Recursive Low-rank and Sparse Recovery of Surveillance Video using Compressed Sensing*

Shuangjiang Li, Hairong Qi
Electrical Engineering and Computer Science
University of Tennessee, Knoxville, TN 37996
{shuangjiang, hqi}@utk.edu

ABSTRACT

This paper focuses on surveillance video processing using Compressed Sensing (CS). The CS measurements are used for recovery of the video frame into a low-rank background component and sparse component that corresponds to the moving object. The spatial and temporal low-rank features of the video frame, e.g., the nonlocal similar patches within the single video frame and the low-rank background component residing in multiple frames, are successfully exploited. We propose rLSDR that consists of three major components. First we develop an efficient single frame CS recovery algorithm, called NLDR, that operates on the nonlocal similarity patches within each frame to solve the low-rank optimization problem with the CS measurements constraint using Douglas-Rachford splitting method. Second, after obtaining a few NLDR recovered frames as training, a fast bilateral random projections (BRP) scheme is adopted for quick low-rank background initialization. Third, rLSDR then incorporates real-time single video frame to recursively recover the sparse component and update the background, where the proposed NLDR algorithm can also be used here for sparse component estimation. Experimental results on standard surveillance videos demonstrate that NLDR performs the best for single frame CS recovery compared with the state-of-the-art and rLSDR could successfully recover the background and sparse object with less resource consumption.

Keywords

Compressed sensing, low-rank approximation, sparse recovery, surveillance video processing

1. INTRODUCTION

Smart Camera Networks (SCNs) have been traditionally used in surveillance and security applications [20], where a plural of cameras are deployed and networked with each other through wireless connections. The cameras transmit surveillance videos to a processing center where the videos are processed and analyzed. Of particular interest in surveillance video processing is the ability to detect anomalies and moving objects in a scene automatically and quickly [14].

Detection of moving objects is a well-established problem that has received a great deal of attention from the research community [22, 25]. Classical techniques often involve performing background subtraction, object segmentation, and sequential estimation for the objects of interest [21]. Another approach is based on low-rank and sparse modeling [4], where the background is modeled by a low rank matrix, and the moving objects are identified by a sparse component (e.g., [27, 1]). These methods require all pixels of surveillance video to be captured, transmitted and analyzed.

However, due to the growing availability of cheap, high-quality cameras, the amount of data generated by the video surveillance system has grown drastically. The challenge arises on how to process, store or transmit such enormous amount of data under real-time and bandwidth constraints [24]. At the same time, most of the data is uninteresting due to inactivity (e.g., background). There is a high risk of the network being overwhelmed by the mostly uninteresting data that prevents timely detection of anomalies and moving objects [14]. Thus, it is imperative for SCNs to transmit a small amount of data with enough information for reliable detection and tracking of moving objects or anomalies. The theory of Compressed Sensing (CS) [11, 3] allows us to address this problem. Under certain conditions related to sparse representations of video frames, CS can effectively reduce the amount of data collected by the system while retaining the ability to faithfully reconstruct the information of interest.

When applying CS on the surveillance video acquisition, the CS measurements are transmitted to the processing center. The original pixel values of the video frame are unknown, and therefore, the traditional background subtraction [21], low-rank and sparse modeling [27, 1] cannot be applied directly. A direct approach is to recover the video frame first and then apply the traditional techniques. The drawbacks are two-fold: First, the CS recovery algorithm does not take advantage of special characteristics of surveillance video in which a well defined, relatively static background exists [14]. Second, in many applications, one would like to quickly obtain the background and object estimates on-the-fly, rather

*This work was supported in part by the National Science Foundation under grant NSF CNS-1017156.

ICDSC '14 November 04 - 07 2014, Venezia Mestre, Italy

ACM 978-1-4503-2925-5/14/11 $15.00.
http://dx.doi.org/10.1145/2659021.2659029

than in a batch fashion, it is also desirable to incorporate real-time sample-by-sample (i.e., streaming) frame to update the recovery result.

