LETTER **Inequality-Constrained RPCA for Shadow Removal and Foreground Detection**

Hang LI^{†a}), Yafei ZHANG[†], Jiabao WANG[†], Yulong XU[†], Yang LI[†], Nonmembers, and Zhisong PAN[†], Member

SUMMARY State-of-the-art background subtraction and foreground detection methods still face a variety of challenges, including illumination changes, camouflage, dynamic backgrounds, shadows, intermittent object motion. Detection of foreground elements via the robust principal component analysis (RPCA) method and its extensions based on low-rank and sparse structures have been conducted to achieve good performance in many scenes of the datasets, such as Changedetection.net (CDnet); however, the conventional RPCA method does not handle shadows well. To address this issue, we propose an approach that considers observed video data as the sum of three parts, namely a row-rank background, sparse moving objects and moving shadows. Next, we cast inequality constraints on the basic RPCA model and use an alternating direction method of multipliers framework combined with Rockafeller multipliers to derive a closedform solution of the shadow matrix sub-problem. Our experiments have demonstrated that our method works effectively on challenging datasets that contain shadows.

key words: robust PCA, foreground detection, shadow removal, inequality constraint, ADMM

1. Introduction

Detecting moving objects and removing shadows continue to play fundamental roles in the analysis and understanding of video sequences. As a first step, most applications, including content-based video retrieval (CBVR), video surveillance and security technology, object tracking, motion capture, video compression, and scene understanding for robots, need to detect a moving object and subtract the background elements.

The robust principal component analysis (RPCA) method [6] has been applied in the above fields because of its robustness and ease of interpretability. A recent evaluation [3] demonstrated that some advanced low-rank subspace models based on RPCA have already made progress in different environments and scenes. Remarkably, the blocksparse RPCA [1] method, which emphasizes on the choice of the regularization parameter and successfully addresses the object scale problem, has outperformed many state-ofthe-art approaches when applied to the CDnet [9] dataset; however, this approach did not successfully process the shadow scene.

Besides the issue of object scale, we suggest that

a) E-mail: miipl606@163.com

DOI: 10.1587/transinf.2014EDL8234

shadow detection and removal will increasingly be a problem inherent in RPCA background subtraction. From a systematic evaluation of shadow detection [5], we conclude that extracting objects in a bounding box without shadow removal will result in 50% error pixels or worse in some scenes. Also, the error will propagate to subsequent phases, such as object recognition, tracking, and object retrieval, especially when applied to huge video stream databases. A recent idea for addressing the shadow problem is to use different sensors and near-infrared information [8], but most video streams are still in the visible spectrum.

To improve shadow detection performance while using RPCA method, we propose an intuitive method for decomposing the video matrix into three parts. Through casting an inequality constraint on the shadow matrix, we effectively separate the moving object matrix and shadow matrix in one step via the alternating direction method of multipliers (ADMM) framework.

2. **Inequality Constraint RPCA for Shadow Detection**

Due to similar patterns of moving objects and corresponding shadows, the RPCA methods introduced above face significant difficulty in extracting moving shadows without any constraints. Some constrained methods [4] have been shown to facilitate extraction and tracking of the object in an image sequence. In this section, we introduce a three-part inequality-constrained RPCA (ICRPCA) model and its algorithm.

Three-Part ICRPCA Modeling 2.1

Before decomposition, every image in a video stream with resolution $w \times h$ will first be reshaped into vector $d \in \mathbb{R}^m$, where $m = w \times h$. To process *n* frames in the video, we stack *n* vectorized image frames into observed matrix $D \in \mathbb{R}^{m \times n}$. Here, D can be decomposed into low-rank matrix A representing the background and two sparse outlier matrices Eand S that represent moving objects and shadows, respectively. Detecting moving objects and shadows is encapsulated by the following problem:

$$\min_{\substack{A,E,S\\ s.t.}} rank(A) + \lambda_1 ||E||_0 + \lambda_2 ||S||_0$$

$$D = A + E + S, \quad S \leq C.$$

$$(1)$$

Here, $\|\cdot\|_0$ represents the number of non-zero entries in a matrix. Given this problem, we need to assume that the rank of A is low. This is a weak assumption, i.e., most of the back-

1

Manuscript received November 18, 2014.

