Efficient Low-Rank Matrix Factorization Based on $\ell_{1,\varepsilon}$-Norm for Online Background Subtraction

Qi Liu, *Member, IEEE*, and Xiaopeng Li

***Abstract*— Background subtraction refers to extracting the foreground from an observed video, and is the fundamental problem of various applications. There are two kinds of popular methods to deal with background separation, namely, robust principal component analysis (RPCA) and low-rank matrix factorization (LRMF). Nevertheless, the drawback of RPCA requires tuning penalty parameter to attain an ideal result. Compared with RPCA, the ℓ_1-norm based LRMF does not involve extra parameters tuning, but it is challenging to optimize the ℓ_1-norm based minimization because of the nonsmooth ℓ_1-norm. In addition, it becomes time-consuming to find the optimal solution. In this work, we propose to employ smooth $\ell_{1,\epsilon}$-norm, an approximation of ℓ_1-norm, to tackle background subtraction. Thus, the proposed model inherits the superiority of LRMF and even becomes tractable. Then the resultant optimization problem is solved by alternating minimization and gradient descent where the step-size of the gradient descent is adaptively updated via backtracking line searching approach. The proposed method is proved to be locally convergent. Experimental results on synthetic and real-world data demonstrate that our method outperforms the state-of-the-art algorithms in terms of reconstruction loss, computational speed and hardware performance.**

***Index Terms*— Background subtraction, online subspace learning, low-rank matrix factorization.**

I. Introduction

BACKGROUND subtraction aims to separate moving objects from the background in the video streams (a.k.a., high-dimensional tensor) collected from real scenes, which dwells in a diversified range of applications, including urban traffic detection [1], long-term video monitoring [2], automatic vascular enhancement [3] and social signal processing [4].

One popular and feasible method to tackle the background separation task is batch robust principal component analysis (RPCA) [5]–[8], that is, a variant of PCA [9]. The RPCA is formulated as low-rank and ℓ_0-norm minimization problem. Unfortunately, it is NP-hard as the rank and ℓ_0-norm functions are nonconvex and discrete. In [10], [11], nuclear norm and ℓ_1-norm replace them to deal with background separation since nuclear norm and ℓ_1-norm are the convex envelops of the rank function and ℓ_0-norm [12], [13], respectively. However, the nuclear norm is the sum of all singular values, hence it may cause the solution to deviate seriously from the ground truth. To handle this issue, [14] proposes a nonconvex surrogate function to approximate the rank function, and then combines the ℓ_1-norm to enhance the accuracy of the background estimation. Recently, motivated by the plug-and-play framework, panoramic RPCA (P-RPCA) [15] considers the weighted anisotropic total variation (TV) to penalize the sparse component and employs an improved low-rank estimator "OptShrink" to deal with the resultant low-rank minimization. On the other hand, mixture of Gaussian (MoG) for low-rank matrix factorization (MoG-LRMF) [16] employs LRMF to handle background subtraction and exploits a specific MoG distribution to model the foreground. Compared with the RPCA based methods, MoG-LRMF does not require the penalty parameters. Other variants of RPCA also include [17], [18], details can refer to the review reference [19].

It is worth noting that the mentioned methods above belong to offline strategy (viz., batch LRMF/RPCA), processing the whole video to obtain the foreground. The video data, in practice, comprise a mass of frames, and thus the dimensions of the constructed matrix are very huge. Hence, these approaches may be infeasible due to the limited memory storage on mobile and wearable devices. Besides, they may not meet the real-time requirement for large scaled videos.

An online scheme is necessary to tackle background subtraction, which handles the incoming data of each frame on-the-fly without storing the whole video in memory. Some online methods are developed based on RPCA, e.g., OR-PCA [20], truncated nuclear norm based online RPCA (TNN-ORPCA) [21], online nonconvex lp-norm based RPCA (OLP-RPCA) [22], Grassmannian robust adaptive subspace tracking algorithm (GRASTA) [23], transformed GRASTA (T-GRASTA) [24], online RPCA via stochastic optimization (RPCA-STOC) [25] and online moving window RPCA (OMWRPCA) [26]. GRASTA and RPCA-STOC employ Frobenius norm to minimize the loss function and thus they are not robust to outliers. To be robust to outliers, ℓ_1-norm, in T-GRASTA, is adopted to minimize the cost function. Besides, RPCA-STOC utilizes stochastic optimization to tackle the online background subtraction. In addition, OMWRPCA uses a sliding window to detect the same part between two frames, and then minimizes the loss function in the window. The online version of MoG-LRMF, online mixture of Gaussian matrix factorization (OMoGMF) [27], utilizes LRMF to deal with online background subtraction problem and achieves better performance.

In this work, we utilize $\ell_{1,\varepsilon}$-norm and LRMF to formulate the online background subtraction problem where the target matrix is represented by the product of two low-dimensional

Manuscript received September 20, 2021; revised November 10, 2021; accepted November 16, 2021. Date of publication November 19, 2021; date of current version July 5, 2022. This article was recommended by Associate Editor V. Velisavljevic. *(Corresponding author: Xiaopeng Li.)*

Qi Liu is with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117583 (e-mail: elelqi@nus.edu.sg).