In this paper, we propose a method named **rLSDR** (recursive Low-rank and Sparse estimation through Douglas-Rachford splitting) for segmentation of background by recursively estimating low-rank and sparse components in the reconstructed surveillance video frames from CS measurements. As in [4], the low-rank component is the background, and the sparse component identifies moving objects. In this method, First, we propose an algorithm named **NLDR** (NonLocal Douglas-Rachford splitting) to solve the single frame CS recovery problem. NLDR takes advantage of self-similarities within the single frame and models it as a low-rank matrix. An efficient algorithm based on Douglas-Rachford splitting (DR) is proposed to solve the low-rank optimization problem with the CS measurements constraint. Second, after obtaining a few NLDR recovered frames as training, a fast bilateral random projections (BRP) scheme is adopted for quick low-rank background initialization. Third, we propose a scheme to recursively estimate the low-rank background part and sparse object part in a "frame-by-frame" fashion, where the proposed NLDR algorithm can also be used for sparse component estimation.

The rest of the paper is organized as follows. Section 2 presents some related work on background subtraction, low-rank and sparse modeling. Section 3 discusses the problem formulation. Section 4 introduces the proposed rLSDR algorithm. The performance evaluation on three videos is given in Section 5. Finally, we conclude in Section 6.

2. RELATED WORK

In [18], the authors first proposed to use Principal Component Analysis (PCA) to model the background. Object detection is then achieved by thresholding the difference between generated background image and the current image. PCA provides a robust model of the probability distribution function of the background, but not the moving objects [1]. The work in [9, 10] improved classical PCA with respect to outlier and noise, yielding the field of robust PCA. Later on, this was advanced by very recent works based on the idea that the data matrix X can be decomposed into two components such that $X = L + S$, where L is a low-rank matrix and S is a matrix that is sparse. This decomposition can be obtained by robust Principal Component Analysis (rPCA) solved via Principal Component Pursuit (PCP) [23, 4]. While PCP is an elegant solution, it suffers some practical limitations. First, it requires the number of nonzero pixels in the moving objects to be small (i.e., the object should be exact sparse), this may not hold if there are large size or multiple moving objects. Second, PCP is a batch method and computationally expensive, it would be more useful to quickly obtain the low-rank matrix and the sparse matrix in an incremental way for each new frame and gradually improve the estimates.

The bandwidth challenge in the network of surveillance cameras was addressed by CS. The author in [14] proposed to recovery the CS measurements into low-rank and sparse components and adopted the alternative direction method (ADM) for solving the optimization problem in a batch fashion. ReProCS [19], an algorithm that addressed the limitation in PCP by recursively projecting the CS recovered frame to the subspace perpendicular to the subspace spanned by the PC component to nullify the background. It then recovers the sparse component by solving a noisy CS problem. Although robust and can be implemented on-the-fly, it needed to acquire the high accurate estimation of background PC component (e.g., through PCA) to successfully nullify the low-rank part in the data matrix. The performance could easily be affected by the training process to obtain the PC component.

3. PROBLEM FORMULATION

We consider a video sequence consisting of a number of frames (i.e., images). Let $x_t \in \mathbb{R}^{m \times n}$ be a vector formed from pixels of frame t of the video sequence, for $t = 1, \cdots, T$, where T is the total number of frames, m and n are the dimensions of each frame. The current frame x_t, is an overlay of foreground image, F_t, over the background image, B_t. The goal is to recover both F_t and B_t at each time frame t in real-time. Many foreground pixels are zero and hence F_t is a sparse matrix. We let T_t denote the foreground support set, i.e., $T_t := \{i : (F_t)_i \neq 0\}$. Thus,

$$(x_t)_i := \begin{cases} (F_t)_i & \text{if } i \in T_t \\ (B_t)_i & \text{otherwise} \end{cases} \tag{1}$$

where i is the entry indices corresponding to the raster scan order in the data matrix.

