# Adaptive Robust Principal Component Analysis

Yang Liu[a], Xinbo Gao[a], Quanxue Gao[a,*], Ling Shao[b], Jungong Han[c]

[a] *State Key Laboratory of Integrated Services Networks, Xidian University, Shaanxi 710071, China.*
[b] *Inception Institute of Artificial Intelligence, Abu Dhabi, United Arab Emirates.*
[c] *WMG Data Science, University of Warwick, CV4 7AL Coventry, United Kingdom*

**Abstract**

Robust Principal Component Analysis (RPCA) is a powerful tool in machine learning and data mining problems. However, in many real-world applications, RPCA is unable to well encode the intrinsic geometric structure of data, thereby failing to obtain the lowest rank representation from the corrupted data. To cope with this problem, most existing methods impose the smooth manifold, which is artificially constructed by the original data. This reduces the flexibility of algorithms. Moreover, the graph, which is artificially constructed by the corrupted data, is inexact and does not characterize the true intrinsic structure of real data. To tackle this problem, we propose an adaptive RPCA (ARPCA) to recover the clean data from the high-dimensional corrupted data. Our proposed model is advantageous due to: 1) The graph is adaptively constructed upon the clean data such that the system is more flexible. 2) Our model simultaneously learns both clean data and similarity matrix that determines the construction of graph. 3) The clean data has the lowest-rank structure that enforces to correct the corruptions. Extensive experiments on several datasets illustrate the effectiveness of our model for clustering and low-rank recovery tasks.

*Keywords:* `RPCA, Flexibility, Adaptively`

## 1. Introduction

Principal Component Analysis (PCA) is the most widely used tool for linear dimensionality reduction, image denoising and clustering. It aims to recover efficien-

*Corresponding author
*Email address:* `qxgao@xidian.edu.cn` (Quanxue Gao)

t representation with low-rank structure which is the best reconstruction in the least squared sense. However, in many machine learning and data mining problems, one often encounters high-dimensional samples with severe noise caused by corruptions or outliers, in which case the performance of PCA degenerates dramatically [1, 2, 3, 4, 5]. Thus, how to find an effective representation from the high-dimensional corrupted data has become an active topic in machine learning and information processing.

$\ell_1$-norm PCA, $\ell_{21}$-norm PCA and nuclear norm PCA are three of the most representative techniques to improve the robustness of PCA to outliers or noises. $\ell_1$-norm PCA and $\ell_{21}$-norm PCA aim to seek a robust projection matrix by solving the $\ell_1$-norm and $\ell_{21}$-norm optimization problems for different data types [1, 3, 6], where $\ell_1$-normal and $\ell_{21}$-normal are respectively employed as the distance metrics to characterize the variation among data in the criterion function. While they can well extract the robust low-dimensional representation for subsequent analysis such as classification and clustering, these methods are neither rotation invariant [1, 6] nor reconstruction error considered [3, 7].

Nuclear norm based PCA attempts to recover clean data with low-rank structure from the corrupted data so that the robustness of the PCA series mentioned above can be enhanced. Due to its great potential for being used in the real-life applications such as image denosing [8, 9],video surveillance [10] and image clustering [11, 12], this research has attracted a lot of attention from both academia and industry. Candes *et al.* [8] demonstrated that PCA can be made robust against outliers by exactly recovering the low-rank representation even from grossly corrupted data via solving a simple convex problem, named Robust PCA (RPCA). Vaswani *et al.* [13] and Bouwmans *et al.* [14] applied RPCA to background/foreground separation in video-surveillance. As some RPCA based methods suffer from large memory requirement and high computational complexity, some online algorithms [15, 16] and real-time algorithms [17, 18] are recently proposed to improve the efficiency. In addition, when the size of the input data grows and due to the lack of sparsity constraints, some RPCA based methods cannot cope with the real-time challenges and always show a weak performance in video processing task. In order to address the above mentioned issues, Javed*et al.* [19] proposed an efficient and reliable low-rank component using matrix decomposition with

max-norm of super pixels.

In order to effectively deal with color images or high-order images, RPCA was extended to tensor RPCA [20]. In [21], Oh *et al.* noticed that the performance of RPCA drops considerably when the number of data is small. To solve this problem, they proposed a method named PSSV to minimize partial sum of singular values of data. RPCA, PSSV, and TRPCA, however, do not take the local structure into account. To tackle this problem, Jiang *et al*. [22] proposed Graph Laplacian PCA (GLPCA) which integrates the graph regularization of principal components into the criterion function of PCA. They also developed a robust version of GLPCA (RGLPCA). But both of them suffer from non-convexity and the resulting alternating direction method can get stuck in local minima. Zhang and Zhao [23] proposed manifold regularized matrix factorization (MMF) which imposes the orthonormality constraint on the projection directions and integrates the graph regularization of low-dimensional representation to learn a low-rank representation. The extension works of MMF were proposed by Tao *et al*. [24] and Jin *et al*. [25]. However, they are not robust to data corruptions and suffer from non-convexity [9]. To handle it, recently, Shahid *et al*. [9] proposed Graph-RPCA which integrates graph regularization of clean data with low-rank structure into the objective function of RPCA.

In general, smoothness manifold regularization in the aforementioned methods heavily depends on the graph, which is artificially constructed on the corruption data. This reduces the flexibility of algorithm due to the complex and unknown distribution of data. Furthermore, the corrupted data do not seem to characterize the true geometric structure of real data well. Therefore, the graph, which is constructed on the corrupted data, is inexact and may make algorithm degenerate obviously in real applications. Finally, all of the aforementioned robust PCA methods based on nuclear norm cannot obtain the lowest-rank representation of the entire data jointly [26]. Thus, they cannot well characterize the global and local geometric structure of data.

To tackle the aforementioned problem, in this paper, we propose an adaptive RPCA to recover the clean data which improves the stability of RPCA in clustering and image denoising. Our method adaptively constructs graph on the clean data in the sense that the weight attached to each edge on the graph is learned from the data, rather than an

empirical value used in the existing algorithm. Doing so, on the one hand, will definitely improve the graph quality and eventually lead to better algorithm performance. On the other hand, learning weights, instead of assigning them manually, will make the system more flexible, especially when changing the application. Moreover, our model simultaneously learns both clean data with low-rank structure and similarity matrix. This enables to capture the global geometric structure, and meanwhile, preserve the local intrinsic structure. Extensive experimental results for image denoising, clustering and extraction background demonstrate that our proposed model is more robust to outliers and missing values, as compared to the state-of-the-art methods.

## 2. Related Works

In this section, we briefly review some works closely related to our experiments and proposed model.

