as noise or errors, which are represented by a sparse sensory matrix. It is shown in [24] that if the singular vectors of $\mathbf{B}$ component satisfy some incoherent conditions, $\mathbf{B}$ is low-rank and $\mathbf{F}$ is sufficiently sparse, then $\mathbf{B}$ and $\mathbf{F}$ can be recovered with high probability by solving the convex optimization method proposed in [24].

In (1), $||\mathbf{B}||_*$ denotes the nuclear norm (sum of the singular values of $\mathbf{B}$), $||\mathbf{F}||_1$ is the l_1-norm (sum of the absolute values of all the entries in $\mathbf{F}$), and $\lambda = 1/\sqrt{\max(p, n)}$. RPCA and its extensions have been sucessfully applied for background-foreground detection [14, 19, 16, 15]. Interested readers can explore more details in [21, 22].

One major shortcoming of RPCA is that it can only handle two dimensional input data matrix. Hence, the spatial information losses and causes performance degradation. Second, majority of the RPCA models work under batch processing, that is, in order to solve (1), one has to store all video frames in a memory before optimization. Such a batch processing fashion is not ideal for real-time processing. Third, previous state-of-the-art RPCA models [14, 19, 21] processes only color or intensity information as a result the performance also degrades because of lack of additional features.

To address these problems, this paper presents the evaluation of *Online Spatiotemporal* RPCA (OS-RPCA) algorithm [1] for moving object detection on RGB-D video

sequences. OS-RPCA 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 RPCA to incorporate the preceding two steps for the separation of background and foreground components. 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 video samples from the original sequence $\mathbf{X}$ and creates a set comprising only the dynamic video clips. Second, spatiotemporal graph Laplacians are computed to encode the local similarity in dynamic sequence. The eigen values between $\mathbf{B}$ and estimated from the Laplacian matrix preserves the local structure of sequence $\mathbf{X}$. Finally, iterative optimization scheme 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 [19], OS-RPCA processes only one video frame from the set of dynamic sequence per time instance via stochastic optimization. The low-rank and sparse components estimated by the OS-RPCA method is more robust and accurate than that obtained by previous RPCA approaches [21, 22]. It is because of the manifold information encoded in the graph Laplacian between the frames and pixels of the set of dynamic sequence.

We evaluate the performance of OS-RPCA algorithm on a new RGB-D dataset designed for moving object detection. We test OS-RPCA algorithm on each video of RGB-D dataset by using only intensity, RGB, and depth features. Experimental evaluations show that the performance OS-RPCA algorithm by employing intensity features is significantly better than RGB and depth features.

The rest of this paper is organized as follows. In Section 2, the related work is reviewed. Section 3 describes the OS-RPCA algorithm in detail. Experimental results are discussed in Section 4, and finally the conclusion and some future directions are shown in Section 5.

2 Related Work

Over the past few decades, numerous research works have been carried out on background subtraction also known as foreground detection [21, 22] as well as background initialization [1, 13]. In background subtraction, emphasize is to improve the accuracy of foreground detection. While the task of estimating a foreground-free image is called background modeling. Many surveys are also contributed to these topics [21, 22]. In this paper we propose a novel algorithm for background modeling which is inspired from subspace learning methods. Therefore, in this section we mainly focus on background-foreground separation methods based on subspace learning.

Wright *et al.* [23] presented the first proposal of RPCA to handle the outliers in data. Candeś *et al.* [24] used RPCA for background-foreground separation. RPCA-based approaches for background-foreground separation are not ideal for surveillance applications mainly because these approaches suffer from high computational complexity. The traditional RPCA implementations processed data in batches. Many studies have been reported in literature to make the batch methods faster [1, 19]. However batch methods are not real-time and mostly work offline. Some online methods have also been

reported to handle this problem while global optimality is still the challenging issue in these approaches [1, 21].

Many authors have contributed interesting works in the direction of enhancing only foreground detection[1]. For this purpose, a number of constraints have been suggested [11, 19, 20]. For example, Cao *et al.* [11] improved the performance of foreground detection by proposing total variation regularized RPCA method. Zhao *et al.* [19] proposed a markov random field constraint on the foreground matrix to eliminate noise and small background movements. Though the segmentation performance improved, but foreground regions tend to be over-smoothed [16] because of neighboring pixels smoothing constraints. Unfortunately, research has not been focused improving low-rank background modeling[2] [22]. Therefore, there is need to design robust algorithm to recover background component in the challenging scenarios of real-life [25]. In this study, the main goal it to evaluate the performance of OS-RPCA algorithm [1] by using only intensity, RGB, and depth features for moving object segmentation. We also compare the performance of OS-RPCA algorithm with several state-of-the-art methods on RGB-D videos.

