



HAL
open science

Moving Object Detection by Robust PCA solved via a Linearized Symmetric Alternating Direction Method

Charles Guyon, Thierry Bouwmans, El-Hadi Zahzah

► **To cite this version:**

Charles Guyon, Thierry Bouwmans, El-Hadi Zahzah. Moving Object Detection by Robust PCA solved via a Linearized Symmetric Alternating Direction Method. International Symposium on Visual Computing, ISVC 2012, Jul 2012, Greece. pp.427-436. hal-00809467

HAL Id: hal-00809467

<https://hal.science/hal-00809467>

Submitted on 9 Apr 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Moving Object Detection by Robust PCA solved via a Linearized Symmetric Alternating Direction Method

C. Guyon, T. Bouwmans, E. Zahzah

Department of Mathematics, Images and Applications
University of La Rochelle, France
tbouzman@univ-lr.fr

Abstract. Robust Principal Components Analysis (RPCA) gives a suitable framework to separate moving objects from the background. The background sequence is then modeled by a low rank subspace that can gradually change over time, while the moving objects constitute the correlated sparse outliers. RPCA problem can be exactly solved via convex optimization that minimizes a combination of the nuclear norm and the l_1 -norm. This convex optimization is commonly solved by an Alternating Direction Method (ADM) that is not applicable in real application, because it is computationally expensive and needs a huge size of memory. In this paper, we propose to use a Linearized Symmetric Alternating Direction Method (LSADM) to achieve RPCA for moving object detection. LSADM in its fast version requires less computational time than ADM. Experimental results on the Wallflower and I2R datasets show the robustness of the proposed approach.

1 Introduction

The detection of moving objects is a key issue in a video surveillance system with static cameras. This detection is commonly done using foreground detection. This basic operation consists of separating the moving objects called "foreground" from the static information called "background" [1]. In 1999, Oliver et al. [2] are the first authors who model the background by Principal Component Analysis (PCA). PCA provides a robust model of the probability distribution function of the background, but not of the moving objects while they do not have a significant contribution to the model. The main limitation of this model [3] is that the size of the foreground object must be small in relation to the size of the image, and don't appear in the same location during a long period in the training sequence, that is the object does not have to be present more than half of the training sequence. Recent research on robust PCA [4][5] can be used to alleviate these limitations. For example, Candès et al. [5] proposed a convex optimization to address the robust PCA problem. The observation matrix is assumed represented as:

$$A = L + S \quad (1)$$

where L is a low-rank matrix and S must be sparse matrix with a small fraction of nonzero entries. This research seeks to solve for L with the following optimization problem:

$$\min_{L,S} \|L\|_* + \lambda \|S\|_1 \quad \text{subj } A = L + S \quad (2)$$

where $\|\cdot\|_*$ and $\|\cdot\|_1$ are the nuclear norm and l_1 -norm, respectively, and $\lambda > 0$ is an arbitrary balanced parameter. This approach assumed that all entries of the matrix to be recovered are exactly known via the observation. The other assumption is that the distribution of corruption should be sparse and random enough without noise. Under these minimal assumptions, this approach perfectly recovers the low-rank and the sparse matrices. The optimization in Equation (2) can be solved as a general convex optimization problem by any iterative thresholding techniques [6]. However, the iterative thresholding scheme converges extremely slowly. To alleviate this slow convergence, Lin et al. [7] proposed the accelerated proximal gradient (APG) algorithm and the gradient-ascent algorithm applied to the dual of the problem in the Equation (2). However, these algorithms are all the same to slow for real application. More recently, Lin et al. [8] proposed two algorithms based on augmented Lagrange multipliers (ALM). The first algorithm is called exact ALM (EALM) method that has a Q-linear convergence speed, while the APG is only sub-linear. The second algorithm is an improvement of the EALM that is called inexact ALM (IALM) method, which converges practically as fast as the exact ALM, but the required number of partial SVDs is significantly less. The IALM is at least five times faster than APG, and its precision is also higher. However, the direct application of ALM treats Equation (2) as a generic minimization problem and doesn't take into account its separable structure emerging in both the objective function and the constraint. Hence, the variables S and L are minimized simultaneously. Yuan and Yang [9] proposed to alleviate this problem by the Alternating Direction Method (ADM) which minimizes the variables L and S serially. However, this method is computationally expensive and needs a huge size of memory. Recently, Ma[10] and Goldfarb et al. [11] proposed a Linearized Symmetric Alternating Direction Method (LSADM) for minimizing the sum of two convex functions. This method requires at most $O(1/\epsilon)$ iterations to obtain an ϵ -optimal solution, while its fast version called Fast-LSADM requires at most $O(1/\sqrt{\epsilon})$ with little change in the computational effort required at each iteration. These algorithms [10] have shown encouraging qualitative results on background extraction. These properties allow us to use and evaluate it for moving object detection by using RPCA. The rest of this paper is organized as follows. In Section 2, we present how the background and the moving objects can be separated by using RPCA. Then, the proposed moving object detection method with the LSADM algorithm is presented. Finally, performance evaluation and comparison are given in Section 3.