Let Φ_t be an $M \times N$ measurement matrix, where $M < N$. The measurement matrix Φ_t may be chosen as a random Gaussian or Fourier Scrambled matrices [3]. In this paper, we choose the Φ_t as a sparse binary measurement matrix based on the expander graph [15, 16] which serves the same purpose as the traditional CS measurement matrices but further reduces the computation (e.g., only addition operations on cameras) and the amount of measurements transmitted.

Assume, each frame can be re-arranged as an $N \times 1$ vector (i.e., $N = m \times n$). The single frame CS measurements from the video are defined as

$$y_t = \Phi_t x_t \tag{2}$$

where y_t is a vector of length M. To recover x_t from y_t, first y_t is sparsely coded with respect to the basis $\Psi \in \mathbb{R}^{N \times N}$ by solving the following minimization problem

$$\hat{\alpha} = \arg\min_{\alpha} \{\|y_t - \Phi_t \Psi \alpha\|_2^2 + \lambda_\alpha \|\alpha\|_1\} \tag{3}$$

and then x_t is reconstructed by $\hat{x}_t = \Psi \hat{\alpha}$.

Following the same notation in [19]. Let μ_t denote the mean background image and let $L_t := B_t - \mu_t$, and $M_t := \hat{x}_t$ be the frame t reconstructed from CS recovery algorithm (e.g., [8]) with mean subtracted. By defining

$$(S_t)_i := \begin{cases} (F_t - B_t)_i = (F_t - \mu_t - L_t)_i & \text{if } i \in T_t \\ 0 \quad \text{otherwise} \end{cases} \tag{4}$$

we can formulate as a problem of recovering L_t and S_t from

$$M_t := S_t + L_t \tag{5}$$

Here, S_t is a sparse vector with support set, T_t, and L_t are dense matrices lie in a slowly changing low dimensional subspace.

4. THE PROPOSED ALGORITHM

The proposed rLSDR algorithm consists of three major components: (1) single frame recovery from CS measurement, (2) low-rank component initialization, and (3) recursive recovery of sparse component and update of the low-rank component.

4.1 Single Frame Recovery

The Iterative Soft Thresholding (IST) algorithm [8] can be very efficient in solving the problem in Eq. (3), where the processing is conducted in a block-by-block manner through the same CS measurement operator. However, dividing the frame into blocks and treating each block as an independent sub-CS recovery task would inevitably lose some global properties of the frame. Thus, we propose NLDR that takes advantage of the nonlocal similar patches across the entire frame to better recover the single frame under CS measurements.

4.1.1 Nonlocal Similarity Patches

Buades et al. introduced in [2] the *nonlocal means* approach to image denoising, where the self-similarities are used as a prior on natural images. The basic idea of nonlocal means (NL) filtering is simple. For a given pixel u_i in a video frame x_t, its NL filtered new intensity value, denoted by $\mathsf{NL}(u_i)$, is obtained as a weighted average of its neighborhood pixels within a search window of size w.

In our work, we extend the pixel-wise nonlocal filtering to the patch-based filtering. Specifically, for a single video frame x_t, we search for the nonlocal similar patches $p_{i,j}, j = 1, 2, \cdots, q$, to the given patch p_i in a large window of size w centered at pixel u_i. Here, q is the total number of similar patches to be selected. The weight of patch $p_{i,j}$ to p_i, denoted by ω_{ij}, is then computed by

$$\omega_{ij} = \frac{1}{c_i}\exp\left(\frac{-\|p_i - p_{i,j}\|_2^2}{h^2}\right),\ j = 1, \cdots, q \tag{6}$$

where h is a pre-determined scalar and c_i is the normalization factor. Accordingly, for each patch p_i, we have a set of its similar patches, denoted by Ω_i. Then the nonlocal estimates of each patch $\hat{p}_i$ can be computed as $\hat{p}_i = \sum_{j\in\Omega_i}\omega_{ij}p_{i,j}$. Further, this can be written in a matrix form as

$$\hat{x}_i \doteq \mathbf{W}\sum\hat{p}_i,\ \mathbf{W}(i,j) = \begin{cases}\omega_{ij}, & \text{if } x_j \in \Omega_i\\ 0, & \text{otherwise.}\end{cases} \tag{7}$$

where $\hat{x}_i$ is the nonlocal estimated single video frame output.