3 The OS-RPCA Algorithm [1]

The proposed *Online Spatiotemporal RPCA* (OS-RPCA) algorithm consists of several components. Initially, a dense optical flow is computed between each pair of consecutive video frames and motion-compensated binary mask $\mathbf{M}$ is generated. This motion mask is utilized to remove the motionless video frames from sequence $\mathbf{X}$ and to prepare a set of dynamic frames $\mathbf{D} \in \mathbb{R}^{p \times c}$ (where c is the number 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 optimization scheme. The objective function is solved using matrix factorization based on alternating minimization strategy. Most of the existing methods [19, 14] use batch processing while the proposed OS-RPCA algorithm is made computationally efficient by using iterative processing approach. In the following, we describe each step of the proposed OS-RPCA algorithm in detail.

3.1 Objective Function Formulation

Given a sequence $\mathbf{D}$, OS-RPCA algorithm decomposes it into the sum of background and foreground components by minimizing the loss function defined by (1). For ease of optimization, we use the maximum norm [26] on the matrix $\mathbf{B}$ to relax the matrix rank. The general definition of maximum norm is given in 5[26]. Model (1) becomes

$$\min_{\mathbf{B},\mathbf{F}} ||\mathbf{B}||^2_{max} + \lambda||\mathbf{F}||_1, \text{ such that}, \mathbf{D} = \mathbf{B} + \mathbf{F}. \tag{2}$$

[1] http://wordpress-jodoin.dmi.usherb.ca/results2014/

[2] http://pione.dinf.usherbrooke.ca/results/

We incorporate temporal smoothness constraint into (2) by encoding pairwise similarities among the video frames. We also enforce the spatial graph regularization into (2) to incorporate the spatial smoothness among the background pixels. With these constraints, the proposed OS-RPCA model is then re-formulated as:

$$\min_{\mathbf{B},\mathbf{F}} ||\mathbf{B}||^2_{max} + \lambda||\mathbf{F}||_1 + \gamma_1 \text{tr}(\mathbf{B}^\top \Phi_s \mathbf{B}) + \gamma_2 \text{tr}(\mathbf{B}^\top \Phi_t \mathbf{B}), \text{ such that, } \mathbf{D} = \mathbf{B} + \mathbf{F}, \quad (3)$$

where $\Phi_t \in \mathbb{R}^{c\times c}$ 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 discrete graph. Similarly, $\Phi_s \in \mathbb{R}^{p\times p}$ is the Laplacian of a spatial graph computed among the spatial locations of sequence $\mathbf{D}$. The regularization terms $\text{tr}(\mathbf{B}^\top \Phi_s \mathbf{B})$ and $\text{tr}(\mathbf{B}^\top \Phi_t \mathbf{B})$ in (3) are known as spatiotemporal graph regularization of background model. These terms encode a weighted penalization on the Laplacian basis.

Incorporation of spatiotemporal graph constraints on matrix $\mathbf{B}$ enhances the robustness of the component $\mathbf{F}$ against noise and dynamic pixels. As a result, spatiotemporally coherent foreground mask can be obtained. The parameters γ_1 and γ_2 assign relative importance to each of the terms while optimizing (3). Before optimizing (3), we first compute the matrix $\mathbf{D}$ and graph Laplacian matrices as described in the following sections.

3.2 Removing Motionless Video Frames

We remove motionless video frames by computing the dense optical flow [27] between two consecutive frames $\mathbf{x}_i$ and $\mathbf{x}_{i-1}$ at times *t* and *t-1*, respectively. This pre-processing step assists OS-RPCA algorithm to avoid overwhelming outliers appear in the intermittent foreground object motion sequences. To prepare matrix $\mathbf{D}$, 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 $\mathbf{X}$ is considered to be redundant or motionless, if all entries are 1 in the corresponding n^{th} column of motion mask; otherwise, if some parts of the entry are 0, then the frame is considered as dynamic and it is augmented in matrix $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_c] \in \mathbb{R}^{p\times c}$. More details can be found in [1].