### *2.1. Background Extraction and Hyperspectral Image Processing*

In the recent years, RPCA based approaches such as Dynamic Super Pixel Structured-Sparse (DSPSS) [27] and Motion Aware Graph regulerized RPCA (MAG-RPCA) [28] have gained some popularity due to their computational simplicity and effectiveness in background extraction. However, these models ignore the spatial distribution of outliers. Shape and Confidence Mapbased RPCA (SCM-RPCA) [29] was proposed to improve the background extraction in maritime scenes, where the sparse component is constrained by shape and confidence maps both extracted from spatial saliency maps. For hyperspectral image processing, Cheng *et al*. [30] proposed a novel method that combines PCA and LDA method to maximizes the representation and classification effects on the extracted new feature bands. Xu *et al*. [31] proposed a novel tensor RPCA method to decompose the original hyperspectral image into background and anomaly parts by Mahalanobis distance regularization. In [32], authors gave an hyperspectral image mixed-noise removal method by simultaneously exploiting the local low-rank structure and the global spatial-spectral piecewise smoothness. Recently, NonLRMA [32] was proposed to decompose the degraded hyperspectral image into a low rank component and a sparse term with a more robust and less biased formulation.

### 2.2. Robust Principal Component Analysis (RPCA)

Robust principal component analysis (RPCA) [8], which is one of the most popular robust methods, aims to recover a low-rank matrix $\mathbf{D} \in R^{m\times n}$ from corrupted observations $\mathbf{X} = \mathbf{D} + \mathbf{E}$, where $\mathbf{E} \in R^{m\times n}$ represents errors with arbitrary magnitude and distribution. The rank minimization approach assumes $\mathbf{E}$ is sparse and formulates the problem as

$$\min_{\mathbf{D},\mathbf{E}} \ rank\left(\mathbf{D}\right) + \lambda\|\mathbf{E}\|_0 \qquad s.t.\ \mathbf{X} = \mathbf{D} + \mathbf{E} \tag{1}$$

where $rank\left(\mathbf{D}\right)$ is the rank of matrix $\mathbf{D}$. The $\|\cdot\|_0$ is the pseudo-norm, i.e., the number of nonzero elements in the matrix. $\lambda$ is a positive penalty parameter for trading off between the low rank term and sparse term. This optimization model is a NP-hard problem, which is usually transformed into the model (2) in real applications [8, 33].

$$\min_{\mathbf{D},\mathbf{E}} \ \|\mathbf{D}\|_* + \lambda\|\mathbf{E}\|_1 \qquad s.t.\ \mathbf{X} = \mathbf{D} + \mathbf{E} \tag{2}$$

where $\|\mathbf{D}\|_* = \sum_i \sigma_i$ is nuclear norm of $\mathbf{D}$, $\sigma_i$ denotes the $i$th singular value of $\mathbf{D}$ (sorted in decreasing order).

It can be seen in the model (2), RPCA only imposes the low-rank constraint on clean data but ignores the relationship among columns of matrix $\mathbf{X}$ which characterizes the geometric structure of data. To well characterize the geometric structure of data, motivated by the fact that the performance of RPCA can be significantly improved by smoothness manifold regularization, many enhanced methods have been developed, among which RPCAG [9] is one of the representative method. Its objective function is

$$\min_{\mathbf{D},\mathbf{E}} \|\mathbf{D}\|_* + \lambda\|\mathbf{E}\|_1 + \gamma\mathrm{tr}(\mathbf{D}\mathbf{\Phi}\mathbf{D}^T) \quad \text{s.t. } \mathbf{X}{=}\mathbf{D}{+}\mathbf{E} \tag{3}$$

where parameters $\lambda$ and $\gamma$ control the amount of sparsity of $\mathbf{E}$ and smoothness of $\mathbf{D}$ on the graph $\mathbf{\Phi}$ respectively.

In the model (3), graph is usually artificially constructed based on the corrupted data. **This results in the following limitations**: First, it reduces the flexibility of algorithm due to the complex and unknown distribution of data. Second, the corrupted data do not characterize the true geometric structure of real data. So, the graph, which is constructed on the corrupted data, is inexact and may make algorithm degenerate

dramatically in real applications. Third, constructing graph is independent of the clean data, thus making RPCAG impossible to obtain lowest-rank structure which characterizes the global structure of data.

### *2.3. Low-rank Representation (LRR)*

Assume that we have data matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N] \in R^{d \times N}$, which are drawn from a union of $k$ subspace. LRR [26] aims to seek the lowest-rank representation of data $\mathbf{X}$ with respect to given dictionary $\mathbf{B}$. This formulates the following convex optimization problem.

$$\min_{\mathbf{Z}} \ \|\mathbf{Z}\|_* \quad s.t. \ \mathbf{X} = \mathbf{B}\mathbf{Z} \tag{4}$$

The optimal solution $\mathbf{Z}$ of the problem (4) is called the lowest-rank representations of data matrix $\mathbf{X}$. In order to well characterize the intrinsic structure of each subspace, the data itself $\mathbf{X}$ is used as the dictionary in LRR. In this case, the problem (4) becomes

$$\min_{\mathbf{Z}} \ \|\mathbf{Z}\|_* \quad s.t. \ \mathbf{X} = \mathbf{X}\mathbf{Z} \tag{5}$$

In real applications, the observed data $\mathbf{X}$ are often noisy or even grossly corrupted. To handle the noise or model errors, a more reasonable objective might be:

$$\min_{\mathbf{Z},\mathbf{E}} \ \|\mathbf{Z}\|_* + \lambda\|\mathbf{E}\|_{2,1} \quad s.t. \ \mathbf{X} = \mathbf{X}\mathbf{Z} + \mathbf{E} \tag{6}$$

where the parameter $\lambda > 0$ is used to balance the effects of the two parts, which could be tuned empirically.

Different from RPCA, LRR represents each data vector as a linear combination of the other data vectors. Thus, LRR obtains the lowest-rank representation $\mathbf{Z}$ which well characterizes local intrinsic geometric structure. According to the matrix theory, we get $\mathbf{X}\mathbf{Z}$ also the lowest-rank structure, which well captures global geometric structure. However, **a critical shortcoming of LRR is that**, when the clean data is insufficient, it cannot obtain the good low-rank representation which captures the geometric structure of data.