4.1.2 Patch Denoising by Low-rank Approximation

Although we can use Eq. (7) to remove noise in the IST recovered video frame $\hat{x}_t$ to a certain degree, this is based on a weighted average of patches in $\hat{x}_t$, which are inherently noisy. Thus, it is imperative to apply some denoising techniques before the nonlocal similar patches reweight using Eq. (7) to prevent the noise from accumulating. Formulating the nonlocal similarity patches into the matrix format, we have $B_i = [p_{i,1}; p_{i,2}; , \cdots ; p_{i,q}]$, where each column of B_i is a vector representation of $p_{i,j}, j = 1, 2, \cdots, q$ for patch p_i. Since all columns of B_i share the similarity with the patch p_i, the columns of B_i should bear a high degree of similarity between each other, and subsequently, we can safely treat B_i as a low-rank matrix. If we treat B_i as a noisy frame, then the denoising process can be conducted by low-rank approximation. We formulate the nonlocal patch denoising into the low-rank matrix approximation problem [5] as follows

$$\min_{\hat{B}_i}\frac{1}{2}\|B_i - \hat{B}_i\|_2^2 + \lambda_{B_i}\|\hat{B}_i\|_*, \tag{8}$$

where $\|\hat{B}_i\|_*$ is the nuclear norm of the low-rank approximated patch matrix $\hat{B}_i$ defined by $\|\hat{B}_i\|_* \triangleq \mathrm{trace}(\sqrt{\hat{B}_i^T\hat{B}_i}) = \sum_{r=1}^q \sigma_r$, where σ_r are the singular values of $\hat{B}_i$.

In addition, since the columns of B_i (or patches) are also a subset of the reconstructed video frame from IST recovery algorithm, it should be subjected to the CS measurement constraint $y_t = \Phi_t x_t$. Therefore, we formulate the denoising problem of Eq. (8) by

$$\min_{x_t}\frac{1}{2}\|x_t - \mathbf{W}B_i\|_2^2 + \lambda_{x_t}\|x_t\|_* \text{ s.t. } y_t = \Phi_t x_t. \tag{9}$$

We adopt the Douglas-Rachford splitting method to solve Eq. (9).

4.1.3 Douglas-Rachford Splitting

The Douglas-Rachford splitting method was originally proposed in [12] for solving matrix equations. Later on it was advanced as an iterative scheme to minimize the functions of the form, $\min_x F(x) + G(x)$, where F and G are convex functions for which one is able to compute the proximal mappings $\mathsf{prox}_{\gamma F}$ and $\mathsf{prox}_{\gamma G}$ which are defined as

$$\mathsf{prox}_{\gamma F}(x) = \arg\min_y \frac{1}{2}\|x - y\|_2^2 + \gamma F(y)$$

The same definition applies to $\mathsf{prox}_{\gamma G}$ [6]. In order to solve Eq. (8), we have $F(x) = \iota_{\mathcal{C}}(x)$ and $G(x) = \|x\|_*$, where $\mathcal{C} = \{x : y = \Phi x\}$ and $\iota_{\mathcal{C}}$ is the indicator function.