Xiaopeng Li is with the Department of Electrical Engineering, City University of Hong Kong, Hong Kong, SAR, China (e-mail: x.p.li@my.cityu.edu.hk).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2021.3129503.

Digital Object Identifier 10.1109/TCSVT.2021.3129503

matrices. The $\ell_{1,\varepsilon}$-norm is the smooth approximation of ℓ_1-norm, and thus the resulting model is tractable. Compared with the existing methods, the proposed algorithm takes the advantage of not requiring any penalty parameter tuning. Alternating minimization (AM) and gradient descent [28], [29] are employed to solve the resultant optimization problem. Moreover, experimental results demonstrate that our method is computationally attractive and hardware friendly.

The remainder of the letter is organized as follows. In Section II, we review two online background subtraction models. The proposed method is introduced in Section III. In addition, its convergence and computational complexity are analyzed. In Section IV, simulation and experimental results on synthetic and real-world data show that our method outperforms state-of-the-art methods. Finally, conclusions are drawn in Section V. The code is shared at https://sites.google.com/site/qiliucityu.

II. Problem Statement

A. Notations

Scalars, vectors and matrices are represented by italic, bold lower-case and bold upper-case letters, respectively. For matrices, $\|\cdot\|_*$, equalling to the sum of all singular values, denotes nuclear norm. In addition, $\|\boldsymbol{M}\|_F^2 = \sum_{i=1}^{m}\sum_{j=1}^{n} m_{i,j}^2$ is Frobenius norm where $m_{i,j}$ is the (i, j) entry of $\boldsymbol{M}$. Besides, the jth column and ith row of $\boldsymbol{M}$ are denoted by $\boldsymbol{m}_j$ and $\boldsymbol{m}_i^T$, respectively. Moreover, $\|\cdot\|_1$ is the ℓ_1-norm for vectors and matrices, which is calculated using the sum of absolute value of all entries.

B. Online RPCA Framework

Consider a video with n frames, each frame can be converted into a vector with the size of m and thus the video data can be modeled as a matrix $\boldsymbol{M} \in \mathbb{R}^{m\times n}$. Inspired by that background motion caused by orthographic cameras lies in a low-rank subspace and pixels belonging to one trajectory tend to group together, it is able to decompose the given video trajectory matrix into low-rank (i.e., background) and sparse (i.e., foreground) matrices.

It is well known that RPCA enables to separate an observed matrix into a low-rank matrix and a sparse matrix [10], [11]. Thus, most of existing methods apply RPCA to solve BS, leading to:

$$\min_{\boldsymbol{L},\boldsymbol{S}} \|\boldsymbol{L}\|_* + \lambda_1\|\boldsymbol{S}\|_1, \quad \text{s.t. } \boldsymbol{L}+\boldsymbol{S}=\boldsymbol{M} \tag{1}$$

where $\boldsymbol{L}$ and $\boldsymbol{S}$ denote the low-rank and sparse matrices, respectively, and $\lambda_1 > 0$ is a penalty parameter to trade off the rank of $\boldsymbol{L}$ and the sparsity of $\boldsymbol{S}$. To tackle (1), RPCA-STOC suggests employing the penalty method to convert (1) into the following unconstrained problem [25]:

$$\min_{\boldsymbol{L},\boldsymbol{S}} \|\boldsymbol{L}\|_* + \lambda_1\|\boldsymbol{S}\|_1 + \frac{\lambda_2}{2}\|\boldsymbol{M}-\boldsymbol{L}-\boldsymbol{S}\|_F^2 \tag{2}$$

where $\lambda_2 > 0$ is a penalty parameter. Furthermore, it has been proved that the nuclear norm is upper bounded by [30]:

$$||\boldsymbol{L}||_* = \inf_{\boldsymbol{U},\boldsymbol{V}} \left\{ \frac{1}{2}\|\boldsymbol{U}\|_F^2 + \frac{1}{2}\|\boldsymbol{V}\|_F^2 : \boldsymbol{L}=\boldsymbol{U}\boldsymbol{V} \right\}. \tag{3}$$

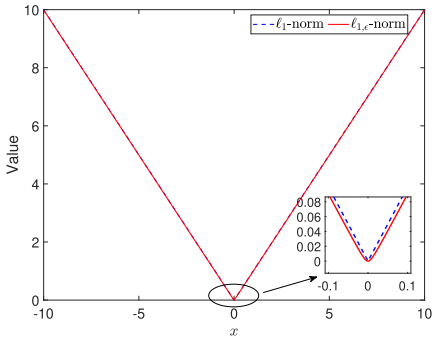

Fig. 1. Illustration of ℓ_1-norm and $\ell_{1,\epsilon}$-norm with $\epsilon = 1\times 10^{-2}$.

