Comparison of Matrix Completion Algorithms for Background Initialization in Videos

Andrews Sobral[1,2](✉), Thierry Bouwmans[2], and El-hadi Zahzah[1]

[1] Lab. L3I, Université de La Rochelle, 17000 La Rochelle, France
[2] Lab. MIA, Université de La Rochelle, 17000 La Rochelle, France
andrews.sobral@univ-lr.fr

Abstract. Background model initialization is commonly the first step of the background subtraction process. In practice, several challenges appear and perturb this process such as dynamic background, bootstrapping, illumination changes, noise image, etc. In this context, this work aims to investigate the background model initialization as a matrix completion problem. Thus, we consider the image sequence (or video) as a partially observed matrix. First, a simple joint motion-detection and frame-selection operation is done. The redundant frames are eliminated, and the moving regions are represented by zeros in our observation matrix. The second stage involves evaluating nine popular matrix completion algorithms with the Scene Background Initialization (SBI) data set, and analyze them with respect to the background model challenges. The experimental results show the good performance of LRGeomCG [17] method over its direct competitors.

Keywords: Matrix completion · Background modeling · Background initialization

1 Introduction

Background subtraction (BS) is an important step in many computer vision systems to detect moving objects. This basic operation consists of separating the moving objects called "foreground" from the static information called "background" [2,16]. The BS is commonly used in video surveillance applications to detect persons, vehicles, animals, etc., before operating more complex processes for intrusion detection, tracking, people counting, etc. Typically the BS process includes the following steps: a) background model initialization, b) background model maintenance and c) foreground detection. With a focus on the step (a), the BS initialization consists in creating a background model. In a simple way, this can be done by setting manually a static image that represents the background. The main reason is that it is often assumed that initialization can be achieved by exploiting some clean frames at the beginning of the sequence. Naturally, this assumption is rarely encountered in real-life scenarios, because of continuous clutter presence. In addition, this procedure presents several limitations,

V. Murino et al. (Eds.): ICIAP 2015 Workshops, LNCS 9281, pp. 510–518, 2015.
DOI: 10.1007/978-3-319-23222-5_62

because it needs a fixed camera with constant illumination, and the background needs to be static (commonly in indoor environments), and having no moving object in the first frames. In practice, several challenges appear and perturb this process such as noise acquisition, bootstrapping, dynamic factors, etc [11].

The main challenge is to obtain a first background model when more than half of the video frames contain foreground objects. Some authors suggest the initialization of the background model by the arithmetic mean [9] (or weighted mean) of the pixels between successive images. Practically, some algorithms are: (1) batch ones using N training frames (consecutive or not), (2) incremental with known N or (3) progressive ones with unknown N as the process generates partial backgrounds and continues until a complete background image is obtained. Furthermore, initialization algorithms depend on the number of modes and the complexity of their background models. However, BS initialization has also been achieved by many other methodologies [2,11]. We can cite for example the computation of eigen values and eigen vectors [15], and the recent research on subspace estimation by sparse representation and rank minimization [3]. The background model is recovered by the low-rank subspace that can gradually change over time, while the moving foreground objects constitute the correlated sparse outliers.

In this paper, the initialization of the background model is addressed as a matrix completion problem. The matrix completion aims at recovering a low rank matrix from partial observations of its entries. The image sequence (or video) is represented as a partially observed real-valued matrix. Figure 1 shows the proposed framework. First, a simple joint motion-detection and frame-selection operation is done. The redundant frames are eliminated, and the moving regions are represented with zeros in our observation matrix. This operation is described in the Section 2. The second stage involves evaluating nine popular matrix completion algorithms with the Scene Background Initialization (SBI) data set [12] (see Section 3). This enables to analyze them with respect to the background model challenges. Finally, in Sections 4 and 5, the experimental results are shown as well as conclusions.

Throughout the paper, we use the following notations. Scalars are denoted by lowercase letters, e.g., x; vectors are denoted by lowercase boldface letters, e.g., $\boldsymbol{x}$; matrices by uppercase boldface, e.g., $\mathbf{X}$. In this paper, only real-valued data are considered.