Given that $F(x) = \iota_{\mathcal{C}}(x)$, the solution to Eq. (10) is the same as projections onto convex sets (POCS), and does not depend on γ.

$$\mathsf{prox}_{\gamma\iota_{\mathcal{C}}F}(x) = \mathsf{prox}_{\iota_{\mathcal{C}}F}(x) = x + \Phi^+(y - \Phi x), \tag{10}$$

where $\Phi^+ = \Phi^T(\Phi\Phi^T)^{-1}$. The proximal operator of $G(x)$ is the soft thresholding of the singular values

$$\mathsf{prox}_{\gamma G}(x) = U(x)\cdot\rho_{\lambda_x}(S(x))\cdot V(x)^* \tag{11}$$

where $x = U\cdot S\cdot V^*$ is the singular value decomposition of the matrix x and $S = \mathrm{diag}(s_i)_i$ is the diagonal matrix of singular values s_i, and $\rho_{\lambda_x}(S)$ is defined as a diagonal operator.

$$\rho_\lambda(S) = \mathrm{diag}(\max(0, 1 - \lambda_x/|s_i|)s_i)_i \tag{12}$$

We can then solve the problem in Eq. (8) using the Douglas-Rachford iterations given by

$$\tilde{x}_{k+1} = (1 - \frac{\mu}{2})\tilde{x}_k + \frac{\mu}{2}\mathsf{rprox}_{\gamma G}(\mathsf{rprox}_{\gamma F}(\tilde{x}_k)) \tag{13}$$

and the $(k+1)$-th solution $\hat{x}_{k+1}$ is calculated by $\hat{x}_{k+1} = \mathsf{prox}_{\gamma F}(\tilde{x}_{k+1})$. Here the reversed-proximal mappings is given by $\mathsf{rprox}_{\gamma F} = 2\mathsf{prox}_{\gamma F} - x$ for $F(x)$ and $G(x)$ respectively. The parameters are selected as $\lambda_x > 0$ and $0 < \mu < 2$ which guarantee $\hat{x}$ to be a solution that minimizes $F(x) + G(x)$ based on the proof in [7].

4.1.4 The Proposed NLDR algorithm

Algorithm 1 provides a pseudo-code for the proposed Nonlocal Douglas-Rachford splitting (NLDR) algorithm. Given the measurements y_t, the NLDR algorithm first obtains an intermediate reconstruction result $\hat{x}_{IST}$ through the IST algorithm [8]. This soft-thresholding output is then used to calculate the nonlocal estimated frame $\hat{x}_{nl}$. The final nonlocal estimates is used to initialize the low-rank optimization problem in Eq. (8) where the Douglas-Rachford splitting method will be carried out iteratively based on Eq. (13).

Algorithm 1: NLDR Algorithm

Input:
- ▶ CS Measurement matrix $\Phi_t \in \mathbb{R}^{M\times N}$
- ▶ Basis matrix $\Psi \in \mathbb{R}^{N\times N}$
- ▶ Measurements $y_t \in \mathbb{R}^{M}$
- ▶ Number of the iterations iter.

Output:
- ▶ An estimate $\hat{x}_t \in \mathbb{R}^N$ of the original single frame x_t.

```
Obtain an initial recovery x̂_IST from IST [8]
Initialize x̂_nl ← x̂_IST
Calculate nonlocal weights ω_ij using Eq. (6)
Update x̂_nl ← W x̂_i using Eq. (7)
for k = 0, 1, 2, ··· , iter do
   Initialize x̃_0 ← x̂_nl
   Calculate x̃_{k+1} using Eq. (13)
end for
return x̂_t ← x̃_{k+1}
```

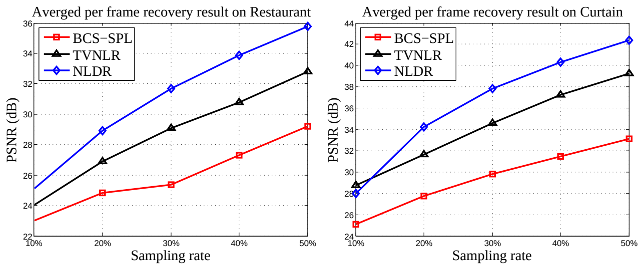
4.2 Low-rank Component Initialization

After denoising the CS recovered frame using NLDR, the second component of the proposed rLSDR algorithm is to estimate the low-rank background image based on the first few video frames (e.g., around 50). In order to estimate the background, a common approach would be applying SVD on the recovered video frames to obtain its low-rank approximation. However, performing SVD operation is usually very time-consuming, especially for large resolution video frames which hinders the "on-the-fly" estimation. The other drawback is that, often we just need a rough estimation of the low-rank component which can later be refined upon receiving new video frames.