Manuscript revised January 26, 2015.

Manuscript publicized March 2, 2015.

[†]The authors are with the College of Command Information Systems, PLA University of Science and Technology (PLAUST), Nanjing, China.

ground matrix *A*'s columns are linearly correlated; however, the assumption is often valid since the background is often almost static. In (1), inequality constraint $S \leq C$ means that the pixel intensity of shadow matrix *S* might be less than the proper threshold *C*. Since we will not know the extract position of moving outliers, the inequality constraint is valid for all pixels in the frames.

Further, the convex relaxation of (1) is as follows:

$$\begin{array}{ll} \min_{A,E,S} & \|A\|_* + \lambda_1 \|E\|_1 + \lambda_2 \|S\|_1 \\ s.t. & D = A + E + S, \quad S \leqslant C. \end{array} \tag{2}$$

Here, $\|\cdot\|_*$ is the nuclear norm of a matrix, i.e., the sum of the singular values and $\|\cdot\|_1$ denotes L_1 norm of the given matrix. Further, λ_1 and λ_2 are regularization parameters.

The augmented Lagrangian function of (2) is defined as follows:

$$L(A, E, S, Y, R, \mu) = ||A||_{*} + \lambda_{1} ||E||_{1} + \lambda_{2} ||S||_{1} + \langle Y, D - A - E - S \rangle + \frac{\mu}{2} ||D - A - E - S||_{F}^{2} + \frac{1}{2\mu} (||P_{+}(R + \mu(S - C))||_{F}^{2} - ||R||_{F}^{2})$$
(3)

Here, the non-positive function $P_+(\cdot)$ is as follows:

$$P_+(u) = \begin{cases} u & u \ge 0, \\ 0 & u < 0. \end{cases}$$
(4)

Based on the augmented Lagrangian method (ALM), *Y* in (3) is the multiplier matrix and μ is the penalty factor. The \langle, \rangle operator calculates the inner product of two matrices; and $\|\cdot\|_F$ denotes Frobenius norm of a matrix. Therefore, the two terms in the second line of (3) convert the equality constraint into the Lagrangian function. Similarly, the term in the third line of (3) adds the inequality constraint via Rockafeller multiplier methods [10]. By introducing a slack variable and solving the minimization problem for the slack variable, the inequality constraint problem is naturally equivalent to the equality constraint problem. Thus, matrix *R* in (3) is the multiplier matrix of the inequality constraint term (its specific proofs and details are explained in [10]).

2.2 Algorithm Framework and Solution of Shadow Matrix

In this subsection, we solve the problem formulated in Eq. (3), i.e., the three-part ICRPCA. With the help of the ADMM framework and the Inexact ALM (IALM) described in [7], we develop our three-part ICRPCA via IALM as Algorithm 1.

The convergence of three-part ADMM is explained in [2], in which He et al. suggested that there is little difference between ADMMs with or without the use of Gaussian back substitution, and in practice, the process always achieves an O(1/k) convergence rate. Therefore, under the convergence guarantee, the next major step is to derive the closed-form solution of the sub-problem for shadow matrix *S*. According to (3), we can use $Z = D - A - E + Y/\mu$; then, the local first-order condition for variable *S* is as follows:

Algorithm 1 Three-part ICRPCA via IALM Input:

Matrix $D \in \mathbb{R}^{m \times n}, \lambda_1, \lambda_2;$

Output:

Estimates of E. S. A:

1: Initialization: $Y_0 = D/J(D)$, $E_0 = 0$, $S_0 = 0$, $\mu > 0$, $\rho > 1$, k = 0;