3.3 Spatiotemporal Regularization

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 $\mathbf{D}$, 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 frames connected with similar pixel values most likely are part of $\mathbf{B}$. Therefore a segmentation that is temporally consistent with $\mathbf{B}$ can be obtained. For this purpose we find similarity between every two frames in the temporal direction. The graph $\mathbf{G}_t$ is then built using *s*-nearest neighbor strategy [28]. 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 $(\mathbf{d}_i\ ,\mathbf{d}_j) \in \mathbf{D}$ as $\Delta_{i,j} = \sqrt{||\mathbf{d}_i - \mathbf{d}_j||_2^2}$. Then, the adjacency matrix $\mathbf{A}_t$ for the $\mathbf{G}_t$ can be computed using

$$\mathbf{A}_t(i,j) = exp\left(-\frac{\Delta_{i,j} - \omega_{min}}{\sigma^2}\right), \tag{4}$$

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 $\left\{\mathbf{d}_i, \mathbf{d}_j\right\}$ 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 $\Phi_t = \mathbf{W}^{-1/2}(\mathbf{W}-\mathbf{A})\mathbf{W}^{-1/2} = \mathbf{I}-\mathbf{W}^{-1/2}\mathbf{A}\mathbf{W}^{-1/2}$, 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 $p \times p$.

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 $\mathbf{D}$. The pairwise relationship between the pixels is information that could alternatively be exploited to refine matrix $\mathbf{B}$ for the spatially consistent background modeling. For the construction of Laplacian of spatial graph, $\Phi_s \in \mathbb{R}^{p\times p}$, we enforce smoothness on the patch level of matrix $\mathbf{D}$ rather than on the pixel level, because comparing patches of the image allows one to use the local information of the image.

3.4 Solution for Model (3)

One of the main deficiency of model (3) is that it requires the computation of full (or partial) SVD of matrix $\mathbf{B}$ in every iteration, which could become prohibitively expensive when the dimensions are large. We overcome this problem by first adopting a proxy for the max-norm of rank of matrix $\mathbf{B}$ defined in [26] as:

$$||\mathbf{B}||_{\max} = \min_{\mathbf{U}\in\mathbb{R}^{p\times r},\mathbf{V}\in\mathbb{R}^{r\times c}} \frac{1}{2}(||\mathbf{U}||_{2,\infty}.||\mathbf{V}||_{2,\infty}),\ \text{such that},\ \mathbf{B} = \mathbf{U}\mathbf{V}. \tag{5}$$

where $\mathbf{U}$ is termed the spatial basis, and $\mathbf{V}$ represents the temporal coefficients of $\mathbf{U}$ in r-dimensional linear space, and r is a rank that is upper bounded by the rank of $\mathbf{B}$. The product $\mathbf{UV}$ is known as the approximation $\mathbf{B}$ of matrix $\mathbf{D}$. Taking this into account (5) can be substituted into (3) by considering that $||\mathbf{V}||_{2,\infty}^2 = 1$[26] as:

$$\min_{\mathbf{U}\in\mathbb{R}^{p\times r}\mathbf{V}\in\mathbb{R}^{r\times c}} ||\mathbf{D} - \mathbf{U}\mathbf{V}||_F^2 + \frac{1}{2}||\mathbf{U}||_{2,\infty}^2 + \gamma_1 tr(\mathbf{V}^\top\mathbf{U}^\top\Phi_s\mathbf{U}\mathbf{V}) + \gamma_2 tr(\mathbf{V}^\top\mathbf{U}^\top\Phi_t\mathbf{U}\mathbf{V}). \tag{6}$$

Model (6) is based on batch processing optimization in which all video frames have to be available in a memory prior to any processing. OS-RPCA is targeted to derive an

iterative solution of (6) over spatiotemporal graph regularizations which only processes one frame per time instance. To do so, the iterative solution of (6) is formulated as:

$$\min_{\mathbf{U}\in\mathbb{R}^{p\times r},\mathbf{v}_t}\sum_{t=1}^{c}\Big(||\mathbf{m}_t\circ(\mathbf{d}_t-\mathbf{U}\mathbf{v}_t)||_2^2+\gamma_1(\mathbf{v}_t^\top\mathbf{U}^\top\Phi_s\mathbf{U}\mathbf{v}_t)+\gamma_2(\mathbf{v}_t^\top\mathbf{U}^\top\Phi_t\mathbf{U}\mathbf{v}_t)\Big)+\frac{\lambda_1}{2}||\mathbf{U}||_{2,\infty}^2. \tag{7}$$

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. The detail solution for solving (7) can be found in [1].