2 Moving object detection by RPCA solved via LSADM

2.1 Background and moving object separation via RPCA

Denote the training video sequences $D = \{I_1, \dots, I_N\}$ where I_t is the frame at time t and N is the number of training frames. Let each pixel (x,y) be characterized by its intensity in the grey scale. The decomposition involves the following model:

$$D = L + S \quad (3)$$

where L and S are the low-rank component and sparse component of D, respectively. The matrix L contains the background and the matrix S contains mostly zero columns,

with several non-zero ones corresponding to the moving objects. The matrices L and S can be recovered by the convex program based on the Alternating Direction Method (ADM). For this work on moving object detection, we propose to use the Linearized Symmetric Alternating Direction Method (LSADM) proposed in [10][11].

2.2 Linearized Symmetric Alternating Direction Method

This sub-section briefly reminds the principle of LSADM developed in [10][11]. Theoretically, the RPCA problem can be initially formulated as a linear constrained convex program [10].

$$\min(f(x) + g(y)) \quad \text{subj } x - y = 0 \quad (4)$$

Solving this problem by ADM, one operates on the following augmented Lagrangian function:

$$\mathcal{L}_\mu(x, y; \lambda) = f(x) + g(y) + \langle \lambda, x - y \rangle + \frac{1}{2\mu} \|x - y\|^2 \quad (5)$$

with respect to x and y , i.e, it solves the subproblem:

$$(x^k, y^k) = \operatorname{argmin}_{x, y} \mathcal{L}_\mu(x, y; \lambda^k) \quad (6)$$

and then the Lagrange multipliers λ^k are updated as follows:

$$\lambda^{k+1} = \lambda^k + \frac{1}{\mu}(x^k - y^k) \quad (7)$$

Minimizing $\mathcal{L}_\mu(x, y; \lambda)$ with respect to x and y alternately is often easy. Such an alternating direction method for solving Equation (4) is given below as Algorithm 1.

Algorithm 1: Alternating Direction Method (ADM)

Initialization: Choose μ, λ^0 and $x^0 = y^0$

Line 1 for $k = 0, 1, \dots$ **do**

Line 2 $x^{k+1} = \operatorname{argmin}_x \mathcal{L}_\mu(x, y^k; \lambda^k)$

Line 3 $y^{k+1} = \operatorname{argmin}_y \mathcal{L}_\mu(x^{k+1}, y; \lambda^k)$

Line 4 $\lambda^{k+1} = \lambda^k + \frac{1}{\mu}(x^{k+1} - y^{k+1})$

Line 5 **end**

In each iteration of ADM, the Lagrange multiplier λ is updated just once after the augmented Lagrangian is minimized with respect to y (Line 4). For the ADM to be symmetric with respect to x and y , Ma [10] updated also λ after solving the subproblem with respect to x . Such as symmetric ADM is given below as Algorithm 2.