where $\boldsymbol{U} \in \mathbb{R}^{m\times r}$ and $\boldsymbol{V} \in \mathbb{R}^{r\times n}$ with r being the rank of the target matrix. Thus, (2) is equivalent to:

$$\min_{\boldsymbol{U},\boldsymbol{V},\boldsymbol{S}} \frac{1}{2}(\|\boldsymbol{U}\|_F^2 + \|\boldsymbol{V}\|_F^2) + \lambda_1\|\boldsymbol{S}\|_1 + \frac{\lambda_2}{2}\|\boldsymbol{M}-\boldsymbol{U}\boldsymbol{V}-\boldsymbol{S}\|_F^2 \tag{4}$$

where $\boldsymbol{U}$ can be considered as the basis for low-rank subspace and $\boldsymbol{V}$ represents the coefficients of observations w.r.t. $\boldsymbol{U}$. To meet the online requirement, the online RPCA computes the loss function on each frame, resulting in:

$$\min_{\boldsymbol{U},\boldsymbol{v}_j,\boldsymbol{s}_j} \frac{1}{2}(\|\boldsymbol{U}\|_F^2 + \|\boldsymbol{v}_j\|_2^2) + \lambda_1\|\boldsymbol{s}_j\|_1 + \frac{\lambda_2}{2}\|\boldsymbol{m}_j-\boldsymbol{U}\boldsymbol{v}_j-\boldsymbol{s}_j\|_F^2 \tag{5}$$

where $\boldsymbol{v}_j$, $\boldsymbol{s}_j$ and $\boldsymbol{m}_j$ are the jth column of $\boldsymbol{V}$, $\boldsymbol{S}$ and $\boldsymbol{M}$, respectively. It is worth noting that the online RPCA based performance is determined by these two penalty parameters, namely λ_1 and λ_2. In practice, tuning these two parameters is time-consuming and it is challenging to determine for different observed videos.

C. Online LRMF Framework

MoG-LRMF proposes to formulate the background subtraction problem as:

$$\min_{\boldsymbol{U},\boldsymbol{V}} \|\boldsymbol{M}-\boldsymbol{U}\boldsymbol{V}\|_{L_p} \tag{6}$$

where $\|\cdot\|_{L_p}$ denotes Frobenius norm or ℓ_1-norm. After seeking $\boldsymbol{U}$ and $\boldsymbol{V}$, the foreground is reconstructed via $\boldsymbol{S}=\boldsymbol{M}-\boldsymbol{U}\boldsymbol{V}$. L_p norm is convex, and thus it is easy to find $\boldsymbol{U}$ and $\boldsymbol{V}$. Nevertheless, note that $\boldsymbol{S}$ is not sparse when Frobenius norm is utilized to minimize (6), which produces an unclear foreground due to the noise corruptions. In contrast, ℓ_1-norm is able to obtain a sparse $\boldsymbol{S}$ but is a nonsmooth function, leading to intractable optimization. On the other hand, OMoGMF improves (6) and proposes online LRMF as:

$$\min_{\boldsymbol{U},\boldsymbol{v}_j} \|\boldsymbol{m}_j-\boldsymbol{U}\boldsymbol{v}_j\|_{L_p}. \tag{7}$$

Compared with the online RPCA, OMoGMF only requires a prior rank information.

III. Proposed Method

To avoid the intractable optimization caused by ℓ_1-norm, we suggest adopting $\ell_{1,\epsilon}$-norm, defined as:

$$\|\boldsymbol{X}\|_{1,\epsilon}^1 = \sum_{i=1}^{m}\sum_{j=1}^{n}\left((x_{i,j}^2+\epsilon^2)^{1/2}-\epsilon\right) \tag{8}$$

where $\epsilon > 0$ enables $\ell_{1,\epsilon}$-norm to be smooth. Compared with ℓ_1-norm, the $\ell_{1,\epsilon}$-norm has two crucial properties, as shown in Fig. 1: (i) $\ell_{1,\epsilon}$-norm with $\epsilon > 0$ is smooth and convex. (ii) $\lim_{\epsilon \to 0} \ell_{1,\epsilon}$-norm $= \ell_1$-norm.

Based on the $\ell_{1,\epsilon}$-norm, we formulate the background subtraction problem as:

$$\min_{\boldsymbol{X}} ||\boldsymbol{X} - \boldsymbol{M}||_{1,\epsilon} + \|\boldsymbol{X}\|_* \tag{9}$$

According to (3), (9) is modeled as:

$$\min_{\boldsymbol{U},\boldsymbol{V}} ||\boldsymbol{U}\boldsymbol{V} - \boldsymbol{M}||_{1,\epsilon}^1 + \frac{1}{2}||\boldsymbol{U}||_F^2 + \frac{1}{2}||\boldsymbol{V}||_F^2. \tag{10}$$

Based on the online strategy, (10) on the jth frame can be rewritten as:

$$\min_{\boldsymbol{U},\boldsymbol{v}_j} ||\boldsymbol{U}\boldsymbol{v}_j - \boldsymbol{m}_j||_{1,\epsilon}^1 + \frac{1}{2}||\boldsymbol{U}||_F^2 + \frac{1}{2}||\boldsymbol{v}_j||_2^2. \tag{11}$$