In this work, we adopt the bilateral random projections (BRP) based low-rank approximation with closed-form solution. Given r bilateral random projections of a $p\times q$ dense matrix X (w.l.o.g, $p \geq q$), i.e., $U = XA_1$ and $V = X^T A_2$, where $A_1 \in \mathbb{R}^{q\times r}$ and $A_2 \in \mathbb{R}^{p\times r}$ are random matrices,

$$L = U(A_2^T U)^{-1} V^T \tag{14}$$

is a fast rank-r approximation of X. The L in Eq. (14) has been proposed in [13] as a recovery of a rank-r matrix X from U and V, where A_1 and A_2 are independent Gaussian or subsampled Fourier random matrices. It was later



(a) (b)

Figure 1: Averaged per frame recover result comparison on (a) **Restaurant** (b) **Curtain**.

advanced by Zhou et al. in [27] showed that L is a tight rank-r approximation to a full rank matrix X, when A_1 and A_2 are correlated random matrices updated from Y_2 and Y_1, respectively.

The computation of L includes an inverse of an $r\times r$ matrix and three matrix multiplications. Thus, for a dense X, $2pqr$ floating-point operations (flops) are required to obtain BRP, $r^2(2q+r)+pqr$ flops are required to obtain L. The computational cost is much less than SVD based approximation.

4.3 Recursive Sparse Recovery and Low-rank Updates

After the low-rank background component L_t has been estimated, we now proceed to the third component of rLSDR, where we recursively update the sparse component and background estimation upon receiving the CS measurements y_{t+1} of new frame x_{t+1}. The CS recovered new frame $\hat{x}_{t+1}$ is obtained using the proposed NLDR algorithm. The sparse recovery problem to find S_{t+1} can be formulated as follows

$$\begin{aligned} &\min_{S_{t+1}} \frac{1}{2}\|\hat{x}_{t+1} - L_t - S_{t+1}\|_2^2 + \lambda_s \|S_{t+1}\|_1 \\ &\text{s.t.} \quad \|y_{t+1} - \Phi_{t+1}(L_t + S_{t+1})\|_2^2 \leq \epsilon \end{aligned} \tag{15}$$

where L_t is estimated background at the frame t. The only unknown in Eq. (15) is S_{t+1}. Again it can be solved using NLDR algorithm to estimate $\hat{S}_{t+1}$.

After the sparse component is obtained using Eq. (15), the corresponding low-rank background component at $t{+}1$ frame can be calculated as $L_{t+1} = \hat{x}_{t+1} - \hat{S}_{t+1}$. This single frame background will be incorporated into Eq. (14) to update L, which is the initial trained background matrix. The final low-rank background at frame $t+1$ is then estimated as $\hat{L}_{t+1} = L(t+1)$ from the output of Eq. (14).

We summarize the proposed rLSDR in Algorithm 2.

5. EXPERIMENTAL RESULTS

We apply **rLSDR** to two surveillance videos [1], i.e., *Restaurant* and *Curtain*. *Curtain* consists of 304 frames each of dimension 64×80. *Restaurant* contains 200 frames with dimension 144×176. We first experiment on the single frame

[1] http://perception.i2r.a-star.edu.sg/bk_model/bk_index.html

Algorithm 2: rLSDR Algorithm

Input:

- CS Measurement matrix $\Phi_t \in \mathbb{R}^{M \times N}$
- Measurements data matrix $y_t \in \mathbb{R}^{M \times p}$
- Initialize random matrices A_1, A_2
- Number of training frames trn.

Output:

- CS recovered frames $\hat{x} \in \mathbb{R}^{N \times p}$,
- Background and object estimate $\hat{L}, \hat{S}$.