## 3. Adaptive Robust Principal Component Analysis (ARPCA)

### *3.1. Objective function*

We aim to recover clean data $\mathbf{D}$ from the corruption data $\mathbf{X}$ such that $\mathbf{D}$ has the lowest-rank structure and simultaneously captures global and local geometric structures of data. According to the aforementioned analysis, RPCA cannot obtain clean data $\mathbf{D}$ with the lowest-rank structure due to the fact that it does not take the membership of the samples into account. Moreover, motivated by LRR, if we represent each column vector in $\mathbf{D}$, which is clean data, as a linear combination of the other column vectors in $\mathbf{D}$, i.e, $\mathbf{D} = \mathbf{DS}$, then $\mathbf{S}$ can well reveal the membership of the column vectors of $\mathbf{D}$ with the nuclear norm minimization. In real applications, the constraint $\mathbf{D} = \mathbf{DS}$ is very strict, which may result in over-fitting. Thus, we relax the constraint for $\mathbf{D} \approx \mathbf{DS}$. Inspired by the fact that $\ell_1$-norm helps improve the robustness, we use $\|\mathbf{D} - \mathbf{DS}\|_1$, which constructs a graph as in manifold learning [34], to measure the representation error.

The proposed model is

$$\begin{aligned} \min_{\mathbf{D},\mathbf{E},\mathbf{S}} \ & \|\mathbf{D}\|_* + \lambda_1\|\mathbf{E}\|_1 + \lambda_2\left(\lambda_3\|\mathbf{D} - \mathbf{DS}\|_1 + \|\mathbf{S}\|_*\right) \\ & s.t.\ \mathbf{X} = \mathbf{D} + \mathbf{E} \end{aligned} \tag{7}$$

Our approach has three advantages. (1) The proposed model adaptively constructs graph on the clean data. This improves the flexibility of our model. (2) The proposed model simultaneously learns both clean data with low-rank structure and similarity matrix that determines the construction of graph. (3) The proposed model represents each data vector of clean data as a linear combination of clean data. This helps obtain a lowest-rank representation to correct corruptions and to capture the global geometric structure.

### *3.2. Algorithm*

In this section, we propose an efficient iterative algorithm to solve the problem (7). By simple algebra, an ADMM (Alternating Direction Method of Multipliers) [35] is

used to rewrite Problem (7) as

$$\begin{aligned} \min_{\substack{\mathbf{D}_1,\mathbf{E},\mathbf{E}_1,\\ \mathbf{S}_1,\mathbf{D},\mathbf{S}}} \quad & \|\mathbf{D}_1\|_* + \lambda_1\|\mathbf{E}\|_1 + \lambda_2\left(\lambda_3\|\mathbf{E}_1\|_1 + \|\mathbf{S}_1\|_*\right) \\ s.t. \quad & \mathbf{X} = \mathbf{D} + \mathbf{E},\ \ \mathbf{D} = \mathbf{D}_1,\ \ \mathbf{E}_1 = \mathbf{D} - \mathbf{DS},\ \ \mathbf{S} = \mathbf{S}_1 \end{aligned} \tag{8}$$

Thus, the augmented Lagrangian and iterative scheme are

$$\begin{aligned} & L\left(\mathbf{D}_1, \mathbf{E}, \mathbf{E}_1, \mathbf{S}_1, \mathbf{D}, \mathbf{S}\right) \\ & = \arg\min_{\substack{\mathbf{D},E,S,\\ D_1,\mathbf{E}_1,\mathbf{S}}} \|\mathbf{D}_1\|_* + \lambda_1\|\mathbf{E}\|_1 + \lambda_2\left(\lambda_3\|\mathbf{E}_1\|_1 + \|\mathbf{S}_1\|_*\right) \\ & \quad + tr\left({\mathbf{Y}_1}^T\left(\mathbf{X} - \mathbf{D} - \mathbf{E}\right)\right) + tr\left({\mathbf{Y}_2}^T\left(\mathbf{D} - \mathbf{D}_1\right)\right) \\ & \quad + tr\left({\mathbf{Y_3}}^T\left(\mathbf{D} - \mathbf{DS} - \mathbf{E}_1\right)\right) \ + tr\left({\mathbf{Y_4}}^T\left(\mathbf{S} - \mathbf{S}_1\right)\right) \\ & \quad + \tfrac{\mu}{2}\left(\|\mathbf{X} - \mathbf{D} - \mathbf{E}\|_F^2 + \|\mathbf{D} - \mathbf{D}_1\|_F^2\right) \\ & \quad + \tfrac{\mu}{2}\left(\|\mathbf{D} - \mathbf{DS} - \mathbf{E}_1\|_F^2 + \|\mathbf{S} - \mathbf{S}_1\|_F^2\right) \end{aligned} \tag{9}$$

where $\mathbf{Y}_1$, $\mathbf{Y}_2$, ,$\mathbf{Y}_3$, and $\mathbf{Y}_4$ are lagrange multipliers, and $\mu > 0$ is a penalty parameter.

Step 1: [Update $\mathbf{D}_1$]. In this case, the other variables are fixed. Thus, the problem Eq (9) becomes

$$\begin{aligned} {\mathbf{D}_1}^* &= \arg\min_{\mathbf{D}_1} \|\mathbf{D}_1\|_* + tr\left({\mathbf{Y}_2}^T\left(\mathbf{D} - \mathbf{D}_1\right)\right) + \tfrac{\mu}{2}\|\mathbf{D} - \mathbf{D}_1\|_F^2 \\ &= \arg\min_{\mathbf{D}_1} \tfrac{1}{2\mu}\|\mathbf{D}_1\|_* + \tfrac{1}{2}\|\mathbf{D}_1 - \mathbf{\Gamma}\|_F^2 \\ &= \Omega_{1/(2\mu)}\left(\mathbf{\Gamma}\right) \end{aligned} \tag{10}$$

where $\mathbf{\Gamma} = \mathbf{D} + \mathbf{Y}_2/\mu$, $\Omega_{1/(2\mu)}\left(\mathbf{\Gamma}\right) = \mathbf{U}R_{1/(2\mu)}\left[\sum\right]\mathbf{V}$ is the singular value shrinkage operator [36], $\mathbf{U}\sum\mathbf{V}$ is the SVD of $\mathbf{\Gamma}$, and $R_\varepsilon\left[x\right] = \mathrm{sgn}(x)max(|x| - \varepsilon, 0)$ is the scalar shrinkage operator.

Step 2: [Update $\mathbf{E}$ ]. In this case, the problem Eq (9) becomes

$$\begin{aligned} \mathbf{E}^* &= \arg\min_{\mathbf{E}} \lambda_1\|\mathbf{E}\|_1 + tr\left({\mathbf{Y}_1}^T\left(\mathbf{X} - \mathbf{D} - \mathbf{E}\right)\right) + \tfrac{\mu}{2}\|\mathbf{X} - \mathbf{D} - \mathbf{E}\|_F^2 \\ &= \arg\min_{\mathbf{E}} \tfrac{\lambda_1}{\mu}\|\mathbf{E}\|_1 + \tfrac{1}{2}\|\mathbf{E} - \mathbf{\Pi}\|_F^2 \\ &= R_{\lambda_1/\mu}\left[\mathbf{\Pi}\right] \end{aligned} \tag{11}$$

where $\mathbf{\Pi} = \mathbf{X} - \mathbf{D} + \frac{\mathbf{Y}_1}{\mu}$.