2 Joint Motion Detection and Frame Selection

In order to reduce the number of redundant frames, a simple joint motion detection and frame selection operation is applied. First, the color images are converted into its gray-scale representation. So, let a sequence of N gray-scale images (frames) $\mathbf{I}_0 \dots \mathbf{I}_N$ captured from a static camera, that is, $\mathbf{I} \in \mathbb{R}^{m \times n}$ where m and n denote the frame resolution (rows by columns). The difference between two consecutive frames (motion detection step) is calculated by:

$$\mathbf{D}_t = \sqrt{(\mathbf{I}_t - \mathbf{I}_{t-1})^2}\,|_{\,t=1,\dots,N}\ , \tag{1}$$

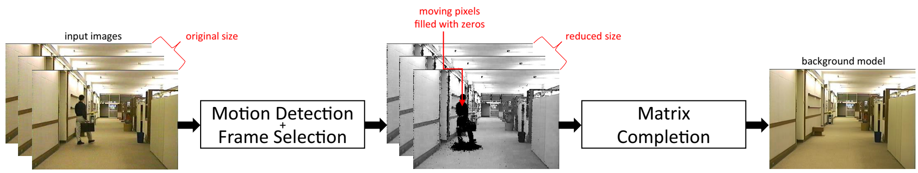

Fig. 1. Block diagram of the proposed approach. Given an input image, a joint motion detection and frame selection operation is applied. Next, a matrix completion algorithm tries to recover the background model from the partially observed matrix. In this paper, the processes described here are conducted in a batch manner.

where $\mathbf{D}_t \in \mathbb{R}^{m \times n}$ denotes the matrix of pixel-wise L_2-norm differences from frame $t-1$ to frame t. Next, the sum of all elements of $\mathbf{D}_t$, for $t = 1, \dots, N$, is stored in a vector $\boldsymbol{d} \in \mathbb{R}^N$ whose t-th element is given by:

$$d_t = \sum_{i=1}^{m} \sum_{j=1}^{n} \mathbf{D}_t(i, j), \tag{2}$$

where $\mathbf{D}_t(i, j)$ is the matrix element located in the row $i \in [1, \dots, m]$ and column $j \in [1, \dots, n]$. Then, the vector $\boldsymbol{d}$ is normalized between 0 and 1 by:

$$\hat{\mathbf{d}} = \frac{d_t - d_{min}}{d_{max} - d_{min}} \,|_{\, t=1,\dots,N} \;, \tag{3}$$

where d_{min} and d_{max} denote the minimum value and the maximum value of the vector $\boldsymbol{d}$. The frame selection step is done by calculating the derivative of $\hat{\mathbf{d}}$ by:

$$\boldsymbol{d}' = \frac{d}{dt} \hat{\mathbf{d}}, \tag{4}$$

Next, the vector $\boldsymbol{d}'$ is also normalized by Equation 3 and represented by $\hat{\mathbf{d}}'$. Finally, the index of the more relevant frames are given by thresholding $\hat{\mathbf{d}}'$:

$$\boldsymbol{y} = \begin{cases} 1 & \text{if } |\hat{\mathbf{d}}' - \hat{\mu}'| > \tau \\ 0 & \text{otherwise} \end{cases}, \tag{5}$$

where $\hat{\mu}'$ denotes the mean value of the vector $\hat{\mathbf{d}}'$, and $\tau \in [0, \dots, 1]$ controls the threshold operator. In this paper, $R \leq N$ represent the set of all frames where $\boldsymbol{y} = 1$, and the parameter τ was chosen experimentally for each scene: $\tau = 0.025$ for HallAndMonitor, $\tau = 0.05$ for HighwayII, $\tau = 0.10$ for HighwayI, and $\tau = 0.15$ to all other scenes. Figure 3 illustrates our frame selection operation, in this example, with $\tau = 0.025$, only 92 relevant frames are selected from a total of 296 frames ($68,92\%$ of reduction). In the next section, the matrix completion process is described.