Step 1: Initial frame recovery
for $i = 1, \cdots, \mathsf{trn}$ do
  $X(1 : \mathsf{trn}) \leftarrow \mathbf{NLDR}(y_i)$
end for
Step 2: Background initialization
Estimate L using Eq. (14)
Step 3: Recursive update L and S
for $t = \mathsf{trn}, \cdots, \mathsf{p}$ do
  Frame recovery: $\hat{x}_{t+1} \leftarrow \mathbf{NLDR}(y_{t+1})$
  Sparse est.: Solve Eq. (15) for $\hat{S}_{t+1}$ using **NLDR**
  Calculate L_{t+1}: $L_{t+1} = \hat{x}_{t+1} - \hat{S}_{t+1}$, update Eq. (14)
  Background est.: $\hat{L}_{t+1} = L(t+1)$
end for
return $\hat{x}, \hat{L}, \hat{S}$

recovery result by comparing **NLDR** with two popular CS image recovery algorithms, BCS-SPL [17] and TVNLR [26].

The block-based image patch is of size 6×6. We set the number of similar patches q in the nonlocal estimation step as 45. We use the scrambled Fourier matrix as the CS measurement matrix Φ and DCT matrix as the basis Ψ to represent the original image in the initial IST recovery. The parameter is selected as $\mu = 1$ for DR iteration and $\lambda_f = \frac{c_i}{\max(s_i)}$ for each iteration where $c_i = C_0 * \epsilon, 0 < \epsilon < 1$ and C_0 is a constant.

Fig. 1 shows the averaged per frame recover result of NLDR compared with BCS-SPL and TVNLR using the PSNR metric. Generally, NLDR outperforms the state-of-the-art CS image recovery algorithm in the two video frames.

We then conduct experiments to compare the rLSDR on background and object estimation. For each video sequence, a number of frames, 150 for *Curtain* and 50 for *Restaurant*, are selected as the training frames to initialize the background.

Fig. 2 shows the CS recovered frame on *Restaurant* with background and object extracted. We also compare the result with **PCP** [4] and **ReProCS** [19] in Fig. 3 where the NLDR recovery video frames are used as the batch input. rLSDR could successful recover the background and the object and performs better than PCP, while having similar result as ReProCS. Compared with ReProCS, rLSDR requires much less initial training frames and thus less resource consumptions.

Figure 2: First column: original *Restaurant* video frames at t = 70, 116, 140. Second column: frame recovered by **NLDR** with 30% measurements. Next 2 columns: background and object estimated by **rLSDR**.

6. CONCLUSION

In this paper, we presented **rLSDR**, a CS-based surveillance video processing algorithm to recursively estimate the low-rank background and sparse object. The spatial and temporal low-rank features of the video frame were successfully exploited. Capitalized on the self-similarities within each spatial frame, we proposed **NLDR** for the single frame CS recovery that had high recovery PSNR under various sampling rates compared with the-state-of-art recovery algorithm. We proposed rLSDR that recursively estimates the background through efficient bilateral random projection (BPR). Experimental results on three standard surveillance videos showed that NLDR performs best for CS frame recovery and rLSDR could successfully recovery the background and sparse object with less resource consumption.

7. REFERENCES

[1] T. Bouwmans and E. H. Zahzah. Robust PCA via principal component pursuit: A review for a comparative evaluation in video surveillance. *Computer Vision and Image Understanding*, 122:22–34, 2014.

[2] A. Buades, B. Coll, and J.-M. Morel. A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation*, 4(2):490–530, 2005.

[3] E. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *Info. Theory, IEEE Trans. on*, 52(2):489–509, 2006.

[4] E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11, 2011.

[5] E. J. Candès and T. Tao. The power of convex relaxation: Near-optimal matrix completion. *Info. Theory, IEEE Trans. on*, 56(5):2053–2080, 2010.

[6] P. L. Combettes and J.-C. Pesquet. Proximal splitting methods in signal processing. In *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, pages 185–212. Springer, 2011.