Step 3: [Update $\mathbf{E}_1$]. The problem Eq (9) becomes

$$\begin{aligned}\mathbf{E}_1{}^* &= \arg\min_{\mathbf{E}_1} \lambda_2\lambda_3\|\mathbf{E}_1\|_1 + tr\left(\mathbf{Y}_3{}^T\left(\mathbf{D}-\mathbf{DS}-\mathbf{E}_1\right)\right) + \tfrac{\mu}{2}\left\|\mathbf{D}-\mathbf{DS}-\mathbf{E}_1\right\|_F^2 \\ &= \arg\min_{\mathbf{E}_1} \tfrac{\lambda_2\lambda_3}{\mu}\|\mathbf{E}_1\|_1 + \tfrac{1}{2}\left\|\mathbf{E}_1-\mathbf{Z}\right\|_F^2 \\ &= R_{\lambda_2\lambda_3/\mu}\left(\mathbf{Z}\right)\end{aligned} \tag{12}$$

where $\mathbf{Z} = \mathbf{D} - \mathbf{DS} + \frac{\mathbf{Y}_3}{\mu}$.

Step 4: [Update $\mathbf{S}_1$]. In this case, the problem Eq (9) becomes

$$\begin{aligned}\mathbf{S}_1{}^* &= \arg\min_{\mathbf{S}_1} \lambda_2\|\mathbf{S}_1\|_* + tr\left(\mathbf{Y_4}{}^T\left(\mathbf{S}-\mathbf{S}_1\right)\right) + \tfrac{\mu}{2}\left\|\mathbf{S}-\mathbf{S}_1\right\|_F^2 \\ &= \arg\min_{\mathbf{S}_1} \tfrac{\lambda_2}{\mu}\|\mathbf{S}_1\|_* + \tfrac{1}{2}\left\|\mathbf{S}_1-\mathbf{\Psi}\right\|_F^2 \\ &= \Omega_{\lambda_2/\mu}\left[\mathbf{\Psi}\right]\end{aligned} \tag{13}$$

where $\mathbf{\Psi} = \mathbf{S} + \frac{\mathbf{Y_4}}{\mu}$.

Step 5: [Update $\mathbf{D}$ ]. The problem Eq (9) becomes

$$\begin{aligned}\mathbf{D}^* = \arg\min_{\mathbf{D}}\ & tr(\mathbf{Y}_1{}^T(\mathbf{X}-\mathbf{D}-\mathbf{E})) + \tfrac{\mu}{2}\left\|\mathbf{D}-\mathbf{D}_1\right\|_F^2 \\ & + tr(\mathbf{Y}_2{}^T(\mathbf{D}-\mathbf{D}_1)) + \tfrac{\mu}{2}\left\|\mathbf{D}-\mathbf{DS}-\mathbf{E}_1\right\|_F^2 \\ & + tr(\mathbf{Y_3}{}^T(\mathbf{D}-\mathbf{DS}-\mathbf{E}_1)) + \tfrac{\mu}{2}\left\|\mathbf{X}-\mathbf{D}-\mathbf{E}\right\|_F^2\end{aligned} \tag{14}$$

Taking the derivative of Eq. (14) with respective to $\mathbf{D}$ and setting it to zero, we have

$$\mathbf{D}^* = \mathbf{K}\left(3\mathbf{I}-\mathbf{S}-\mathbf{S}^T+\mathbf{S}\mathbf{S}^T\right)^{-1} \tag{15}$$

where $\mathbf{K} = \mathbf{X}+\mathbf{D}_1+\mathbf{E}_1-\mathbf{E}-\mathbf{E}_1\mathbf{S}^T+\frac{\mathbf{Y}_1-\mathbf{Y}_2-\mathbf{Y_3}+\mathbf{Y_3}\mathbf{S}^T}{\mu}$. $\mathbf{I}$ is the identity matrix.

Step 6: [Update $\mathbf{S}$ ]. The problem Eq (9) becomes

$$\begin{aligned}\mathbf{S}^* = \ & \arg\min_{\mathbf{S}} tr(\mathbf{Y_3}{}^T\left(\mathbf{D}-\mathbf{DS}-\mathbf{E}_1\right)) + \tfrac{\mu}{2}\left\|\mathbf{D}-\mathbf{DS}-\mathbf{E}_1\right\|_F^2 \\ & + tr\left(\mathbf{Y_4}{}^T\left(\mathbf{S}-\mathbf{S}_1\right)\right) + \tfrac{\mu}{2}\left\|\mathbf{S}-\mathbf{S}_1\right\|_F^2\end{aligned} \tag{16}$$

Taking the derivative of Eq. (16) with respective to $\mathbf{S}$ and setting it to zero, we have

$$\mathbf{S}^* = \left(\mathbf{D}^T\mathbf{D}+\mathbf{I}\right)^{-1}\mathbf{P} \tag{17}$$

where $\mathbf{P} = \mathbf{D}^T\mathbf{D}+\mathbf{S}_1-\mathbf{D}^T\mathbf{E}_1+\frac{\mathbf{D}^T\mathbf{Y_3}-\mathbf{Y_4}}{\mu}$. Algorithm 1 lists the pseudo code for solving our model (9).

**Algorithm 1** Algorithm to solve the model (9)

**Input:** Data matrix $\mathbf{X}$, parameter $\lambda_1$, $\lambda_2$, $\lambda_3$.

**Initialize** $\mathbf{D} = \mathbf{S} = \mathbf{E} = \mathbf{0}$, $\mathbf{Y}_1 = \mathbf{Y}_2 = \mathbf{Y}_3 = \mathbf{Y}_4 = \mathbf{0}$, $\mu = 0.1$, $\mu_{\max} = 10^6$, $\rho = 1.1$, $\varepsilon = 10^{-6}$.