Following the rationale of alternating minimization (AM), we solve (11) via the following iterative procedure:

$$\begin{aligned} \boldsymbol{v}_j^{t+1} &= \arg\min_{\boldsymbol{v}_j} h(\boldsymbol{v}_j) \\ &= \arg\min_{\boldsymbol{v}_j} ||\boldsymbol{U}^t\boldsymbol{v}_j - \boldsymbol{m}_j||_{1,\epsilon}^1 + \frac{1}{2}||\boldsymbol{v}_j||_2^2. \end{aligned} \tag{12}$$

$$\begin{aligned} \boldsymbol{U}^{t+1} &= \arg\min_{\boldsymbol{U}} f(\boldsymbol{U}) \\ &= \arg\min_{\boldsymbol{U}} ||\boldsymbol{U}\boldsymbol{v}_j^{t+1} - \boldsymbol{m}_j||_{1,\epsilon}^1 + \frac{1}{2}||\boldsymbol{U}||_F^2. \end{aligned} \tag{13}$$

Since both (12) and (13) are convex and smooth, gradient descent is utilized to address them. Hence, the solution to $\boldsymbol{v}_j^{t+1}$ is computed by

$$\boldsymbol{v}_j^{k+1} = \boldsymbol{v}_j^k - \rho\nabla h(\boldsymbol{v}_j^k) \tag{14}$$

where $\rho > 0$ is the step size, and $\nabla h(\boldsymbol{v}_j) = (\boldsymbol{U}^t)^T\boldsymbol{d} + \boldsymbol{v}_j$. Herein, the ith entry of $\boldsymbol{d}$ is $d_i = ((\boldsymbol{u}_i^t)^T\boldsymbol{v}_j - (\boldsymbol{m}_j)_i)^2 + \epsilon^2)^{-0.5}((\boldsymbol{u}_i^t)^T\boldsymbol{v}_j - (\boldsymbol{m}_j)_i)$ with $\boldsymbol{u}_i^T$ and $(\boldsymbol{m}_j)_i$ being the ith row of $\boldsymbol{U}^T$ and ith entry of $\boldsymbol{m}_j$, respectively. When $\boldsymbol{v}_j^{k+1}$ converges, it becomes the solution to $\boldsymbol{v}_j^{t+1}$. Similarly, $\boldsymbol{U}^{t+1}$ is calculated by

$$\boldsymbol{U}^{k+1} = \boldsymbol{U}^k - \rho\nabla f(\boldsymbol{U}^k) \tag{15}$$

where $\nabla f(\boldsymbol{U}) = \boldsymbol{g}(\boldsymbol{v}_j^{t+1})^T + \boldsymbol{U}$ with $g_i = (\boldsymbol{u}_i^T\boldsymbol{v}_j^{t+1} - (\boldsymbol{m}_j)_i)^2 + \epsilon^2)^{-0.5}(\boldsymbol{u}_i^T\boldsymbol{v}_j^{t+1} - (\boldsymbol{m}_j)_i)$. When $\boldsymbol{U}^{k+1}$ converges, $\boldsymbol{U}^{t+1}$ is founded, that is, $\boldsymbol{U}^{t+1} = \boldsymbol{U}^{k+1}$. The background of the jth frame is $\boldsymbol{l}_j = \boldsymbol{U}^{t+1}\boldsymbol{v}_j^{t+1}$ and thus the foreground of the jth frame is $\boldsymbol{s}_j = \boldsymbol{m}_j - \boldsymbol{l}_j$. The proposed method is termed as online background subtraction using $\ell_{1,\epsilon}$-AM (OBSL1), summarized in Algorithm 1, where the ϵ is set at 0.01.

For ρ, we adopt backtracking line searching approach [28], [31] to adaptively update it. Consider a general convex function $g(\boldsymbol{x})$, in the $(k+1)$th iteration, the process to determine ρ is based on:

$$g(\boldsymbol{x}^k - \rho\nabla g(\boldsymbol{x}^k)) > g(\boldsymbol{x}^k) - \frac{\rho}{2}\|\nabla g(\boldsymbol{x}^k)\|_F^2, \tag{16}$$

which requires $\rho \leftarrow \beta\rho$. That is, ρ is determined as the value with $g(\boldsymbol{x}^k - \rho\nabla g(\boldsymbol{x}^k)) \leq g(\boldsymbol{x}^k) - \frac{\rho}{2}\|\nabla g(\boldsymbol{x}^k)\|_F^2$. It is suggested that ρ starts with 1, and $\beta = 0.8$ [28]. Moreover, the stopping criteria for both $t-$ and $k-$ loops are $\|\boldsymbol{v}^{k+1} - \boldsymbol{v}^k\|_2^2/\|\boldsymbol{v}^k\|_2^2 \leq 10^{-5}$ and $\|\boldsymbol{U}^{k+1} - \boldsymbol{U}^k\|_F^2/\|\boldsymbol{U}^k\|_F^2 \leq 10^{-5}$, respectively.