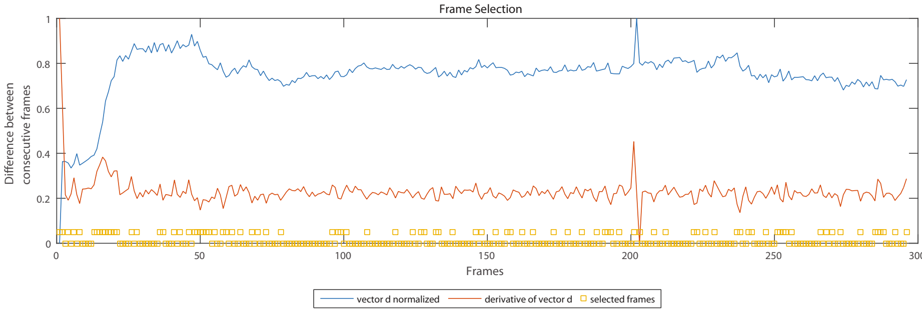

Fig. 2. Illustration of frame selection operation. The normalized vector (in blue) shows the difference between two consecutive frames. The derivative vector draw how much the normalized vector changes (in red), and then it is thresholded and the frames are selected (in orange).

3 Matrix Completion

As explained previously, the matrix completion aims to recover a low rank matrix from partial observations of its entries. Considering the general form of low rank matrix completion, the optimization problem is to find a matrix $\mathbf{L} \in \mathbb{R}^{n1 \times n2}$ with minimum rank that best approximates the matrix $\mathbf{A} \in \mathbb{R}^{n1 \times n2}$. Candès and Recht [6] show that this problem can be formulated as:

$$\begin{aligned} &\text{minimize} \quad rank(\mathbf{A}), \\ &\text{subject to} \quad P_\Omega(\mathbf{A}) = P_\Omega(\mathbf{L}), \end{aligned} \tag{6}$$

where $rank(\mathbf{A})$ is equal to the rank of the matrix $\mathbf{A}$, and P_Ω denotes the sampling operator restricted to the elements of Ω (set of observed entries), i.e., $P_\Omega(\mathbf{A})$ has the same values as $\mathbf{A}$ for the entries in Ω and zero values for the entries outside Ω. Later, Candès and Recht [6] propose to replace the $rank(.)$ function with the nuclear norm $||\mathbf{A}||_*= \sum_{i=1}^{r} \sigma_i$ where $\sigma_1, \sigma_2, ..., \sigma_r$ are the singular values of $\mathbf{A}$ and r is the rank of $\mathbf{A}$. The nuclear norm make the problem tractable and Candès and Recht [6] have proved theoretically that the solution can be exactly recovered with a high probability. In addition, Cai et. al [4] propose an algorithm based on soft singular value thresholding (SVT) to solve this convex relaxation problem. However, in real world application the observed entries may be noisy. In order to make the Equation 6 robust to noise, Candès and Plan [5] propose a stable matrix completion approach. The equality constraint is replaced by $||P_\Omega(\mathbf{A} - \mathbf{L})||_F \leq \epsilon$, where $||.||_F$ denotes the Frobenius norm and ϵ is an upper bound on the noise level. Recently, several matrix completion algorithms have been proposed to deal with this challenge, and a complete review can be found in [21].

In this paper, we address the background model initialization as a matrix completion problem. Once frame selection process is done, the moving regions

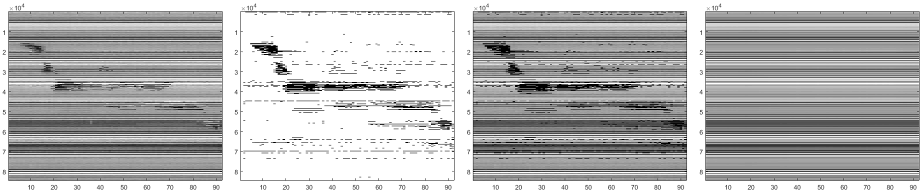

Fig. 3. Illustration of the matrix completion process. From the left to the right: a) the selected frames in vectorized form (our observation matrix), b) the moving regions are represented by non-observed entries (black pixels), c) the moving regions filled with zeros (modified version of the observation matrix), and d) the recovered matrix after the matrix completion process.