Algorithm 2: Symmetric Alternating Direction Method (SADM)

Initialization: Choose μ, λ^0 and $x^0 = y^0$

Line 1 for $k = 0, 1, \dots$ **do**

Line 2 $x^{k+1} = \operatorname{argmin}_x \mathcal{L}_\mu(x, y^k; \lambda^k)$

Line 3 $\lambda^{k+\frac{1}{2}} = \lambda^k + \frac{1}{\mu}(x^{k+1} - y^k)$

Line 4 $y^{k+1} = \operatorname{argmin}_y \mathcal{L}_\mu(x^{k+1}, y; \lambda^{k+\frac{1}{2}})$
Line 5 $\lambda^{k+1} = \lambda^{k+\frac{1}{2}} + \frac{1}{\mu}(x^{k+1} - y^{k+1})$
Line 6 **end**

Ma [10] assumed that both $f(x)$ and $g(x)$ are differentiable to linearize the SADM. It follows from the first order optimality conditions that $\lambda^{k+\frac{1}{2}} = \operatorname{grad}(f(x^{k+1}))$ and $\lambda^{k+1} = \operatorname{grad}(g(y^{k+1}))$. Then, the Lagrangian functions can be replaced by:

$$Q_f(u, v) = f(u) + g(v) + \langle \operatorname{grad}(f(u)), u - v \rangle + \frac{1}{2\mu} \|u - v\|^2 \quad (8)$$

$$Q_g(u, v) = f(u) + g(v) + \langle \operatorname{grad}(g(u)), u - v \rangle + \frac{1}{2\mu} \|u - v\|^2 \quad (9)$$

Algorithm 3: Linearized Symmetric Alternating Direction Method (LSADM)

Initialization: Choose μ and $x^0 = y^0$
Line 1 **for** $k = 0, 1, \dots$ **do**
Line 2 $x^{k+1} = \operatorname{argmin}_x Q_g(x, y^k)$
Line 3 $y^{k+1} = \operatorname{argmin}_y Q_f(y, x^{k+1})$
Line 4 **end**

In Algorithm 3, the functions f and g are alternatively replaced by their linearizations plus a proximal regularization term to get an approximation to the original function F . The complexity of LSADM is $O(\epsilon)$ for obtaining an ϵ -optimal solution for Equation (4). Finally, to apply the LSADM to the RPCA, the formulation in Equation 3 is firstly restated as the following convex optimization problem:

$$\min_{L, S} \|L\|_* + \lambda \|S\|_1 \quad \text{subj } D = L + S \quad (10)$$

where $\|\cdot\|_*$ and $\|\cdot\|_1$ are the nuclear norm (which is the L_1 norm of singular value) and l_1 -norm, respectively, and $\lambda > 0$ is an arbitrary balanced parameter. In order to apply the LSADM algorithm to Equation 3, both the nuclear norm $f(L) = \|L\|_*$ and the l_1 norm $g(S) = \lambda \|S\|_1$ are smoothed by using the Nesterov's smoothing technique. A smoothed approximation of $f(L)$ is obtained as follows:

$$f_\sigma(L) = \max(\langle L, W \rangle - \frac{\sigma}{2} \|W\|_F^2 : \|W\|_F \leq 1) \quad (11)$$

where $W_\sigma(L) = U \operatorname{Diag}(\min(\gamma, 1)) V^T$. $U \operatorname{Diag}(\min(\gamma, 1)) V^T$ is the singular value decomposition (SVD) of L/σ . A smoothed approximation to the l_1 of $g(S) = \rho \|S\|_1$ is obtained as follows:

$$g_\sigma(S) = \max(\langle S, Z \rangle - \frac{\sigma}{2} \|Z\|_2^2 : \|Z\|_\infty \leq \rho) \quad (12)$$

where $Z_\sigma(S) = \min(\rho, \max(\frac{S}{\sigma}, -\rho))$. After smoothing f and g , the LSADM algorithm is applied to solve the smoothed problem:

$$\min(f_\sigma(L) + g_\sigma(S)) \quad \text{subj } L + S = D \quad (13)$$

2.3 Moving Object Detection Algorithm

The background and moving object separation can be obtained by solving the Equation(13) with LSADM. Then, the proposed algorithm for moving object detection is as follows:

Algorithm for moving object detection by RPCA solved via LSADM

Input data: Training sequence with N images that is contained in D.