- 2: while not converged do
- 3: //Line 4 solves $E_{k+1} = arg \min_E L(A_k, E, S_k, Y_k, R_k, \mu_k)$ as (3)
- 4: $E_{k+1} = S_{\lambda_1/\mu_k} (D A_k S_k + Y_k/\mu_k);$
- 5: //Lines 6–7 solves $A_{k+1} = \arg \min_A L(A, E_{k+1}, S_{k+1}, Y_k, R_k, \mu_k)$
- 6: $(U, \Sigma, V) = svd(D \underbrace{E_k}_{r} S_k);$
- 7: $A_{k+1} = U \mathcal{S}_{1/\mu_k}(\Sigma) V^T;$
- 8: //Line 9 solves $S_{k+1} = \arg \min_{S} L(A_{k+1}, E_{k+1}, S, Y_k, R_k, \mu_k)$
- 9: $S_{k+1} = S_*(D A_{k+1} E_{k+1} + Y_k/\mu_k);$
- 10: $Y_{k+1} = Y_k + \mu_k (D A_{k+1} E_{k+1} S_{k+1});$
- 11: $R_{k+1} = max(0, R_k + \mu_k(S_{k+1} C));$
- 12: $\mu_{k+1} = min(\rho\mu_k, 10^7\mu_0);$
- 13: $k \leftarrow k + 1$.
- 14: end while
- 15: return E_k , S_k , A_k ;



Fig. 1 Asymmetric shrinkage for solution of shadow matrix with (a) $c - r/\mu \ge 0$ and (b) $c - r/\mu < 0$.

$$0 \in \frac{\partial L}{\partial S} = \lambda_2 \partial ||S||_1 + \mu(S - Z) + [R + \mu(S - C)]\partial P$$
(5)

Here, *C* is a constant threshold matrix calculated by observed matrix *D*; in practice, *C* is usually related to the average intensity of *D*. Therefore, different thresholds should be discussed in full range of *C* and its corresponding parameters. For simplicity, we use lower case (s, z, r, c) to represent the entry of the corresponding matrix variables. In (5), the two sub-gradients $(\partial ||S||_1, \partial P)$ should be discussed. Fortunately, when $s = c - r/\mu$, the term $[r + \mu(s - c)]\partial P$ is always equal to 0. We further derive the solution in the two forms shown in Fig. 1, with $c - r/\mu$ shown for different numerical ranges.

Specifically, when $c - r/\mu \ge 0$, the matrix entry of $S_*(Z)$ is defined as follows:

1258

$$S_{*}(z) = \begin{cases} \frac{1}{2}(z+c-\frac{r}{\mu}-\frac{\lambda_{2}}{\mu}), & z \ge c-\frac{r}{\mu}+\frac{\lambda_{2}}{\mu}; \\ z-\frac{\lambda_{2}}{\mu}, & \frac{\lambda_{2}}{\mu} \le z < 0; \\ 0, & -\frac{\lambda_{2}}{\mu} \le z < \frac{\lambda_{2}}{\mu}; \\ z-\frac{\lambda_{2}}{\mu}, & z \le -\frac{\lambda_{2}}{\mu}. \end{cases}$$
(6)

And when $c - r/\mu < 0$, $S_*(z)$ is as follows:

$$S_{*}(z) = \begin{cases} \frac{1}{2}(z+c-\frac{r}{\mu}-\frac{\lambda_{2}}{\mu}), & z \ge -c+\frac{r}{\mu}+\frac{\lambda_{2}}{\mu}; \\ 0, & -c+\frac{r}{\mu}-\frac{\lambda_{2}}{\mu} \le z < -c+\frac{r}{\mu}+\frac{\lambda_{2}}{\mu}; \\ \frac{1}{2}(z+c-\frac{r}{\mu}+\frac{\lambda_{2}}{\mu}), & (7) \\ c-\frac{r}{\mu}-\frac{\lambda_{2}}{\mu} \le z < -c+\frac{r}{\mu}-\frac{\lambda_{2}}{\mu}; \\ z+\frac{\lambda_{2}}{\mu}, & z < c-\frac{r}{\mu}-\frac{\lambda_{2}}{\mu}. \end{cases}$$