4 Experimental Evaluations

We evaluated the OS-RPCA algorithm on new challenging RGB-D dataset [25]. OS-RPCA algorithm is compared with four current state-of-the-art methods 2P-RPCA [14], DECOLOR [19], 3TD [17], and BS-GFL [20]. The implementation of all these methods is publicly available on authors websites. OS-RPCA algorithm requires the following parameters, $\lambda_1 = 1/\sqrt{\max(p,n)}$ [24], γ_1 and γ_2 are set to 10 (see Fig.1 for more details). For the construction of $\mathbf{G}_s$ on image patches, we used patch size of 5×5 pixels. The number of nearest neighbors s is set 10 for both graphs. We used the FLANN [28] libraries for more efficient computation of the graphs. More tuned values of these parameters may generate better results, however, we have emphasized on generalization of the proposed algorithm over unseen datasets.

The RGB-D dataset consists of 35 diverse video sequences with 7 different categories (see Fig.2). The RGB-D dataset comprises of complex scenes in which the goal is to show that the integration of depth features enhance the performance of moving object detection. The visual results over seven selected sequences (one sequence per category) with corresponding ground truth images are illustrated in Fig.2. Fig.2 shows that the estimated mask of moving objects is close to the ground truth images.

We also quantitatively compared OS-RPCA algorithm results with existing methods. For this purpose, we computed a well-known $\mathbf{F}_1$ score as an accuracy measure. The goal is to maximize the $\mathbf{F}_1$ measure for more accurate moving object detection. Table 1 presents the comparison of the OS-RPCA algorithm with other methods using only intensity features. Table 2 presents the comparison of the OS-RPCA algorithm with other methods using intensity, RGB, depth, and RGBD features. On the average, the OS-RPCA algorithm achieved 0.75, 0.71, and 0.72 $\mathbf{F}_1$ score which is significantly higher than existing best performing method 2PRPCA which obtained 0.62 and 0.65.

For 'Illumination Changes' category, only OS-RPCA and 2PRCPA produced good results (see Table 1). While remaining methods such as DECOLOR and BS-GFL etc. perform well with some small discrepancies. This is because the nuclear norm constraints the low-rank background model to capture the background appearance. The additional spatio-temporal constraints encoded in OS-RPCA assisted to achieve good results.

The 'Color Camouflage' and 'Depth Camouflage' are more challenging sequences than 'Illumination Changes' category. In these videos, the moving objects are very close