Output data: Background in L and moving objects in S.

Parameters initialization: $L_0 \leftarrow D$; $S_0 \leftarrow O$; $Y \leftarrow O$; $\mu = 30/\|sign(D)\|_2$; $\rho > 0$, $\sigma > 0$, $k \leftarrow 0$.

$L \leftarrow L_0$; $S \leftarrow S_0$;

Background and Moving Object Separation:

for $k = 1, \dots$ to *maxiter*

$$X = \mu * Y - S + D.$$

$$X = UGV^T \text{ (SVD)}.$$

$$G \leftarrow \text{Diag}(G).$$

$$G \leftarrow G - \mu \frac{G}{\max(G, \mu + \sigma)}.$$

$$L_k \leftarrow L.$$

$$L = U \text{Diag}(G) V^T.$$

$$Y \leftarrow Y - \frac{L+S-D}{\mu}.$$

$$B \leftarrow Y - \frac{L-D}{\mu}.$$

$$S_k \leftarrow S.$$

$$S = \mu B - \mu * \min(\rho, \max(-\rho, \frac{\mu B}{\sigma + \mu})).$$

$$Y \leftarrow Y - \frac{L-S-D}{\mu}.$$

$$d = \|\frac{L+S-D}{\mu}\|_F / \|D\|_F.$$

if $d < \epsilon$

$$L \leftarrow L_k, S \leftarrow S_k.$$

goto Moving Object Detection step.

end if

end for

$L \leftarrow L_k, S \leftarrow S_k$.

Moving Object Detection: Threshold the matrix S to obtain the moving objects.

For the initialization, the low-rank matrix L, the sparse matrix S, and the Lagrange multiplier Y are set respectively to the matrix D, O and O. The parameters *maxiter* and ϵ are set respectively to 100 and $10e - 7$. The computational cost in the LSADM algorithm is mainly related to the singular value decomposition (SVD). It can be reduced significantly by using a partial SVD because only the first largest few singular values are needed. Practically, we used the implementation available in PROPACK¹. Fig. 1 shows the original frames 309, 395 et 462 of the sequence from [10] and their decomposition into the low-rank matrix L and sparse matrix S. We can see that L corresponds to the background whereas S corresponds to the moving objects. The fourth image shows the moving object mask obtained by thresholding the matrix S and the fifth image is the ground truth image.

¹ <http://soi.stanford.edu/rmunk/PROPACK/>

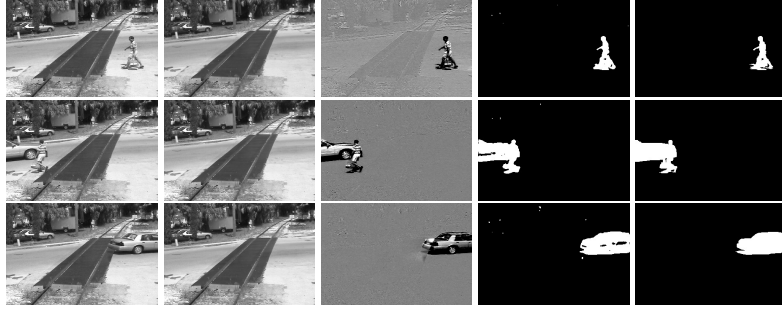


Fig. 1. From left to right: Original image, low-rank matrix L (background), sparse matrix S (moving objects), moving object mask, ground truth.