of the R selected frames are determined by:

$$\mathbf{M}_k(i,j) = \begin{cases} 1 & \text{if } 0.5(\mathbf{D}_k(i,j))^2 > \beta \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

where $k \in R$, and β is the thresholding parameter (in this paper, $\beta = 1e^{-3}$ for all experiments). Next, the moving regions of each selected frame are filled with zeros by $\mathbf{I}_k \circ \overline{\mathbf{M}_k}$, where $\overline{\mathbf{M}_k}$ denotes the complement of $\mathbf{M}_k$, and $\circ$ denotes the element-wise multiplication of two matrices. For color images, each channel is processed individually, then they are vectorized into a partially observed real-valued matrix $\mathbf{A} = [vec(I_1) \dots vec(I_k)]$, where $\mathbf{A} \in \mathbb{R}^{n1 \times n2}$, $n1 = (m \times n)$, and $n2 = k$. Figure 3 illustrates our matrix completion process. It can be seen that the partially observed matrix can be recovered successfully even with the presence of many missing entries. So, let $\mathbf{L}$ the recovered matrix from the matrix completion process, the background model is estimated by calculating the average value of each row, resulting in a vector $\boldsymbol{l} \in \mathbb{R}^{n1 \times 1}$, and then reshaped into a matrix $\mathbf{B} \in \mathbb{R}^{m \times n}$.

4 Experimental Results

In order to evaluate the proposed approach, nine matrix completion algorithms have been selected, and they are listed in Table 1. The algorithms were grouped in two categories, as well as its main techniques (following the same definition of Zhou et al. [21]).

In this paper, the Scene Background Initialization (SBI) data set was chosen for the background initialization task. The data set contains seven image sequences and corresponding ground truth backgrounds. It provides also MATLAB scripts for evaluating background initialization results in terms of eight metrics[1]. Figure 4 show the visual results for the top three best matrix completion algorithms, and Table 2 reports the quantitative results of each algorithm

[1] Please, refer to http://sbmi2015.na.icar.cnr.it/ for a complete description of each metric.

Table 1. List of low-rank matrix completion algorithms evaluated in this paper.

Category	Method	Main techniques	Reference
Rank Minimization	IALM	Augmented Lagrangian	[10, Linetal.(2010)]
	RMAMR	Augmented Lagrangian	[20, Yeetal.(2015)]
Matrix Factorization	SVP	Hard thresholding	[13, Mekaetal.(2009)]
	OptSpace	Grassmannian	[8, Keshavanetal.(2010)]
	LMaFit	Alternating	[19, Wenetal.(2012)]
	ScGrassMC	Grassmannian	[14, NgoandSaad(2012)]
	LRGeomCG	Riemannian	[17, Vandereycken(2013)]
	GROUSE	Online algorithm	[1, Balzanoetal.(2013)]
	OR1MP	Matching pursuit	[18, Wangetal.(2015)]