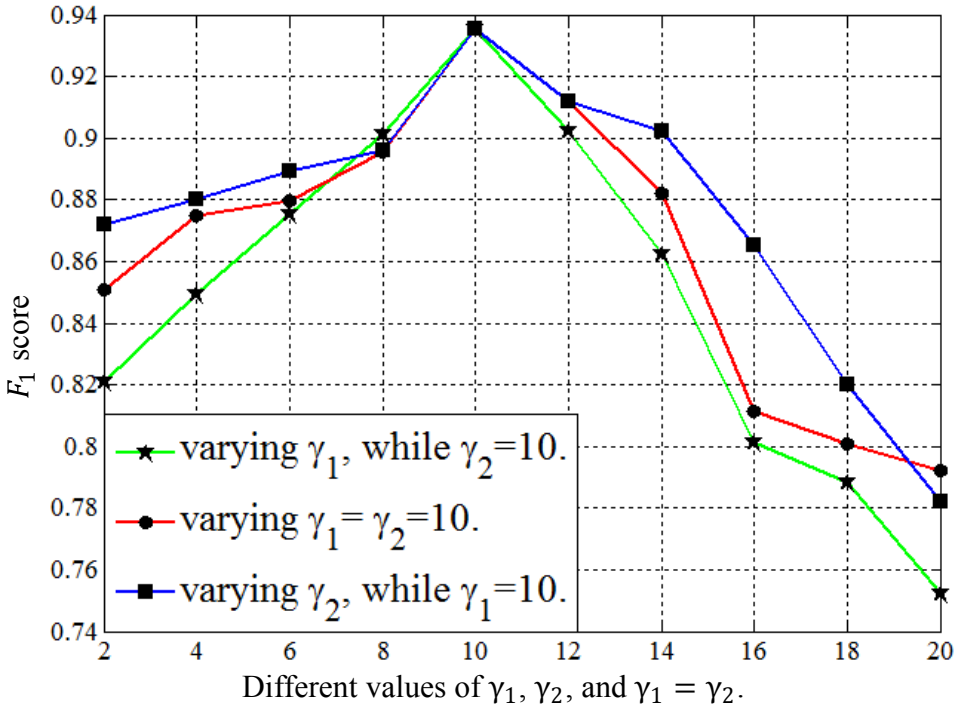

Fig. 1. Variations in $\mathbf{F}_1$ score using different values of γ_1 and γ_2. Experiment performed on 'genSeq1' sequence of RGB-D dataset [25]. The best results are obtained by using $\gamma_1 = \gamma_2 = 10$.

Table 1. Comparison of average $\mathbf{F}_1$ measure score on RGB-D dataset [25]. See Fig. 2 for visual comparisons with other methods. Note that the $\mathbf{F}_1$ score is computed using only intensity features. The first and second best performing methods are shown in red and blue colors.

Methods	Illumination Changes	Color Camouflage	Depth Camouflage	Intermittent Motion	Out of Sensor Range	Shadows	Bootstrapping	Average
DECOLOR	**0.48**	0.70	0.72	0.46	0.58	**0.62**	**0.76**	0.61
2PRPCA	0.38	**0.74**	0.68	**0.51**	**0.74**	0.61	0.69	**0.62**
3TD	0.20	0.66	0.62	0.38	0.66	0.59	0.71	0.54
BS-GFL	0.29	0.73	**0.75**	0.40	0.71	0.58	0.57	0.57
OS-RPCA	**0.44**	**0.85**	**0.81**	**0.77**	**0.80**	**0.77**	**0.81**	**0.75**

in color and depth to the background component. Table 1 shows that the OS-RPCA obtained, on the average, the higher $\mathbf{F}_1$ score. In these cases, the existing methods performed poor discrimination between background and foreground segments. Overwhelming outliers of color saturated foreground objects were incorporated into the estimated background. In contrast, the spatio-temporal constraints in OR-RPCA efficiently handled these outliers and estimated outlier-free low-rank component.

In 'Intermittent Motion' category, some foreground objects remained motionless and then started moving created outliers in the background model. OR-RPCA considered the motionless frames as redundant and removed them. All of the existing methods