3 Implementation

For the implementation, we choose to implement the LSADM algorithm and its fast version called Fast-LSADM which computes an ϵ -optimal solution to the problem in Equation (4) in $O(1/\sqrt{\epsilon})$ iterations [10][11]. It is a successive over-relaxation type algorithm since $(t_k - 1)/t_{k+1} > 0, \forall k \geq 2$ as follows.

Algorithm 4: Fast-LSADM

Initialization: Choose $\mu, x^0 = y^0$ and set $t_1 = 1$

Line 1 for $k = 1, 2, \dots$ do

Line 2 $x^k = \operatorname{argmin}_x Q_g(x, z^k)$

Line 3 $y^k = \operatorname{argmin}_y Q_f(y, x^k)$

Line 4 $t^{k+1} = (1 + \sqrt{1 + 4 * t_k^2})/2$

Line 5 $z^{k+1} = y^k + \frac{t_k+1}{t_{k+1}}(y^k - y^{k+1})$

Line 6 end

The algorithm for moving object detection by RPCA solved via Fast LSADM can be easily derived from the one with LSADM.

4 Experimental Results

We compared LSADM and Fast-LSADM with standard methods (SG [12], MOG [13], PCA [2]) and robust PCA methods (RSL [4], EALM [8], IALM [8]). The experiments were conducted qualitatively and quantitatively on the Wallflower dataset [14] and I2R dataset [15]. The algorithms were implemented in batch mode with matlab. The comparison is essentially conducted against IALM as RSL and EALM are very computational expensive. Furthermore, there was not much difference between the performance of LSADM and that of Fast-LSADM in terms of detection. So, we present only visual results obtained with the Fast-LSADM. Due the 10 page limitation, we present only the visual results obtained by the standard methods on the Wallflower dataset.

4.1 Wallflower dataset²

This dataset consists of seven video sequences, with each sequence presenting one of the difficulties a practical task is likely to encounter: Moved Object (MO), Time of Day (TD), Light Switch (LS), Waving Trees (WT), Camouflage (C), Bootstrapping (B) and Foreground Aperture (F). The images are 160×120 pixels. For each sequence, the ground truth is provided for one image when the algorithm has to show its robustness to a specific change in the scene. Thus, the performance is evaluated against hand-segmented ground truth. The figure 2 shows the qualitative results. For the quantitative evaluation, we used metrics based on the true negative (TN), true positive (TP), false negative (FN), false positive (FP) detections. Then, we computed the detection rate, the precision and the F-measure. The detection rate is given as follows:

$$DR = \frac{TP}{TP + FN} \quad (14)$$

The precision is computed as follows:

$$Precision = \frac{TP}{TP + FP} \quad (15)$$

A good performance is obtained when the detection rate is high without altering the precision. This can be measured by the F-measure [16] as follows:

$$F = \frac{2 \times DR \times Precision}{DR + Precision} \quad (16)$$

A good performance is then reached when the F-measure is closed to 1. Table 1 shows in percentage the F-measure for each sequence and its average on the dataset. The F-measure value of MO sequence can't be computed due to the absence of true positives in its ground-truth. We have highlighted when Fast-LSADM outperforms IALM. It is the case on the sequences (MO, TD, LS, WT, C, B, FA). For the sequence LS, the result is still acceptable. RSL and EALM gives the best results on the sequences (TD, WT, C, FA) but the time requirement is very higher than IALM, LASDM and Fast-LSADM. So, the proposed method offers a nice compromise between robustness and computational efficiency. As these encouraging results are obtained by using one ground-truth image one each sequence, we have evaluated the proposed method on a dataset with more ground-truth images in the following sub-section.