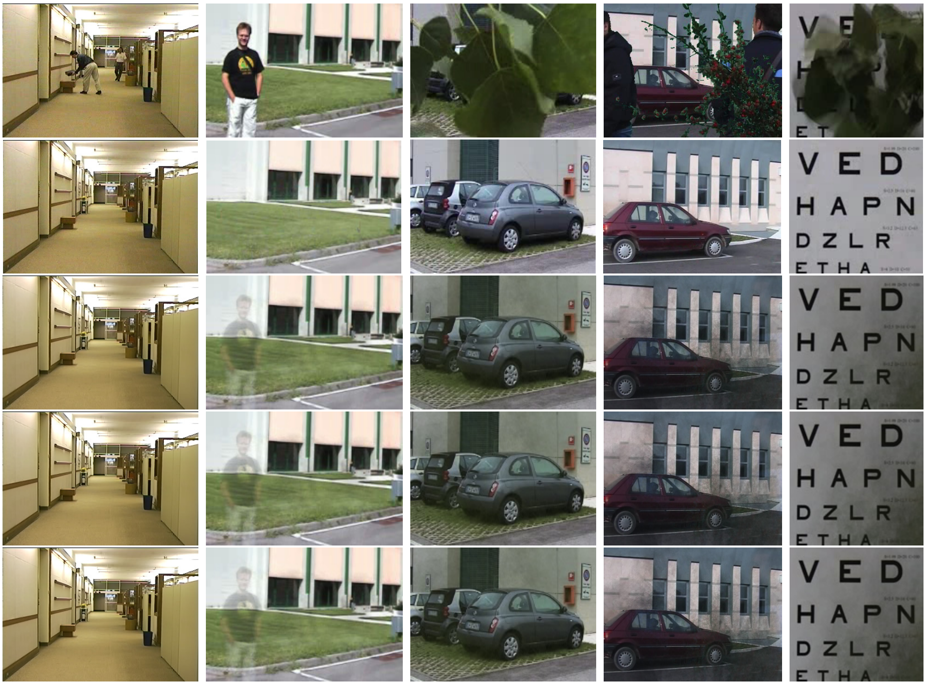

Fig. 4. Visual comparison for the background model initialization. From top to bottom: 1) example of input frame, 2) background model ground truth, and background model results for the top 3 best ranked MC algorithms: 3) LRGeomCG, 4) LMaFit, and 5) RMAMR.

over the data set[2]. The algorithms are ranked as follow: 1) for each algorithm we calculate its rank position for each metric, we call it as *metric rank* (i.e.

[2] Full experimental evaluation and related source code can be found in the main website: https://sites.google.com/site/mc4bmi/

Table 2. Quantitative results over SBI data set, and the global rank for each matrix completion method. The bold metric values show the best score for each metric. For each scene, the results are ordered by the rank column.

HallAndMonitor

Method	AGE?	EPs?	pEPs?	CEPs?	pCEPS?	MSSSIM?	PSNR?	CQM?	Scene Rank?
LRGeomCG	2.0550	**190**	**0.0022**	**0**	**0.0000**	**0.9938**	**37.9811**	**46.3255**	1
RMAMR	**2.0499**	**190**	**0.0022**	**0**	**0.0000**	**0.9938**	37.9755	46.3222	2
LMaFit	2.0583	194	0.0023	**0**	**0.0000**	0.9938	37.8487	46.2893	3
ScGrassMC	2.0693	193	0.0023	**0**	**0.0000**	0.9937	37.9046	46.2838	4
GROUSE	2.2201	191	0.0023	**0**	**0.0000**	0.9923	37.5319	45.8794	5
OR1MP	2.2025	374	0.0044	**0**	**0.0000**	0.9926	34.9021	44.8283	6
IALM	3.6143	2336	0.0277	1190	0.0141	0.9627	30.5907	40.9249	7
SVP	4.1486	3230	0.0382	1574	0.0186	0.9474	28.2921	41.4398	8
OptSpace	6.7051	5950	0.0704	2694	0.0319	0.9299	25.1834	37.6430	9

HighwayI

Method	AGE?	EPs?	pEPs?	CEPs?	pCEPS?	MSSSIM?	PSNR?	CQM?	Scene Rank?
LRGeomCG	2.7715	**192**	**0.0025**	**16**	**0.0002**	0.9769	**35.8950**	58.6193	1
RMAMR	**2.7601**	193	**0.0025**	**16**	**0.0002**	0.9769	35.8899	**58.6283**	1
LMaFit	2.7781	196	0.0026	**16**	**0.0002**	**0.9770**	35.8636	58.6193	3
ScGrassMC	3.2306	1088	0.0142	593	0.0077	0.9714	33.4838	58.5694	4
GROUSE	6.1324	621	0.0081	138	0.0018	0.9614	30.3643	56.2861	5
OR1MP	3.9587	1202	0.0157	691	0.0090	0.9637	32.4145	57.8051	6
IALM	6.4223	1836	0.0239	837	0.0109	0.9507	29.5142	57.9214	7
SVP	6.9160	5694	0.0741	2530	0.0329	0.9095	27.6621	53.1073	8
OptSpace	14.7067	19754	0.2572	13930	0.1814	0.7957	22.8624	43.8142	9