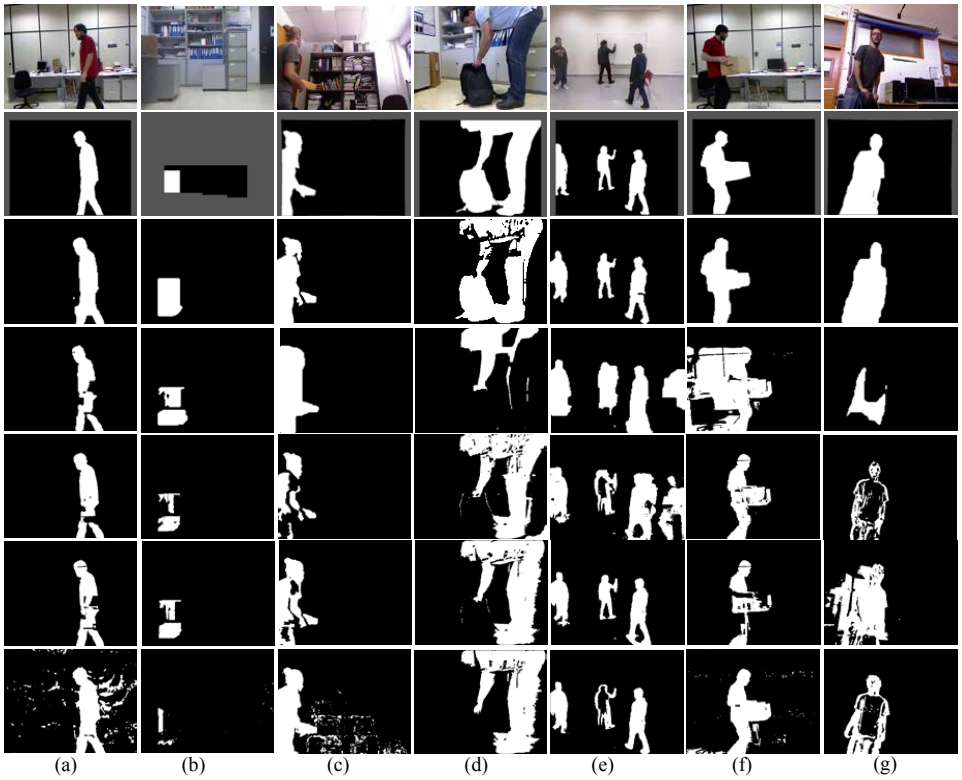

Fig. 2. Visual results of the OS-RPCA algorithm [1] in comparison with four state-of-the-art methods on RGB-D dataset [25]. Results are shown for only intensity features because of space limitation. From left to Right: (a) sequence 'genSeq1' from category 'Illumination Changes', (b) 'colorCam1y' from 'Color Camouflage', (c) 'Despatx_ds' from 'Depth Camouflage', (d) 'moved-Background2' from 'Intermittent Motion', (e) 'MultiPeople2' from 'Out of Sensor Range', (f) 'genSeq2' from 'Shadows', and (g) 'Bootstrapping_ds' from 'Bootstrapping'. From top to bottom: seven input images, ground truth images, the results obtained by OS-RPCA algorithm, DECOLOR [19], 2PRPCA [14], 3TD [17], and BS-GFL [20].

failed to cope with incorporated outliers because of motionless foreground objects. Only OS-RPCA algorithm effectively handled these outliers in the final estimation of background. The proposed OS-RPCA algorithm exhibited best performance ($\mathbf{F}_1 = 0.77$).

For 'Out of Sensor Range' sequences, most of the methods performed poor estimation of moving objects except 2P-RPCA. In this case, OS-RPCA have shown significantly better performance than existing methods. In all of these sequences, the performance of the moving object detection is improved by incorporating spatio-temporal constraints. For sequences that belong to the 'Shadows' category, the proposed OS-RPCA algorithm exhibited significant improvements ($\mathbf{F}_1 = 0.77$) over current methods. The current methods have shown degraded performance of $\mathbf{F}_1 < 0.70$. For complex dynamics in 'Bootstrapping' category, all methods have shown relatively poor per-

Table 2. Comparison of average $\mathbf{F}_1$ measure score on RGB-D dataset [25]. See Fig. 2 for visual comparisons with other methods. The $\mathbf{F}_1$ score is computed using intensity, RGB, Depth, and RGB+Depth (RGBD) features. The first and second best performing methods are shown in red and blue colors.

Methods	Intensity features	Depth features	RGB features	RGBD features
DECOLOR	0.61	0.20	0.56	0.63
2PRPCA	**0.62**	**0.29**	**0.65**	**0.65**
3TD	0.54	**0.31**	0.57	0.59
BS-GFL	0.57	0.21	0.63	0.60
OS-RPCA	**0.75**	0.24	**0.71**	**0.72**

formance. This is because of the absence of foreground-free images. The OS-RPCA algorithm performed significantly better than others by obtaining $\mathbf{F}_1 = 0.81$, while all others are 0.76, 0.69, 0.71, and 0.57, respectively.

Execution time of all algorithms was compared on a machine with 3:0 GHz Intel core i5 processor and 4GB RAM. For fair comparison with the above mentioned approaches, the time is recorded in seconds. It takes about 22s to process a 90 video frames with resolution of $[240 \times 320]$. While, DECOLOR [19] takes 505s, 2PRPCA [14] takes 180s, and BS-GFL [20] takes about 488s, respectively.

5 Conclusion

In this paper, we evaluated the performance of OS-RPCA algorithm on RGB-D dataset using color, intensity, and depth features for moving object detection. OS-RPCA is semi-online algorithm which processes one video frame per time instance. Experimental evaluations showed that the OS-RPCA performed better than other state of the art methods using intensity features. In the future, we will design an efficient fusion strategy to combine the depth and color features for more accurate moving object detection system.