4.2 I2R dataset⁴

This dataset provided by [15] consists of nine video sequences, which each sequence presenting dynamic backgrounds or illumination changes. The size of the images is 176×144 pixels. For each sequence, the ground truth is provided for 20 images. Among this dataset, we have chosen to show results on four representative sequences that are the following ones: airport, shopping mall, water surface and curtain. We ran RSL, IALM,

² <http://research.microsoft.com/en-us/um/people/jckrumm/wallflower/testimages.htm>

³ <http://sites.google.com/site/backgroundsubtraction/test-image-sequences—results>

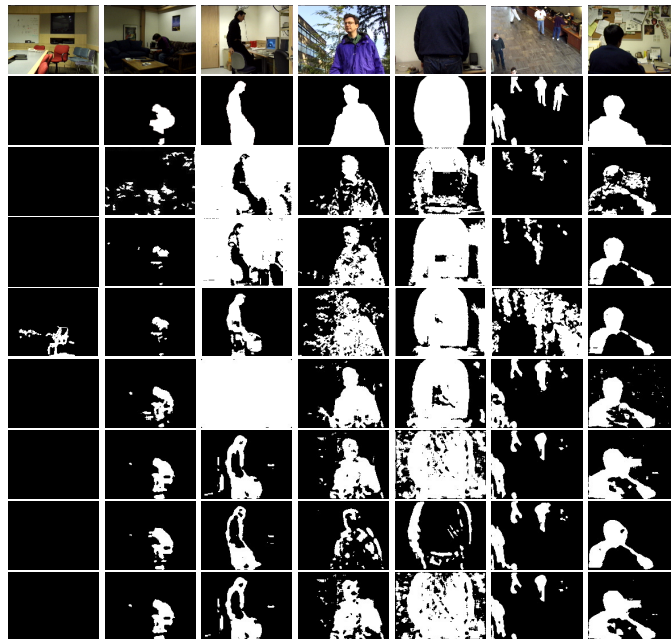


Fig. 2. Experimental results on the Wallflower dataset. From top to bottom: original image, ground truth, SG, MOG, PCA, RSL, EALM, IALM, Fast-LSADM. From left to right: MO (985), TD (1850), LS (1865), WT (247), C (251), B (2832), FA (449). More results with other background subtraction methods available on the Background Subtraction Web Site³.



Fig. 3. Experimental results on the I2R dataset. From left to right: original image, ground truth, RSL, IALM, Fast-LSADM. From top to bottom: airport (2926), shopping mall (1980), water surface (1594), curtain (23257).

	RSL	EALM	IALM	LSADM	Fast-LSADM
Time of Day	75.73	81.18	80.56	80.24	80.84
Light Switch	28.36	70.86	73.16	69.11	69.33
Waving Trees	89.69	86.40	40.88	81.25	81.67
Camouflage	91.78	75.43	22.02	75.71	76.01
Bootstrap	69.38	74.4	73.73	74.11	74.69
Foreground Aperture	74.37	72.07	61.92	71.26	71.56
Average	71.55	76.73	58.71	75.28	75.69

Table 1. F-measure on the Wallflower dataset

	RSL	IALM	LSADM	Fast-LSADM
Airport	65.26	74.26	71.27	71.57
Shopping mall	58.64	50.89	75.24	75.74
Water surface	34.42	31.34	45.17	47.69
Curtain	70.73	76.59	81.11	83.33
Average	57.26	58.27	68.19	69.58

Table 2. F-measure on the I2R dataset

LSADM and Fast-LSADM on these sequences. We skipped EALM since it would take excessive amounts of time due to full SVD calculations. Since all these video clips have more than 1000 frames, we took a part of each clip with 200 frames. Fig. 3 shows the qualitative results. For example, we can see that the LSADM allows to detect the complete silhouette in the sequence called "water surface". Table 2 shows the average F-measure in percentage that is obtained on 20 ground truth images for each sequence and its average on the dataset. We have highlighted when Fast LSADM outperforms IALM. We can see that Fast LSADM outperforms RSL and IALM for each sequence except for the sequence "Airport". The SVDs and CPU time of each algorithm was computed for each sequence. For example, in the case of the sequence "Airport" of resolution 176×144 with 200 images, the CPU time is 40min15s, 3min47s, 4min30s and 1min56s respectively for EALM, IALM, LSADM and Fast-LSADM. The SVDs times is 550 SVDs, 38 SVDs, 43 SVDs and 6 SVDs respectively for EALM, IALM, LSADM and Fast-LSADM. On these problems of extremely low ranks, the partial SVD technique used in IALM, LSADM and Fast-LSADM becomes quite effective. Even so, the CPU times required by IALM are still about two times of those required by Fast-LSADM. Furthermore, the speed can be improved by a GPU implementation.