**while** not converge **do**

1. Update $\mathbf{D}_1$, $\mathbf{E}$, $\mathbf{E}_1$, $\mathbf{S}_1$, $\mathbf{D}$, $\mathbf{S}$ using Eq. (10), Eq. (11), Eq. (12), Eq. (13), Eq. (15) and Eq. (17) respectively.
2. Update $\mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3, \mathbf{Y}_4$ and $\mu$:

$$\begin{aligned}
\mathbf{Y}_1 &= \mathbf{Y}_1 + \mu\left(\mathbf{X} - \mathbf{D} - \mathbf{E}\right) \\
\mathbf{Y}_2 &= \mathbf{Y}_2 + \mu\left(\mathbf{D} - \mathbf{D}_1\right) \\
\mathbf{Y}_3 &= \mathbf{Y}_3 + \mu\left(\mathbf{D} - \mathbf{DS} - \mathbf{E}_1\right) \\
\mathbf{Y}_4 &= \mathbf{Y}_4 + \mu\left(\mathbf{S} - \mathbf{S}_1\right) \\
\mu &= \min\left(\rho * \mu, \mu_{\max}\right)
\end{aligned}$$

3. Check the convergence conditions:

$\|\mathbf{X} - \mathbf{D} - \mathbf{E}\|_\infty < \varepsilon$, $\|\mathbf{D} - \mathbf{D}_1\|_\infty < \varepsilon$,

$\|\mathbf{D} - \mathbf{DS} - \mathbf{E}_1\|_\infty < \varepsilon$, and $\|\mathbf{S} - \mathbf{S}_1\|_\infty < \varepsilon$.

**end while**

**Output:** $\mathbf{D}, \mathbf{E}, \mathbf{S}$

## 4. Experiments

In this section, we validate our proposed method on Scene Background Initialization (SBI) database [37, 38], the Airborne Visible/Infrared Imaging Spectrometer (i.e. AVIRIS) Indian Pines image [39], and ORL database [40], and compare with three representative methods, i.e., RPCA [8], RPCAG [9], PSSV [21] PCPS [41], IMBS-MT [42], LaBGen [43] and BEWIS [44].

### *4.1. Low-Rank Background Extraction from Videos*

Scene Background Initialization (SBI) database is used to extract background. In the experiment, we use CAVIAR1 and HighwayI image sequences as sub-dataset.

CAVIAR1 contains 610 frames with a resolution of $384 \times 256$. HighwayI contains 410 frames with a resolution of $320 \times 240$. For CAVIAR1, we randomly select 14, 21, 42, 63 frames as training data, and then resize each frame to $96 \times 64$ pixels. In the experiments, parameters are set as follows: $\lambda_1 = 4.1/\sqrt{\max(m, N)}$, $\lambda_2 = 1$ and $\lambda_3 = 10/\sqrt{\max(m, N)}$. For HighwayI, we randomly select 5, 10, 20, 30 frames and resize each frame to $80 \times 60$ pixels. Parameters are set as $\lambda_1 = 3.5/\sqrt{\max(m, N)}$, $\lambda_2 = 1.5$ and $\lambda_3 = 3/\sqrt{\max(m, N)}$. Results of CAVIAR1 and HighwayI image sequences are respectively shown in Figure 1 and Figure 2.

As can be seen in Figure 1 and Figure 2, we have following results: First, RPCAG is overall superior to RPCA. This is due to the fact that RPCAG takes the relationship of samples into account. Secondly, our method and PSSV are obviously superior to the other two methods RPCA and RPCAG when the number of training data is insufficient. This is probably due to that RPCA and RPCAG do not fully utilize a priori target rank information. In Figure 2, RPCAG and RPCA have almost the same result. The reason may be that the graph artificially constructed on the original data does not well capture the intrinsic structure. Third, our model is superior to RPCA and RPCAG, because our method adaptively constructs the graph which well reveals the intrinsic geometric structure. Moreover, our model obtains the low-rest-rank representation that enforces to exactly extract background. Fourth, our method has almost the same results with PSSV. It illustrates that our method obtains the lowest-rank clean data because the target rank is set 1 in PSSV.

### *4.2. Hyperspectral Image Denoising*

The AVIRIS Indian Pines [39] was collected by the AVIRIS sensor on the Indian Pines region, Northwest Indiana, USA, in 1992. The scene was acquired over a mixed forest/agricultural area, with a size of $145 \times 145 \times 224$. There exist 224 bands across the spectral range from 0.2 to $2.5um$, nominal spectral resolution of 10nm. The image has a spatial resolution of 20 m per pixel and 16-bit radiometric resolution. It includes 16 classes, most of which are different types of crops (e.g., corns, soybeans, and wheats). For the preconditioning of the data, the gray values of each band of the HSI are normalized between [0, 1]. In the experiments, we randomly placed

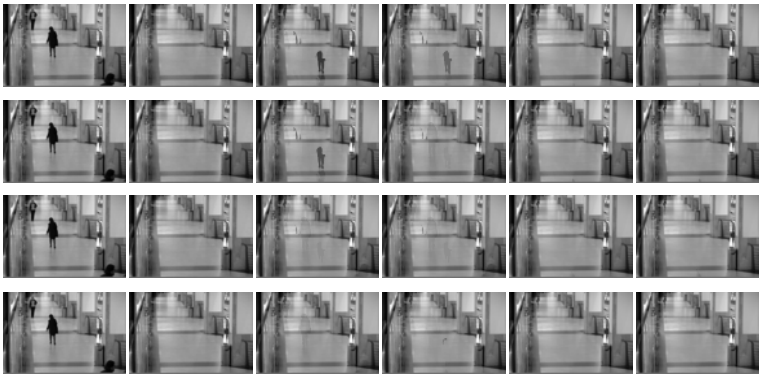

Figure 1: The recovered background from CAVIAR1. From the first row to the fourth row are the results of the 14, 21, 42, and 63 frames, respectively. From left to right are the original frame, reference background, recovered background by RPCA [8], RPCAG [9], PSSV [21] and ARPCA, respectively.

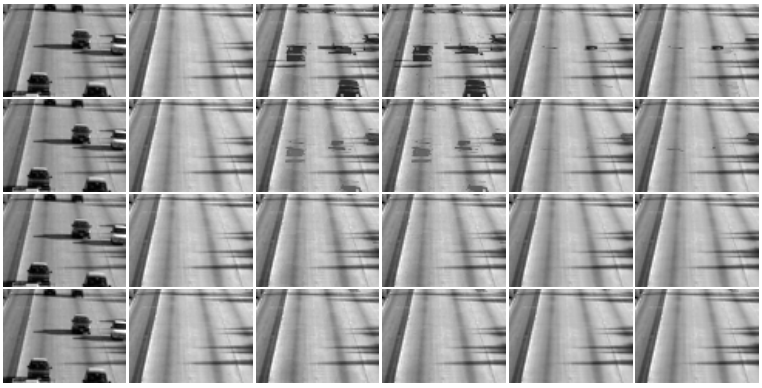

Figure 2: The recovered background from HighwayI. From the first row to the fourth row are the results of the 5, 10, 20, and 30, respectively. From left to right are the original frame, reference background, recovered background by RPCA [8], RPCAG [9], PSSV [21] and ARPCA, respectively.