Algorithm 1 OBSL1 for Background Subtraction

Input: Video matrix $\boldsymbol{M} \in \mathbb{R}^{m\times n}$ and rank r
Initialize: Randomize $\boldsymbol{U}^1 \in \mathbb{R}^{m\times r}$
for $j = 1 : n$ **do**
  for $t = 1, 2, \cdots$ **do**
    for $k = 1, 2, \cdots$ **do**
      (1) $\boldsymbol{v}_j^{k+1} = \boldsymbol{v}_j^k - \rho\nabla h(\boldsymbol{v}_j^k)$
      Stop if stopping criterion is met.
    end for
    (2) $\boldsymbol{v}_j^{t+1} = \boldsymbol{v}_j^{k+1}$
    for $k = 1, 2, \cdots$ **do**
      (3) $\boldsymbol{U}^{k+1} = \boldsymbol{U}^k - \rho\nabla f(\boldsymbol{U}^k)$
      Stop if stopping criterion is met.
    end for
    (4) $\boldsymbol{U}^{t+1} = \boldsymbol{U}^{k+1}$
    Stop if stopping criterion is met.
  end for
  (5) $\boldsymbol{l}_j = \boldsymbol{U}^{t+1}\boldsymbol{v}_j^{t+1}$
end for
Output: $\boldsymbol{L} = [\boldsymbol{l}_1, \cdots, \boldsymbol{l}_n]$ and $\boldsymbol{S} = \boldsymbol{M} - \boldsymbol{L}$

For OBSL1, the complexities for calculating $\boldsymbol{g}$ and $\boldsymbol{d}$ are $\mathcal{O}(mr)$. To calculate $\nabla f(\boldsymbol{U}^k)$ and $\nabla h(\boldsymbol{v}_j^k)$, the complexity is dominated by $\mathcal{O}(mr^2)$. Furthermore, the complexities for updating $\boldsymbol{U}^{k+1}$ and $\boldsymbol{v}_j^{k+1}$ are $\mathcal{O}(Bmr^2)$ and thus the total computational complexity of OBSL1 is $\mathcal{O}(TKBmnr^2)$ where T, K and B are the maximum iteration of the AM, gradient descent and backtacking line searching, respectively. In our experiments, T, K and B are set to 2, 1000 and 10, respectively. Note that $\mathcal{O}(TKBmnr^2)$ is to perform the whole video while the complexity of each frame is $\mathcal{O}(TKBmr^2)$.

To facilitate analyzing the convergence of OBSL1, we first define a loss function as:

$$\mathcal{L}(\boldsymbol{U}, \boldsymbol{v}_j) = ||\boldsymbol{U}\boldsymbol{v}_j - \boldsymbol{m}_j||_{1,\epsilon}^1 + \frac{1}{2}||\boldsymbol{U}||_F^2 + \frac{1}{2}||\boldsymbol{v}_j||_2^2, \tag{17}$$

then prove that $\mathcal{L}(\boldsymbol{U}, \boldsymbol{v}_j)$ is nonincreasing. We have:

$$\begin{aligned} &\mathcal{L}(\boldsymbol{U}^{t+1}, \boldsymbol{v}_j^{t+1}) - \mathcal{L}(\boldsymbol{U}^t, \boldsymbol{v}_j^t) \\ &= \mathcal{L}(\boldsymbol{U}^t, \boldsymbol{v}_j^{t+1}) - \mathcal{L}(\boldsymbol{U}^t, \boldsymbol{v}_j^t) + \mathcal{L}(\boldsymbol{U}^{t+1}, \boldsymbol{v}_j^{t+1}) - \mathcal{L}(\boldsymbol{U}^t, \boldsymbol{v}_j^{t+1}). \end{aligned} \tag{18}$$

Since $\boldsymbol{v}_j^{k+1}$ is the optimal solution of $\min_{\boldsymbol{v}_j} h(\boldsymbol{v}_j)$, we obtain $\mathcal{L}(\boldsymbol{U}^t, \boldsymbol{v}_j^{t+1}) - \mathcal{L}(\boldsymbol{U}^t, \boldsymbol{v}_j^t) \leq 0$. Besides, $\mathcal{L}(\boldsymbol{U}^{t+1}, \boldsymbol{v}_j^{t+1}) - \mathcal{L}(\boldsymbol{U}^t, \boldsymbol{v}_j^{t+1}) \leq 0$ must hold as $\boldsymbol{U}^{t+1}$ is the global solution of $\min_{\boldsymbol{U}} f(\boldsymbol{U})$ because of the convexity. Thus, we get $\mathcal{L}(\boldsymbol{U}^{t+1}, \boldsymbol{v}_j^{t+1}) - \mathcal{L}(\boldsymbol{U}^t, \boldsymbol{v}_j^t) \leq 0$, implying that $\mathcal{L}(\boldsymbol{U}, \boldsymbol{v}_j)$ is nonincreasing. Moreover, $\mathcal{L}(\boldsymbol{U}, \boldsymbol{v}_j)$ is lower bounded by 0. Given that minimizing $\mathcal{L}(\boldsymbol{U}, \boldsymbol{v}_j)$ is a nonconvex (viz., bi-convex) problem, OBSL1 is thus locally convergent.