HighwayII

Method	AGE?	EPs?	pEPs?	CEPs?	pCEPS?	MSSSIM?	PSNR?	CQM?	Scene Rank?
LRGeomCG	**2.6840**	**268**	**0.0035**	4	0.0001	**0.9919**	**35.7076**	46.1997	1
RMAMR	2.7003	271	**0.0035**	5	0.0001	**0.9919**	35.6695	**46.2002**	2
LMaFit	2.6919	275	0.0036	7	0.0001	**0.9919**	35.5752	46.0673	3
ScGrassMC	2.9622	360	0.0047	**2**	**0.0000**	0.9888	34.6782	46.1585	4
IALM	4.9261	306	0.0040	**2**	**0.0000**	0.9830	31.5964	46.1296	5
OR1MP	3.2510	843	0.0110	102	0.0013	0.9888	32.2682	42.0740	6
SVP	4.7779	945	0.0123	153	0.0020	0.9813	30.4590	41.8017	7
GROUSE	4.3955	1751	0.0228	700	0.0091	0.9756	31.5542	45.6062	8
OptSpace	8.6231	4722	0.0615	1307	0.0170	0.9279	25.8722	36.7496	9

CaVignal

Method	AGE?	EPs?	pEPs?	CEPs?	pCEPS?	MSSSIM?	PSNR?	CQM?	Scene Rank?
LMaFit	**11.9504**	3788	0.1393	2700	0.0993	**0.9027**	**24.3417**	39.8279	1
LRGeomCG	11.9506	3789	0.1393	2700	0.0993	0.9026	24.3415	39.8279	2
RMAMR	12.0081	3817	0.1403	2715	0.0998	**0.9027**	24.3147	39.8083	3
GROUSE	12.8057	**3624**	**0.1332**	**2205**	**0.0811**	0.8846	23.2489	38.8799	4
ScGrassMC	12.3375	4084	0.1501	2942	0.1082	0.8916	23.9111	39.8237	5
IALM	12.2618	4764	0.1751	3531	0.1298	0.8779	23.7957	**40.6135**	6
OR1MP	12.5266	4455	0.1638	3356	0.1234	0.8855	23.7925	39.7716	7
SVP	13.2230	5628	0.2069	3982	0.1464	0.8760	23.3352	39.8300	8
OptSpace	14.1744	6176	0.2271	4214	0.1549	0.8927	23.0940	39.8662	9

Foliage

Method	AGE?	EPs?	pEPs?	CEPs?	pCEPS?	MSSSIM?	PSNR?	CQM?	Scene Rank?
GROUSE	26.4036	**17840**	**0.6194**	**13271**	**0.4608**	0.8957	18.4042	**33.4059**	1
LRGeomCG	**26.4006**	18074	0.6276	13459	0.4673	**0.8970**	**18.4522**	33.2733	2
LMaFit	26.4093	18070	0.6274	13442	0.4667	**0.8970**	18.4490	33.2663	2
RMAMR	26.5144	18146	0.6301	13548	0.4704	0.8965	18.4160	33.2392	4
ScGrassMC	29.2509	19189	0.6663	15385	0.5342	0.8645	17.4636	33.2173	5
OR1MP	31.3678	19094	0.6630	15257	0.5298	0.8221	16.7364	33.0468	5
OptSpace	32.1405	19156	0.6651	15234	0.5290	0.8656	16.6268	31.4830	7
IALM	31.6036	19451	0.6754	14986	0.5203	0.7473	16.8016	31.0114	8
SVP	35.3522	19469	0.6760	16003	0.5557	0.7556	15.6496	33.1273	9