$10\%$, $20\%$, $30\%$ ,$40\%$ noise which is the type of salt and pepper, and set parameters as $\lambda_1 = 1.5/\sqrt{\max(m,N)}$, $\lambda_2 = 1$, and $\lambda_3 = 0.8/\sqrt{\max(m,N)}$.

We use peak signal-to-noise ratio (PSNR) and structure similarity (SSIM) index to assess the performance of each method for image denoising. For HSI, we compute the value of the aforementioned two indices on different spectral bands, and then calculate the mean values of these bands, which is denoted by Mean-PSNR and Mean-SSIM, respectively. The higher value of PSNR and SSIM means the better performance of

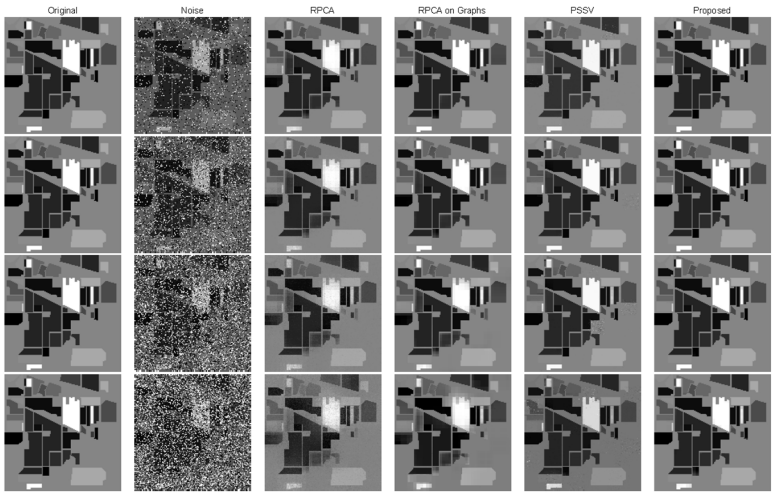


Figure 3: Denoised images of Indian Pines in band 1. From the first row to the fourth row are 10% noise, 20% noise, 30% noise and 40% noise respectively. From left to right are Original Image, noised image with different percentage (10%, 20%, 30%, 40%) noise, recovered image by RPCA [8], RPCAG [9], PSSV [21] and ARPCA, respectively.

method. PSNR and SSIM are defined as follows:

$$PSNR_i = 10 * \log_{10} \frac{MN}{\sum_{x=1}^{M}\sum_{y=1}^{N}[\hat{u}_i(x,y)-u_i(x,y)]^2}$$

$$Mean - PSNR = \frac{1}{B}\sum_{i=1}^{B} PSNR_i$$

$$SSIM_i = \frac{\left(2u_{u_i}u_{\hat{u}_i}+C_1\right)\left(2\sigma_{u_i}\sigma_{\hat{u}_i}+C_2\right)}{\left({u_{u_i}}^2+{u_{\hat{u}_i}}^2+C_1\right)\left({\sigma_{u_i}}^2+{\sigma_{\hat{u}_i}}^2+C_2\right)}$$

$$Mean - SSIM = \frac{1}{B}\sum_{i=1}^{B} SSIM_i$$

where $u_i$ and $\hat{u}_i$ represent the $ith$ band of the reference image and restored image, respectively. $u_{u_i}$ and $u_{\hat{u}_i}$ are the average values of image $u_i$ and $\hat{u}_i$, while $\sigma_{u_i}$ and $\sigma_{\hat{u}_i}$ are variances. $M$ and $N$ are the height and width in the spatial region, respectively. Moreover, $B$ is the number of bands in spectrum region. $C_1$ and $C_2$ are constants.

Figure 3 shows the denosing images that is obtained by the aforementioned four methods. Table 1 and Table 2 list the average PSNR and SSIM of each method. As can

Table 1: The Mean-PSNR of seven methods under different noise level on Indian Pines.

| Noise level | 10% | 20% | 30% | 40% |
|---|---|---|---|---|
| RPCA [8] | 42.4471 | 37.4339 | 31.6359 | 26.5342 |
| RPCAG [9] | 54.6857 | 47.7350 | 41.2454 | 34.6076 |
| PSSV [21] | 47.2177 | 45.1279 | 44.5750 | 41.2198 |
| IMBS-MT [42] | 50.2075 | 46.2346 | 40.9631 | 30.5326 |
| LaBGen [43] | 61.6973 | 53.3651 | 46.2416 | 42.7621 |
| BEWIS [44] | 67.2333 | 61.5326 | 55.2930 | 46.5012 |
| ARPCA | **71.2093** | **67.6107** | **64.8939** | **52.2474** |

Table 2: The Mean-SSIM(%) of seven methods under different noise level on Indian Pines.

| Noise level | 10% | 20% | 30% | 40% |
|---|---|---|---|---|
| RPCA [8] | 99.48 | 98.49 | 93.45 | 82.08 |
| RPCAG [9] | 99.95 | 99.77 | 99.18 | 97.26 |
| PSSV [21] | 99.65 | 99.35 | 99.17 | 98.34 |
| IMBS-MT [42] | 99.55 | 99.01 | 96.73 | 93.03 |
| LaBGen [43] | 99.93 | 99.22 | 98.91 | 97.60 |
| BEWIS [44] | 99.98 | 99.86 | 99.21 | 98.21 |
| ARPCA | **100** | **99.99** | **99.99** | **99.89** |

be seen in Figure 3, Table 1 and table 2, we have that, RPCA is inferior to the other three methods. The reason is due to the fact that RPCA does not take the relationship of samples or utilize a priori target rank information. RPCAG is inferior to PSSV when the images contain $40\%$ noise. The reason is probably because that the graph on the corrupted data does not reveal the true geometric structure of data. Our model achieves

the best denoising results and has the best PSNR and SSIM. This is due to the fact that our model simultaneously optimizes clean data and similarity matrix that determines the graph. Another reason is that our method achieves the lowest-rank clean data and well reveals geometric structure of data.

### *4.3. Data Clustering*

The ORL database contains ten different images of each of 40 distinct subjects with the resolution $112 \times 92$. For some subjects, the images are taken at different times, varying the lighting, facial expressions (open/closed eyes, smiling) and facial details (glasses). All the images are taken against a dark homogeneous background with the subjects in an upright, frontal position (with tolerance for some side movement). In the experiment, each image was normalized to a size of $56 \times 46$ to construct a new gallery. In the new gallery, we randomly placed 10%, 20%, and 30% black and white dots in each image, and set parameters as $\lambda_1 = 0.8/\sqrt{\max(m, N)}$, $\lambda_2 = 3.5$, and $\lambda_3 = 1.5/\sqrt{\max(m, N)}$. All of experiments are repeated 3 times. Table 3 lists the average clustering error.