[7] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale*

(a) Original (b) Recovered (c) **rLSDR** (d) **rLSDR** (e) **PCP** (f) **PCP** (g) **ReProCS** (h) **ReProCS**

Figure 3: First column: original *Curtain* video frames at t = 65, 103, 140. Second column: frame recovered by **NLDR** with 30% measurements. Next 6 columns: background and object estimated by **rLSDR**, **PCP** and **ReProCS**.

Modeling & Simulation, 4(4):1168–1200, 2005.

[8] I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on pure and applied mathematics*, 57(11):1413–1457, 2004.

[9] F. De la Torre and M. J. Black. Robust principal component analysis for computer vision. In *ICCV, 8th IEEE International Conference on*, pages 362–369. IEEE, 2001.

[10] F. De La Torre and M. J. Black. A framework for robust subspace learning. *International Journal of Computer Vision*, 54(1-3):117–142, 2003.

[11] D. Donoho. Compressed sensing. *Information Theory, IEEE Trans. on*, 52(4):1289–1306, 2006.

[12] J. Douglas and H. Rachford. On the numerical solution of heat conduction problems in two and three space variables. *Trans. of the American mathematical Society*, 82(2):421–439, 1956.

[13] M. Fazel, E. Candes, B. Recht, and P. Parrilo. Compressed sensing and robust recovery of low rank matrices. In *Signals, Systems and Computers, 2008 42nd Asilomar Conference on*, pages 1043–1047. IEEE, 2008.

[14] H. Jiang, W. Deng, and Z. Shen. Surveillance video processing using compressive sensing. *Inverse Problems & Imaging*, 6(2), 2012.

[15] S. Li and H. Qi. Distributed data aggregation for sparse recovery in wireless sensor networks. In *Proceedings of the 9th IEEE international conference on Distributed Computing in Sensor Systems*, pages 62–69, 2013.

[16] S. Li and H. Qi. Pattern-based compressed phone sensing. In *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE*, pages 169–172, 2013.

[17] S. Mun and J. E. Fowler. Block compressed sensing of images using directional transforms. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 3021–3024. IEEE, 2009.

[18] N. M. Oliver, B. Rosario, and A. P. Pentland. A bayesian computer vision system for modeling human interactions. *Pattern Analysis and Machine Intelligence, IEEE Trans. on*, 22(8):831–843, 2000.

[19] C. Qiu and N. Vaswani. ReProCS: A missing link between recursive robust PCA and recursive sparse recovery in large but correlated noise. *arXiv preprint arXiv:1106.3286*, 2011.

[20] B. Rinner and W. Wolf. An introduction to distributed smart cameras. *Proceedings of the IEEE*, 96(10):1565–1575, 2008.

[21] C. Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. In *CVPR, IEEE Computer Society Conference on.*, volume 2. IEEE, 1999.

[22] Y.-L. Tian, M. Lu, and A. Hampapur. Robust and efficient foreground analysis for real-time video surveillance. In *CVPR. IEEE Computer Society Conference on*, pages 1182–1187. IEEE, 2005.

[23] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma. Robust principal component analysis: Exact recovery of corrupted low-rank matrices by convex optimization. In *Proc. of Neural Information Processing Systems*, volume 3, 2009.

[24] C. Wu, H. Aghajan, and R. Kleihorst. Real-time human posture reconstruction in wireless smart camera networks. In *Proceedings of the 7th international conference on IPSN*, pages 321–331. IEEE Computer Society, 2008.

[25] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *ACM computing surveys*, 38(4):13, 2006.

[26] J. Zhang, S. Liu, R. Xiong, S. Ma, and D. Zhao. Improved total variation based image compressive sensing recovery by nonlocal regularization. In *Circuits and Systems (ISCAS), 2013 IEEE International Symposium on*, pages 2836–2839. IEEE, 2013.

[27] T. Zhou and D. Tao. Godec: Randomized low-rank & sparse matrix decomposition in noisy case. In *Proceedings of the 28th International Conference on Machine Learning (ICML)*, pages 33–40, 2011.