PeopleAndFoliage

Method	AGE?	EPs?	pEPs?	CEPs?	pCEPS?	MSSSIM?	PSNR?	CQM?	Scene Rank?
LRGeomCG	**38.7467**	64150	0.8353	59210	0.7710	**0.8505**	**15.1638**	27.6167	1
OptSpace	41.8200	**60020**	**0.7815**	**53183**	**0.6925**	0.7535	14.1746	25.9930	2
IALM	42.9444	63161	0.8224	58560	0.7625	0.7951	14.1751	**27.8900**	2
GROUSE	40.3918	63556	0.8276	57097	0.7435	0.8130	14.6485	26.5857	4
OR1MP	42.8580	62335	0.8117	56809	0.7397	0.7861	14.0424	26.7015	4
ScGrassMC	39.6045	64076	0.8343	58991	0.7681	0.8472	14.9449	27.6499	6
LMaFit	38.7738	64155	0.8354	59220	0.7711	0.8501	15.1567	27.6143	7
RMAMR	38.8234	64189	0.8358	59276	0.7718	0.8502	15.1484	27.5987	8
SVP	45.0733	63916	0.8322	58803	0.7657	0.7644	13.6320	27.0416	9

Snellen

Method	AGE?	EPs?	pEPs?	CEPs?	pCEPS?	MSSSIM?	PSNR?	CQM?	Scene Rank?
ScGrassMC	43.8219	17838	0.8602	16070	0.7750	0.8469	14.3702	**37.6451**	1
LRGeomCG	41.8853	18621	0.8980	17379	0.8381	0.8807	14.8166	37.4253	2
OptSpace	49.2605	**16619**	**0.8015**	**15292**	**0.7375**	0.7419	12.8053	29.4492	2
LMaFit	41.8688	18623	0.8981	17387	0.8385	**0.8809**	14.8203	37.4234	4
GROUSE	**41.8123**	18629	0.8984	17403	0.8393	**0.8809**	**14.8421**	37.5570	5
IALM	46.0978	18433	0.8889	17084	0.8239	0.8330	14.0292	37.1551	6
OR1MP	50.4572	18084	0.8721	16677	0.8043	0.7504	13.1602	36.2566	7
SVP	54.6990	17649	0.8511	16189	0.7807	0.7105	12.5506	35.6896	7
RMAMR	42.0476	18625	0.8982	17395	0.8389	0.8800	14.7872	37.3821	9

Global rank over all scenes

Method	Global rank?
LRGeomCG	1
LMaFit	2
RMAMR	3
ScGrassMC	3
GROUSE	5
IALM	6
OR1MP	6
OptSpace	8
SVP	9

RMAMR have the first position for the AGE metric in the HallAndMonitor scene), next, 2) we sum the rank position value of each algorithm over the eight metrics, and finally, 3) we calculate the rank position over the sum, and we call it as *scene rank*. For the Global Rank, first we sum the scene rank for each MC algorithm, then we calculate its rank position over the sum. As we can see, the experimental results show the good performance of LRGeomCG [17] method over its direct competitors. Furthermore, in most cases the matrix completion algorithms outperform the traditional approaches such as Mean [9], Median [7] and MoG [22] as can be seen in the full experimental evaluation available at https://sites.google.com/site/mc4bmi/.

5 Conclusion

In this paper, we have evaluated nine recent matrix completion algorithms for the background initialization problem. Given a sequence of images, the key idea is to eliminate the redundant frames, and consider its moving regions as non-observed values. This approach results in a matrix completion problem, and the background model can be recovered even with the presence of missing entries. The experimental results on the SBI data set shows the comparative evaluation of these recent methods, and highlights the good performance of LRGeomCG [17] method over its direct competitors. Finally, MC shows a nice potential for background modeling initialization in video surveillance. Future research may concern to evaluate incremental and real-time approaches of matrix completion in streaming videos.