Table 3: Average clustering error(%) and standard deviation on the ORL dataset.

| Noise level | RPCA [8] | RPCAG [9] | PSSV [21] | PCPS [41] | ARPCA |
|---|---|---|---|---|---|
| 10% | 34.67±1.91 | 31.42±2.08 | 32.75±1.75 | 32.21±2.00 | **30.83±0.76** |
| 20% | 35.33±2.50 | 31.00±0.75 | 33.17±1.28 | 30.66±2.00 | **29.00±1.09** |
| 30% | 36.92±1.38 | 34.33±1.38 | 32.42±1.04 | 33.29±0.93 | **30.67±1.66** |

As can be seen in table 3, RPCA is inferior to the other three methods for clustering. RPCAG is superior to PSSV in most cases. The reason is probably because that it is difficult to exactly select the target rank. the performance of RPCAG is not good when noise accounts for 30% of data. The reason is due to the fact that graph, which is artificially constructed on the corrupted data, does not well reveal geometric structure of data. Our model is superior to the other methods. This is due to the fact that our

model achieves the lowest-rank clean data, which enforces to correct noise, and well reveals geometric structure of data.

### *4.4. Complexity and Convergence Analysis*

That $\mathbf{D}_1$ and $\mathbf{S}_1$ update in Step 1 is the most costly step of each iteration in Algorithm 1, which requires computing the SVD of a matrix. For a matrix in $\Re^{m\times n}$, the exact SVD has a computational complexity of $\mathrm{O}(min(m^2n, n^2m))$. Thus, for updating $\mathbf{D}_1$, the computational complexity is $\mathrm{O}(min(m^2n, n^2m))$. Similarly, for updating $\mathbf{S}_1$, the computational complexity is $\mathrm{O}(min(n^3))$. In addition, for ADMM method, its convergence has also been well studied when the number of blocks is at most two [45, 46]. However, theoretically ensuring the convergence of ADMM with three or more blocks is always problematic. Following most existing ADMM algorithms, we prove the convergence of this method through an experiment. Figure 1 shows this method converges within 80 steps for all datasets.

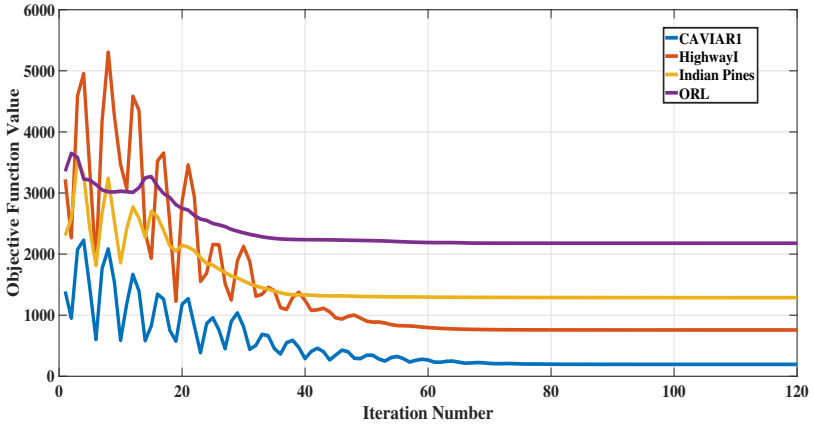


Figure 4: Convergence curve of our method on four datasets.

## 5. Conclusion

In this paper, we propose a novel method, Adaptive Robust Principal Component Analysis (ARPCA). Compared with most existing robust PCA methods, our model

simultaneously learns both clean data and similarity matrix that determines the construction of graph. It helps obtain both clean data with the lowest-rank structure and good graph that characterizes local intrinsic structure. Moreover, we adaptively construct graph on clean data rather than corruption data. This improves the flexibility of our model. Finally, the lowest-rank representation by our model enforces to correct corruption and reveal global geometric structure. Extensive experiments on several datasets illustrate that our model is superior to some related methods for clustering and low-rank recovery tasks.

**Acknowledgment**

The authors would like to thank the anonymous reviewers and AE for their constructive comments and suggestions, which improved the paper substantially. This work is supported by China Postdoctoral Science Foundation (Grant 2019M653564), National Natural Science Foundation of China under Grant 61773302, 61432014, 61772402 and the Fundamental Research Funds for the Central Universities.

**References**

[1] Q. Ke, T. Kanade, Robust l1-norm factorization in the presence of outliers and missing data by alternative convex programming, in: Computer Vision and Pattern Recognition, Vol. 1, 2005, pp. 739–746.

[2] Q. Gao, L. Ma, Y. Liu, X. Gao, F. Nie, Angle 2dpca: A new formulation for 2dpca, IEEE Transactions on Cybernetics.

[3] C. Ding, D. Zhou, X. He, H. Zha, R1-pca:rotational invariant l1-norm principal component analysis for robust subspace factorization, in: International Conference on Machine Learning, 2006, pp. 281–288.

[4] N. Xue, J. Deng, Y. Panagakis, S. Zafeiriou, Informed non-convex robust principal component analysis with features, in: Thirty-Second AAAI Conference on Artificial Intelligence, 2018.

[5] Y. Liu, S. Zhao, Q. Wang, Q. Gao, Learning more distinctive representation by enhanced pca network, Neurocomputing 275 (2018) 924–931.

[6] N. Kwak, Principal component analysis based on l1-norm maximization, IEEE Trans. Pattern Anal. Mach. Intell. 30 (9) (2008) 1672–1680.

[7] S. Liao, J. Li, Y. Liu, Q. Gao, X. Gao, Robust formulation for pca: Avoiding mean calculation with l 2, p-norm maximization, in: Thirty-Second AAAI Conference on Artificial Intelligence, 2018.

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

[9] N. Shahid, V. Kalofolias, X. Bresson, M. Bronstein, P. Vandergheynst, Robust principal component analysis on graphs, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 2812–2820.

[10] J. Wright, Y. Peng, Y. Ma, A. Ganesh, S. Rao, Robust principal component analysis: exact recovery of corrupted low-rank matrices by convex optimization, in: International Conference on Neural Information Processing Systems, 2009, pp. 2080–2088.

[11] F. Nie, G. Cai, X. Li, Multi-view clustering and semi-supervised classification with adaptive neighbours., in: AAAI, 2017, pp. 2408–2414.

[12] F. Nie, X. Wang, H. Huang, Clustering and projected clustering with adaptive neighbors, in: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2014, pp. 977–986.