We call the piecewise functions of (6) and (7) asymmetric shrinkage functions, which are easy to implement via min or max operators. As shown in Fig. 1, when *S* stay in the feasible domain of inequality constraint ($S \le C - R/\mu$), the problem is the same as the original shrinkage. But when *S* jumps out of the feasible domain, there are a penalty cast on input *Z*. So the slope of the shrinkage function became to 0.5 (not the 1.0) and force the input *Z* to reduce faster into the feasible domain than the symmetric shrinkage. In other words, the asymmetric shrinkage has ability of controlling the numeric scale or constraint which the original method doesn't have.

In the experiment, we used Algorithm 1 with welltuned parameters defined as follows: n = 100; $C_{i,j} = 0.3mean(D)$; $\mu_0 = 1.25/||D||_2$, $\rho = 1.5$, and $J(D) = max(||D||_2, \sqrt{m}||D||_{\infty})$, where $\|\cdot\|_2$ and $\|\cdot\|_{\infty}$ are the spectral norm and l_{∞} matrix norm, respectively; λ_1 and λ_2 are set to $0.5/\sqrt{m}$ and $0.4/\sqrt{m}$, respectively; and the convergence condition on Line 2 of Algorithm 1 is $||D - A_k - E_k - S_k||_F^2 + ||max(S_k - C, -R_k/\mu_k)||_F^2 < 10^{-6}$. To post-process the results, we used the 3σ method recommended in [1].

2.3 Results

Representative results are shown in Fig. 2, demonstrating that our ICRPCA method can effectively separate the moving shadow regions. In the first row of the Fig. 2, raw image sequences extracted from the CDnet [9] and shadow evaluation [5] are shown. The second row shows the corresponding ground truth. And in the next 4 rows, there are Chromaticity [12], Large region texture (LR Texture) [11], LOB-STOR [15] and Multi-Cues [13] which are usually used to detect the foreground and remove the shadow. The first two of them are the individual shadow removal process after GMM background subtraction. If the background subtraction don't work well, the shadow removal step is often failed as well. We show this situation in column 2, row 3 and 4. And the rest of them are shadow invariant object detection methods based on multi-feature and local binary similarity patterns. In the bottom three rows, we compare our three-part ICRPCA method to the classic RPCA method via IALM [7] and hard-threshold RPCA method (RPCA-HT).



Fig. 2 ICRPCA results (bottom row) versus other methods for moving object detection.

In the RPCA-HT, we first get the foreground via classic RPCA, and simply use a constant value C which is the same as the one mentioned in (3) to threshold the pixels in the foreground region.

To distinguish the shadow region in our results, we used gray intensity value (255, 50, 0) to represent the moving foreground pixel, the shadow, and the background, respectively. As mentioned in the Introduction, the real intention of using RPCA method is to provide fundamental binary data for further detection or tracking. We therefore added bounding boxes automatically to our results with morphology and image region tools in matlab, also shown in the figure. In column 1, all other methods except the RPCA-HT and ours add a large error region, which is caused by the moving shadow entering the foreground; this results in the subsequent detection phase to incorrectly identify two moving objects instead of just one. And in column 2, methods which are based on chromaticity and texture are failed to detect the foreground because GMM background subtraction doesn't works well on this scene. In column 3, the MutilCue, LOBSTER, and RPCA-IALM methods are not so robust to shadows. In essence, the inequality constraint of our method causes most entries of sparse matrix S (i.e., the background pixels) to be close to constraint c, but less than c. In other words, we do not have a hard threshold here, but rather a soft threshold with constraint of the mean value of the whole sparse matrix. Therefore, as evident in column 2, our method cau distinguish shadows from the moving dark color foreground to a significant extent.