IV. Experimental Results

This section describes experiments that are conducted on both synthetic, CDnet 2014 [32], and BMC 2012 [33] benchmark databases to validate the effectiveness and efficiency of the proposed method.

A. Synthetic Data

Following the scheme from [26] and [34], the simulated video data is modeled. For each frame, the background

TABLE I

COMPARISON OF THE PROPOSED METHOD WITH THE EXISTING ONLINE AND OFFLINE REPRESENTATIVES IN TERMS OF MSE VALUES

Algorithm	**OBSL1**	OMWRPCA [26]	OMoGMF [27]	T-GRASTA [24]	P-RPCA [15]
MSE	1.309×10^{-4}	6.250×10^{-3}	1.932×10^{-4}	7.662×10^{-3}	1.263×10^{-4}

TABLE II

CPU RUNTIME COMPARISON AMONG DIFFERENT METHODS

Dataset \ Algorithm	**OBSL1**	OMWRPCA [26]	OMoGMF [27]	T-GRASTA [24]	P-RPCA [15]
Crossroad	**3.42s**	35.38s	2.62s	74.95s	8.88s
People	**4.12s**	135.07s	3.55s	115.24s	9.90s
Car	**3.23s**	69.35s	2.60s	73.55s	8.68s
PETS	**3.95s**	103.60s	3.42s	143.67s	9.97s
Park	**3.44s**	175.95s	3.39s	112.11s	8.34s
Railway	**4.23s**	272.61s	4.16s	116.88s	10.33s
Station	**5.56s**	354.88s	5.52s	152.28s	17.68s

$\boldsymbol{A} \in \mathbb{R}^{200\times200}$ is generated by a random matrix whose entries satisfy the standard Gaussian distribution and then its ith singular value is set to 2^{10-i}. This distribution of the singular values performs good approximation of image background. In addition, the moving object is represented by a random matrix $\boldsymbol{B} \in \mathbb{R}^{50\times25}$. The mimetic video includes 100 frames and the moving object is randomly embedded into each background. Based on the aforementioned setting, we can acquire $\boldsymbol{M} \in \mathbb{R}^{40000\times100}$ and $\boldsymbol{S} \in \mathbb{R}^{40000\times100}$, which contain the backgrounds and foregrounds of all frames, respectively. Herein, mean squared error (MSE) is utilized as the evaluation metric, namely:

$$\text{MSE} = \|\boldsymbol{S} - \hat{\boldsymbol{S}}\|_F^2/(mn) \tag{19}$$

where $\hat{\boldsymbol{S}}$ denotes the estimate of $\boldsymbol{S}$.

The proposed algorithm is compared with four existing online and offline representatives, viz., OMWRPCA [26], OMoGMF [27], T-GRASTA [24] and P-PRCA [15], as tabulated in Table I. It is seen that our method outperforms OMWRPCA, OMoGMF and T-GRASTA with smaller performance loss, and yet slightly is inferior to P-RPCA. It is worth noting that, however, P-RPCA is an offline background subtraction method.

B. Real-World Data

We evaluate the CPU runtime among compared algorithms, as listed in Table II. We observe that the elapsed time of the proposed method is much less than those of OMWRPCA, T-GRASTA and P-PRCA, and OMoGMF performs the best in computational time, which is partially attributed to the proposed online MoG modeling. Additionally, the reconstruction performance on CDnet 2014 and BMC 2012 are illustrated in Fig. 2 and Fig. 3, respectively. we can see that the rebuilt backgrounds and foregrounds by the proposed algorithm on different videos are more visually clear than OMWRPCA, OMoGMF and T-GRASTA, and comparable with P-RPCA. Nevertheless, the offline P-RPCA requires full video data to achieve satisfactory result. On the other hand, although the proposed method runs slower than OMoGMF, the reconstruction performance of our scheme is much better than that of OMoGMF.