5 Conclusion

In this paper, we have presented a moving object detection method based on RPCA that is solved via a linearized alternating direction method. This method allows us to alleviate the constraints of the identities. Furthermore, experiments on video surveillance datasets show that this approach is more robust than RSL and IALM in the presence of

⁴ <http://perception.i2r.a-star.edu.sg/>

dynamic backgrounds and illumination changes. In terms of computational efficiency, the proposed approach has exhibited a significant speed advantage over IALM. Although the proposed method is an offline one, meaning that the processing is done after a large number of frames are acquired and therefore, it is not done in real time. It is possible to extend the concept to incremental, real time processing by adaptively update the low rank component. This will be investigated in details in a future paper.

6 Acknowledgements

The authors would like to thank Shiqian Ma⁵ (Institute for Mathematics and Its Applications, Univ. of Minnesota, USA) who has kindly provided the algorithm LSADM.

References

1. Bouwmans, T.: Recent advanced statistical background modeling for foreground detection: A systematic survey. *RPCS* **4** (2011) 147–176
2. Oliver, N., Rosario, B., Pentland, A.: A bayesian computer vision system for modeling human interactions. *ICVS 1999* (1999)
3. Bouwmans, T.: Subspace learning for background modeling: A survey. *RPCS* **2** (2009) 223–234
4. Torre, F.D.L., Black, M.: A framework for robust subspace learning. *International Journal on Computer Vision* (2003) 117–142
5. Candes, E., Li, X., Ma, Y., Wright, J.: Robust principal component analysis? *International Journal of ACM* **58** (2011)
6. Wright, J., Peng, Y., Ma, Y., Ganesh, A., Rao, S.: Robust principal component analysis: Exact recovery of corrupted low-rank matrices by convex optimization. *NIPS 2009* (2009)
7. Lin, Z., Ganesh, A., Wright, J., Wu, L., Chen, M., Ma, Y.: Fast convex optimization algorithms for exact recovery of a corrupted low-rank matrix. *UIUC Technical Report* (2009)
8. Lin, Z., Chen, M., Wu, L., Ma, Y.: The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *UIUC Technical Report* (2009)
9. Yuan, X., Yang, J.: Sparse and low-rank matrix decomposition via alternating direction methods. *Optimization Online* (2009)
10. Ma, S.: Algorithms for sparse and low-rank optimization: Convergence, complexity and applications. *Thesis* (2011)
11. Goldfarb, D., Ma, S., Scheinberg, K.: Fast alternating linearization methods for minimizing the sum of two convex function. *Preprint, Mathematical Programming Series A* (2010)
12. Wren, C., Azarbayejani, A., Darrell, T., Pentland, A.: Pfunder : Real-time tracking of the human body. *IEEE Transactions on PAMI* **19** (July 1997) 780–785
13. Stauffer, C., Grimson, W.: Adaptive background mixture models for real-time tracking. *CVPR 1999* (1999) 246–252
14. Toyama, K., Krumm, J., Brumitt, B., Meyers, B.: Wallflower: Principles and practice of background maintenance. *ICCV* (1999) 255–261
15. Li, L., Huang, W., Gu, I., Tian, Q.: Statistical modeling of complex backgrounds for foreground object detection. *IEEE T-IP* (2004) 1459–1472
16. Maddalena, L., Petrosino, A.: A fuzzy spatial coherence-based approach to background foreground separation for moving object detection. *Neural Computing and Applications* (2010) 1–8

⁵ <http://www.columbia.edu/~sm2756/>