Lastly, we report on the amount of computations costed by our ICRPCA algorithm on a work station with Intel i5-4690 CPU and 4GB memory executing Matlab code. The average processing time on a sequence of 100 frames with resolution 320×240 is about 21 seconds. So it can be used as the first step of our surveillance video analysis, indexing and searching system. The comparative experiments are also with the help of BGSLibrary [14].

3. Conclusions and Future Work

Error pixels in an object's bounding box or an erroneously detected object will clearly cause interference to object recognition or tracking applications. Our proposed ICR-PCA method overcomes these limitations and problems by successfully removing moving shadows and effectively separating foreground elements. In the future, we will study moving camera scenes. Moreover, additional types of constraint forms will be discussed. Extensive experiments are also needed on a wider range of different datasets.

Acknowledgement

This work is supported by the National Nature Science Foundation (61203170) and the Nature Science Foundation of Jiangsu Province, China (BK2012512).

References

[1] Z. Gao, L.F. Cheong, and Y.X. Wang, "Block-Sparse RPCA for

- [2] B.S. He, M. Tao, and X.M. Yuan, "Alternating direction method with Gaussian back substitution for separable convex programming," SIAM J. Optimization, vol.22, no.2, pp.313–340, 2012.
- [3] T. Bouwmans and E. Zahzah, "Robust PCA via principal component pursuit: A review for a comparative evaluation in video surveillance," Special Issue on Background Models Challenge, Computer Vision and Image Understanding, vol.122, pp.22–34, May 2014.
- [4] L. Zhu, Y. Tomioka, and H. Kitazawa, "Extraction and tracking moving objects in detail considering visual feature constraint and structure constraint," IEICE Trans. Inf. & Syst., vol.E96-D, no.5, pp.1171–81, May 2013.
- [5] A. Sanin, C. Sanderson, and B. Lovell, "Shadow detection: A survey and comparative evaluation of recent methods," Pattern Recognit., vol.45, no.4, pp.1684–1695, April 2012.
- [6] E. Candes, X.D. Li, Y. Ma, and J. Wright, "Robust principal component analysis," J. ACM, vol.58, no.3, Article 11, May 2011.
- [7] Z.C. Lin, R.S. Liu, and Z.X. Su, "Linearized alternating direction method with adaptive penalty for low rank representation," Proc. 25th Adv. Neural. Inf. Processing Systems., Granada, Spain, pp.612–620, Dec. 2011.
- [8] D. Ruefenacht, C. Fredembach, and S. Suesstrunk, "Automatic and accurate shadow detection using near-infrared information," IEEE Trans. Pattern Anal. Mach. Intell., vol.36, no.8, pp.1672–1678, Aug. 2014.
- [9] N. Goyette, P. Jodoin, F. Porikli, J. Konrad, and P. Ishwar, "Changedetection.net: A new change detection benchmark data set," Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog. Workshops, pp.1–8, 2012.
- [10] D. Bertsekas, Constrained Optimization and Lagrange Multiplier Methods, pp.158–164, Athena Scientific, Belmont, 1996.
- [11] A. Sanin, C. Sanderson, and B. Lovell, "Improved shadow removal for robust person tracking in surveillance scenarios," Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog., pp.141–144, 2010.
- [12] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Detecting moving objects, ghosts, and shadows in video streams," IEEE Trans. Pattern Anal. Mach. Intell., vol.25, no.10, pp.1337–1342, 2003.
- [13] N. SeungJong and M. Jeon, "A new framework for background subtraction using multiple cues," Proc. 11th Asian Conf. Comput. Vis., pp.493–506, Daejeon, Korea, Nov. 2012.
- [14] S. Andrews, "BGSLibrary: An OpenCV C++ background subtraction library," Proc. IX Workshop de Visão Computacional (WVC '2013), Rio de Janeiro, Brazil, June 2013.
- [15] P. St-Charles and G. Bilodeau, "Improving background subtraction using local binary similarity patterns," Proc. IEEE Comput. Soc. Winter Conf. Applications of Computer Vision (WACV), pp.509– 515, Steamboat Springs, USA, March 2014.