[13] N. Vaswani, T. Bouwmans, S. Javed, P. Narayanamurthy, Robust subspace learning: Robust pca, robust subspace tracking, and robust subspace recovery, IEEE signal processing magazine 35 (4) (2018) 32–55.

[14] T. Bouwmans, A. Sobral, S. Javed, S. K. Jung, E.-H. Zahzah, Decomposition into low-rank plus additive matrices for background/foreground separation: A review

for a comparative evaluation with a large-scale dataset, Computer Science Review 23 (2017) 1–71.

[15] P. Narayanamurthy, N. Vaswani, A fast and memory-efficient algorithm for robust pca (merop), in: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2018, pp. 4684–4688.

[16] X. Jia, X. Feng, W. Wang, H. Huang, C. Xu, Online schatten quasi-norm minimization for robust principal component analysis, Information Sciences 476 (2019) 83–94.

[17] X. Guo, X. Cao, Speeding uplow rank matrix recovery for foreground separation in surveillance videos, in: 2014 IEEE International Conference on Multimedia and Expo (ICME), IEEE, 2014, pp. 1–6.

[18] P. Rodriguez, B. Wohlberg, Incremental principal component pursuit for video background modeling, Journal of Mathematical Imaging and Vision 55 (1) (2016) 1–18.

[19] S. Javed, S. Ho Oh, A. Sobral, T. Bouwmans, S. Ki Jung, Background subtraction via superpixel-based online matrix decomposition with structured foreground constraints, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2015, pp. 90–98.

[20] C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, S. Yan, Tensor robust principal component analysis: Exact recovery of corrupted low-rank tensors via convex optimization, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 5249–5257.

[21] T.-H. Oh, Y.-W. Tai, J.-C. Bazin, H. Kim, I. S. Kweon, Partial sum minimization of singular values in robust pca: Algorithm and applications, IEEE transactions on pattern analysis and machine intelligence 38 (4) (2016) 744–758.

[22] B. Jiang, C. Ding, J. Tang, Graph-laplacian pca: Closed-form solution and robustness, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 3492–3498.

[23] Z. Zhang, K. Zhao, Low-rank matrix approximation with manifold regularization, IEEE transactions on pattern analysis and machine intelligence 35 (7) (2013) 1717–1729.

[24] L. Tao, H. H. Ip, Y. Wang, X. Shu, Low rank approximation with sparse integration of multiple manifolds for data representation, Applied Intelligence 42 (3) (2015) 430–446.

[25] T. Jin, J. Yu, J. You, K. Zeng, C. Li, Z. Yu, Low-rank matrix factorization with multiple hypergraph regularizer, Pattern Recognition 48 (3) (2015) 1011–1022.

[26] G. Liu, Z. Lin, Y. Yu, Robust subspace segmentation by low-rank representation, in: Proceedings of the 27th international conference on machine learning (ICML-10), 2010, pp. 663–670.

[27] S. E. Ebadi, E. Izquierdo, Foreground segmentation via dynamic tree-structured sparse rpca, in: European Conference on Computer Vision, Springer, 2016, pp. 314–329.

[28] S. Javed, S. K. Jung, A. Mahmood, T. Bouwmans, Motion-aware graph regularized rpca for background modeling of complex scenes, in: 2016 23rd International Conference on Pattern Recognition (ICPR), IEEE, 2016, pp. 120–125.

[29] A. Sobral, T. Bouwmans, E.-h. ZahZah, Double-constrained rpca based on saliency maps for foreground detection in automated maritime surveillance, in: 2015 12th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), IEEE, 2015, pp. 1–6.

[30] X. Cheng, Y. Chen, Y. Tao, C. Wang, M. Kim, A. Lefcourt, A novel integrated pca and fld method on hyperspectral image feature extraction for cucumber chilling damage inspection, Transactions of the ASAE 47 (4) (2004) 1313.

[31] Y. Xu, Z. Wu, J. Chanussot, Z. Wei, Joint reconstruction and anomaly detection from compressive hyperspectral images using mahalanobis distance-regularized tensor rpca, IEEE Transactions on Geoscience and Remote Sensing 56 (5) (2018) 2919–2930.

[32] W. He, H. Zhang, H. Shen, L. Zhang, Hyperspectral image denoising using local low-rank matrix recovery and global spatial–spectral total variation, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 11 (3) (2018) 713–729.

[33] Keshavan, H. Raghunandan, Montanari, Andrea, Sewoong, Matrix completion from noisy entries, in: NIPS, 2009, pp. 469–476.

[34] X. He, D. Cai, S. Yan, H.-J. Zhang, Neighborhood preserving embedding, in: IEEE International Conference on Computer Vision, Vol. 2, IEEE, 2005, pp. 1208–1213.

[35] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, Foundations and Trends® in Machine Learning 3 (1) (2011) 1–122.

[36] J. F. Cai, E. J. Candes, Z. Shen, A singular value thresholding algorithm for matrix completion, Siam Journal on Optimization 20 (4) (2010) 1956–1982.

[37] L. Maddalena, A. Petrosino, Towards benchmarking scene background initialization, in: International Conference on Image Analysis and Processing, Springer, 2015, pp. 469–476.

[38] T. Bouwmans, L. Maddalena, A. Petrosino, Scene background initialization: A taxonomy, Pattern Recognition Letters 96 (2017) 3–11.

[39] S. Pai, S. Best, J. Roddick, H. Pahao, T. Baskerville, M. Brown, M. Harris, A. Cotterill, R. Thomas, Dataset, Aviris Nw Indiana's Indian Pines 9 (2) (2014) 1 – 4.

[40] B. Anton, J. Fein, T. To, X. Li, L. Silberstein, C. J. Evans, Immunohistochemical localization of orl-1 in the central nervous system of the rat, Journal of Comparative Neurology 368 (2) (1996) 229–251.

[41] N. Xue, Y. Panagakis, S. Zafeiriou, Side information in robust principal component analysis: Algorithms and applications, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 4317–4325.

[42] D. D. Bloisi, A. Pennisi, L. Iocchi, Parallel multi-modal background modeling, Pattern Recognition Letters 96 (2017) 45–54.

[43] B. Laugraud, S. Piérard, M. Van Droogenbroeck, Labgen: A method based on motion detection for generating the background of a scene, Pattern Recognition Letters 96 (2017) 12–21.

[44] M. De Gregorio, M. Giordano, Background estimation by weightless neural networks, Pattern Recognition Letters 96 (2017) 55–65.

[45] Z. Lin, M. Chen, Y. Ma, The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices, UIUC Technical Report.

[46] Y. Zhang, Recent advances in alternating direction methods: Practice and theory, in: IPAM Workshop on Continuous Optimization, 2010.