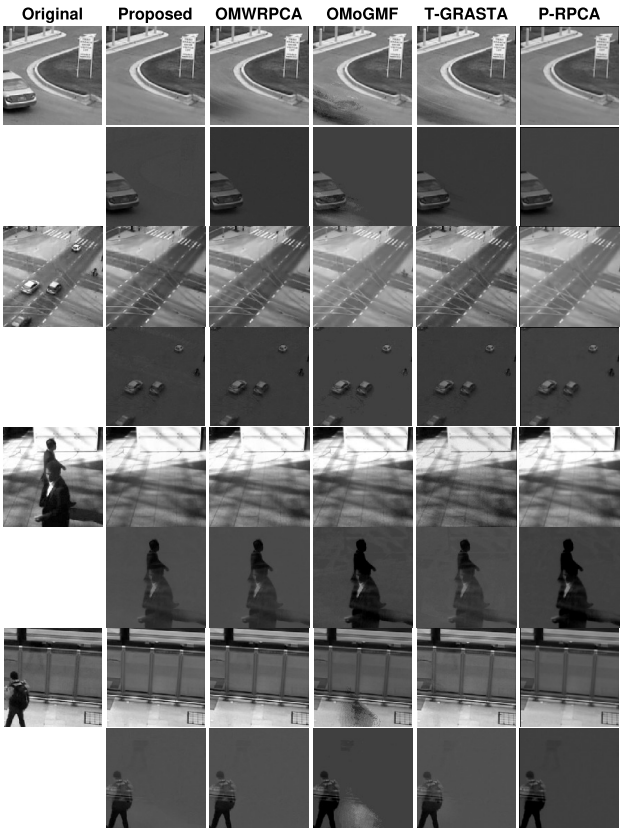

Fig. 2. Comparison results on CDnet 2014 dataset among different background subtraction competitors.

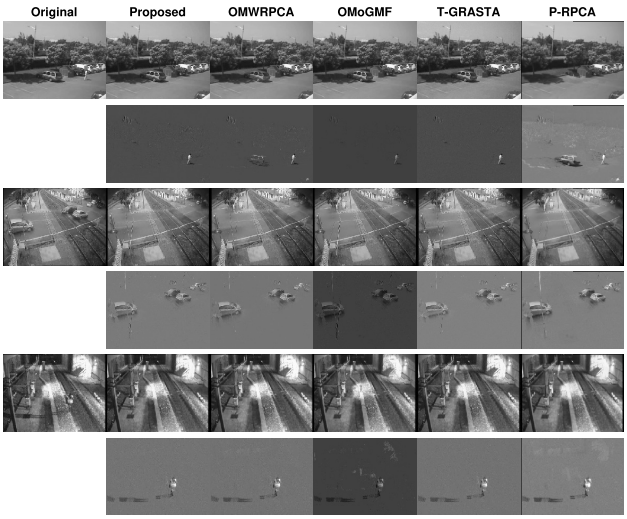

Fig. 3. Comparison results on BMC 2012 dataset among different background subtraction competitors.

As tabulated in Table III, we evaluate the workloads of different approaches in terms of floating point operations per second (Flops) on a Linux system, where the Flops is a measure of hardware performance used for comparing the peak theoretical performance of a system. Herein, we calculate the Flops via Perf,[1] which is an event-oriented observability tool for Linux 2.6+ based systems that abstracts away graphics

[1] http://www.brendangregg.com/perf.html

TABLE III
COMPUTER PERFORMANCE EVALUATION AMONG DIFFERENT APPROACHES IN TERMS OF FLOPS

Algorithm	**OBSL1**	OMWRPCA [26]	OMoGMF [27]	T-GRASTA [24]	P-RPCA [15]
Flops	**4.32M**	5.51M	4.07M	7.53M	4.95M

processing unit (GPU) hardware differences in Linux performance measurements. From Table III, we observe that the number of the proposed method is around 4.32 million (viz., 4.32M), which is more efficient than other competitors, and yet slightly inferior to OMoGMF.

V. CONCLUSION

In this work, on the basis of proposed $\ell_{1,\epsilon}$-norm minimization, a novel online background subtraction method, named as OBSL1, is developed. Compared with the state-of-the-art algorithms, the proposed method does not require any penalty parameters tuning and enjoys simple implementation. Meanwhile, the proposed method shows its effectiveness and efficiency on synthetic and real-world data in terms of MSE metric, CPU runtime and hardware Flops computation, as corroborated by indicative empirical results.

REFERENCES

[1] C. Zhao and A. Basu, "Dynamic deep pixel distribution learning for background subtraction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 11, pp. 4192–4206, Nov. 2020.

[2] A. W. Senior, Y. Tian, and M. Lu, "Interactive motion analysis for video surveillance and long term scene monitoring," in *Proc. Comput. Vis. ACCV Workshops*, R. Koch and F. Huang, Eds. Berlin, Germany: Springer, 2011, pp. 164–174.

[3] S. Song *et al.*, "Spatio-temporal constrained online layer separation for vascular enhancement in X-ray angiographic image sequence," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 10, pp. 3558–3570, Oct. 2020.

[4] A. Vinciarelli, M. Pantic, and H. Bourlard, "Social signal processing: Survey of an emerging domain," *Image Vis. Comput.*, vol. 27, no. 12, pp. 1743–1759, Nov. 2009.

[5] T. Bouwmans, S. Javed, H. Zhang, Z. Lin, and R. Otazo, "On the applications of robust PCA in image and video processing," *Proc. IEEE*, vol. 106, no. 8, pp. 1427–1457, Aug. 2018.

[6] S. Javed, A. Mahmood, S. Al-Maadeed, T. Bouwmans, and S. K. Jung, "Moving object detection in complex scene using spatiotemporal structured-sparse RPCA," *IEEE Trans. Image Process.*, vol. 28, no. 2, pp. 1007–1022, Feb. 2018.

[7] S. Javed, S. Oh, A. Sobral, T. Bouwmans, and S. Jung, "OR-PCA with MRF for robust foreground detection in highly dynamic backgrounds," in *Proc. ACCV*, 2014, pp. 284–299.