References

1. Javed, S., Mahmood, A., Bouwmans, T., Jung, S.K.: Spatiotemporal low-rank modeling for complex scene background initialization. IEEE T-CSVT **PP** (2016)
2. Wang, S., Yang, J., Zhao, Y., Cai, A., Li, S.Z.: A surveillance video analysis and storage scheme for scalable synopsis browsing. In: IEEE ICCV-Workshops. (2011) 1947–1954
3. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: A survey. ACM CSUR **41** (2009) 15
4. Peng, H., Li, B., Ling, H., Hu, W., Xiong, W., Maybank, S.J.: Salient object detection via structured matrix decomposition. IEEE T-PAMI **39** (2017) 818–832
5. Kang, L.W., Lin, C.W., Fu, Y.H.: Automatic single-image-based rain streaks removal via image decomposition. IEEE T-IP **21** (2012) 1742–1755

6. Newson, A., Almansa, A., Fradet, M., Gousseau, Y., Pérez, P.: Video inpainting of complex scenes. SIAM J-IS **7** (2014) 1993–2019
7. Yilmaz, A., Javed, O., Shah, M.: Object tracking: A survey. ACM CSUR **38** (2006) 13
8. Bouwmans, T., El Baf, F., Vachon, B.: Background modeling using mixture of gaussians for foreground detection-a survey. RPCS **1** (2008) 219–237
9. Bouwmans, T., Maddalena, L., Petrosino, A.: Scene background initialization: A taxonomy. PRL (2017)
10. Pierre-Marc, J., Lucia, M., Alfredo, P.: Scene Background Modeling.net. In: IEEE ICPR. (2016)
11. Cao, X., Yang, L., Guo, X.: Total Variation Regularized RPCA for Irregularly Moving Object Detection Under Dynamic Background. IEEE T-Cybernetics **46** (2016) 1014–1027
12. Chen, M., Wei, X., Yang, Q., Li, Q., Wang, G., Yang, M.H.: Spatiotemporal gmm for background subtraction with superpixel hierarchy. IEEE T-PAMI **PP** (2017)
13. Diego, O., Juan, C.S.M., Jose, M.M.: Rejection based multipath reconstruction for background estimation in video sequences with stationary objects. CVIU **147** (2016) 23–37
14. Gao, Z., Cheong, L.F., Wang, Y.X.: Block-sparse RPCA for salient motion detection. IEEE T-PAMI **36** (2014) 1975–1987
15. Javed, S., Ho Oh, S., Sobral, A., Bouwmans, T., Ki Jung, S.: Background subtraction via superpixel-based online matrix decomposition with structured foreground constraints. In: IEEE ICCVW. (2015)
16. Liu, X., Zhao, G., Yao, J., Qi, C.: Background subtraction based on low-rank and structured sparse decomposition. IEEE T-IP **24** (2015) 2502–2514
17. Oreifej, O., Li, X., Shah, M.: Simultaneous video stabilization and moving object detection in turbulence. IEEE T-PAMI **35** (2013) 450–462
18. Staglianò, A., Noceti, N., Verri, A., Odone, F.: Online space-variant background modeling with sparse coding. IEEE T-IP **24** (2015) 2415–2428
19. Zhou, X., Yang, C., Yu, W.: Moving object detection by detecting contiguous outliers in the low-rank representation. IEEE T-PAMI **35** (2013) 597–610
20. Xin, B., Tian, Y., Wang, Y., Gao, W.: Background subtraction via generalized fused lasso foreground modeling. In: IEEE CVPR. (2015)
21. Bouwmans, T., Zahzah, E.H.: Robust PCA via principal component pursuit: a review for a comparative evaluation in video surveillance. CVIU **122** (2014) 22–34
22. Bouwmans, T., Sobral, A., Javed, S., Jung, S.K., Zahzah, E.H.: Decomposition into Low-rank plus additive matrices for background/foreground separation: A review for a comparative evaluation with a large-scale dataset. CSR (2016)
23. Wright, J., Ganesh, A., Rao, S., Peng, Y., Ma, Y.: Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization. In: NIPS. (2009)
24. Candès, E.J., Li, X., Ma, Y., Wright, J.: Robust principal component analysis? JACM **58** (2011) 11
25. Camplani, M., Maddalena, L., Gabriel, M.A., Petrosino, A., Salgado, L.: RGB-D dataset: Background learning for detection and tracking from rgbd videos. In: IEEE ICIAP-Workshops. (2017)
26. Lee, J.D., Recht, B., Srebro, N., Tropp, J., Salakhutdinov, R.R.: Practical large-scale optimization for max-norm regularization. In: NIPS. (2010) 1297–1305
27. Liu, C., et al.: Beyond pixels: exploring new representations and applications for motion analysis. PhD thesis, MIT (2009)
28. Muja, M., Lowe, D.G.: Scalable nearest neighbor algorithms for high dimensional data. IEEE T-PAMI **36** (2014) 2227–2240