References

1. Balzano, L., Wright, S.J.: On GROUSE and incremental SVD. In: CAMSAP 2013, pp. 1–4 (2013). http://dx.doi.org/10.1109/CAMSAP.2013.6713992
2. Bouwmans, T.: Traditional and recent approaches in background modeling for foreground detection: An overview. Computer Science Review (2014)
3. Bouwmans, T., Zahzah, E.: Robust PCA via principal component pursuit: a review for a comparative evaluation in video surveillance. In: Special Isssue on Background Models Challenge, Computer Vision and Image Understanding. vol. 122, pp. 22–34, May 2014
4. Cai, J.F., Candès, E.J., Shen, Z.: A singular value thresholding algorithm for matrix completion. SIAM J. on Optimization **20**(4), 1956–1982 (2010)
5. Candès, E.J., Plan, Y.: Matrix completion with noise. CoRR abs/0903.3131 (2009)
6. Candès, E.J., Recht, B.: Exact matrix completion via convex optimization. CoRR abs/0805.4471 (2008). http://arxiv.org/abs/0805.4471
7. Cucchiara, R., Grana, C., Piccardi, M., Prati, A.: Detecting objects, shadows and ghosts in video streams by exploiting color and motion information. In: ICIAP 2001, pp. 360–365, September 2001
8. Keshavan, R.H., Montanari, A., Oh, S.: Matrix completion from noisy entries. The Journal of Machine Learning Research **99**, 2057–2078 (2010)
9. Lai, A.H.S., Yung, N.H.C.: A fast and accurate scoreboard algorithm for estimating stationary backgrounds in an image sequence. In: IEEE SCS 1998, pp. 241–244 (1998)

10. Lin, Z., Chen, M., Ma, Y.: The Augmented Lagrange Multiplier Method for Exact Recovery of Corrupted Low-Rank Matrices. Mathematical Programming (2010)
11. Maddalena, L., Petrosino, A.: Background model initialization for static cameras. In: Background Modeling and Foreground Detection for Video Surveillance. CRC Press, Taylor and Francis Group (2014)
12. Maddalena, L., Petrosino, A.: Towards benchmarking scene background initialization. CoRR abs/1506.04051 (2015). http://arxiv.org/abs/1506.04051
13. Meka, R., Jain, P., Dhillon, I.S.: Guaranteed rank minimization via singular value projection. CoRR abs/0909.5457 (2009)
14. Ngo, T., Saad, Y.: Scaled gradients on grassmann manifolds for matrix completion. Advances in Neural Information Processing Systems **25**, 1412–1420 (2012)
15. Oliver, N.M., Rosario, B., Pentland, A.P.: A bayesian computer vision system for modeling human interactions. IEEE PAMI **22**(8), 831–843 (2000)
16. Sobral, A., Vacavant, A.: A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. CVIU **122**, 4–21 (2014). http://www.sciencedirect.com/science/article/pii/S1077314213002361
17. Vandereycken, B.: Low-rank matrix completion by Riemannian optimization. SIAM Journal on Optimization **23**(2), 1214–1236 (2013)
18. Wang, Z., Lai, M., Lu, Z., Fan, W., Davulcu, H., Ye, J.: Orthogonal rank-one matrix pursuit for low rank matrix completion. SIAM J. Scientific Computing **37**(1) (2015). http://dx.doi.org/10.1137/130934271
19. Wen, Z., Yin, W., Zhang, Y.: Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm. Mathematical Programming Computation **4**(4), 333–361 (2012). http://dx.doi.org/10.1007/s12532-012-0044-1
20. Ye, X., Yang, J., Sun, X., Li, K., Hou, C., Wang, Y.: Foreground-background separation from video clips via motion-assisted matrix restoration. IEEE T-CSVT **PP**(99), 1 (2015)
21. Zhou, X., Yang, C., Zhao, H., Yu, W.: Low-rank modeling and its applications in image analysis. ACM Computing Surveys (CSUR) **47**(2), 36 (2014)
22. Zivkovic, Z.: Improved adaptive gaussian mixture model for background subtraction. In: ICPR 2004, vol. 2, pp. 28–31, Auguet 2004