[8] W. Song, J. Zhu, Y. Li, and C. Chen, "Image alignment by online robust PCA via stochastic gradient descent," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 7, pp. 1241–1250, Jul. 2016.

[9] I. T. Jolliffe, *Principal Component Analysis*. New York, NY, USA: Springer, 1986.

[10] P. Rodriguez and B. Wohlberg, "Fast principal component pursuit via alternating minimization," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2013, pp. 69–73.

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

[12] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

[13] M. Fazel, "Matrix rank minimization with applications," Ph.D. dissertation, Dept. Elect. Eng., Stanford Univ., Stanford, CA, USA, 2002.

[14] Z. Kang, C. Peng, and Q. Cheng, "Robust PCA via nonconvex rank approximation," in *Proc. IEEE Int. Conf. Data Mining*, Nov. 2015, pp. 211–220.

[15] B. E. Moore, C. Gao, and R. R. Nadakuditi, "Panoramic robust PCA for foreground–background separation on noisy, free-motion camera video," *IEEE Trans. Comput. Imag.*, vol. 5, no. 2, pp. 195–211, Jun. 2019.

[16] D. Meng and F. De la Torre, "Robust matrix factorization with unknown noise," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1337–1344.

[17] S. Javed, A. Mahmood, T. Bouwmans, and S. K. Jung, "Spatiotemporal low-rank modeling for complex scene background initialization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 6, pp. 1315–1329, Jun. 2018.

[18] S. Javed, A. Mahmood, T. Bouwmans, and S. K. Jung, "Background–foreground modeling based on spatiotemporal sparse subspace clustering," *IEEE Trans. Image Process.*, vol. 26, no. 12, pp. 5840–5854, Dec. 2017.

[19] T. Bouwmans, A. Sobral, S. Javed, S. K. Jung, and E.-H. Zahzah, "Decomposition into Low-rank plus additive matrices for background/foreground separation: A review for a comparative evaluation with a large-scale dataset," *Comput. Sci. Rev.*, vol. 23, pp. 1–71, Feb. 2016.

[20] S. Javed, T. Bouwmans, and S. K. Jung, "Depth extended online RPCA with spatiotemporal constraints for robust background subtraction," in *Proc. 21st Korea-Japan Joint Workshop Frontiers Comput. Vis. (FCV)*, Jan. 2015, pp. 1–6.

[21] B. Hong, L. Wei, Y. Hu, D. Cai, and X. He, "Online robust principal component analysis via truncated nuclear norm regularization," *Neurocomputing*, vol. 175, pp. 216–222, Jan. 2016.

[22] K. G. Quach, C. N. Duong, K. Luu, and T. D. Bui, "Non-convex online robust PCA: Enhance sparsity via ℓ_p-norm minimization," *Comput. Vis. Image Understand.*, vol. 158, pp. 126–140, May 2017.

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

[24] J. He, D. Zhang, L. Balzano, and T. Tao, "Iterative online subspace learning for robust image alignment," in *Proc. 10th IEEE Int. Conf. Workshops Autom. Face Gesture Recognit. (FG)*, Apr. 2013, pp. 1–8.

[25] J. Feng, H. Xu, and S. Yan, "Online robust PCA via stochastic optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 26, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds. Red Hook, NY, USA: Curran Associates, 2013, pp. 404–412.

[26] W. Xiao, X. Huang, F. He, J. Silva, S. Emrani, and A. Chaudhuri, "Online robust principal component analysis with change point detection," *IEEE Trans. Multimedia*, vol. 22, no. 1, pp. 59–68, Jan. 2020.

[27] H. Yong, D. Meng, W. Zuo, and L. Zhang, "Robust online matrix factorization for dynamic background subtraction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 7, pp. 1726–1740, Jul. 2018.

[28] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[29] Z.-Q. Luo and W. Yu, "An introduction to convex optimization for communications and signal processing," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1426–1438, Aug. 2006.

[30] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM Rev.*, vol. 52, no. 3, pp. 471–501, 2010.

[31] L. Armijo, "Minimization of functions having Lipschitz continuous first partial derivatives," *Pacific J. Math.*, vol. 16, pp. 1–3, Nov. 1966.

[32] Y. Wang, P.-M. Jodoin, F. Porikli, J. Konrad, Y. Benezeth, and P. Ishwar, "CDnet 2014: An expanded change detection benchmark dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2014, pp. 393–400.

[33] A. Vacavant, L. Tougne, T. Chateau, and L. Robinault, "Evaluation of background models with synthetic and real data," *Background Modeling and Foreground Detection for Video Surveillance*. New York, NY, USA: Chapman & Hall, Aug. 2014, pp. 1–25.

[34] Z. Wang, M.-J. Lai, Z. Lu, W. Fan, H. Davulcu, and J. Ye, "Orthogonal rank-one matrix pursuit for low rank matrix completion," *SIAM J. Sci. Comput.*, vol. 37, no. 1, pp. A488–A514, Jan. 2015.