



Matrix completion and vector completion via robust subspace learning

Zhe Liu^a, Zhanxuan Hu^{a,b,*}, Feiping Nie^b

^aSchool of Natural and Applied Sciences, Northwestern Polytechnical University, Xian, Shaanxi 710072, PR China

^bSchool of computer science and Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xian, Shaanxi 710072, PR China

ARTICLE INFO

Article history:

Received 6 November 2017

Revised 22 February 2018

Accepted 10 April 2018

Available online 4 May 2018

Communicated by Liu Guangcan

Keywords:

Matrix completion

Vector completion

Out-of-sample

Subspace learning

Sparse noise

Outliers

ABSTRACT

Matrix completion is widely used in many practical applications such as computer vision and data mining. In this paper, we consider the following two issues arising in real scenarios. First, the collected datasets are damaged by sparse noise and column outliers simultaneously; second, the datasets are not static in nature due to the existing of out-of-sample. Both of them have been ignored by most existing methods.

In contrast with the traditional matrix completion algorithms which aim at recovering the entire matrix directly, we in this paper aim to first learn a low dimensional subspace by recovering a subset of collected samples, and then utilize it to estimate the missing values of residual data. There are two important advantages about this transformation. First, weakening the deviation caused by column outliers. Second, providing a direction for efficiently solving the out-of-sample problem. Particularly, to further improve the robustness of proposed method to sparse noise, we present a novel robust matrix completion model and a robust vector completion model, and both of them are based on non-convex ℓ_p -norm ($0 < p < 1$).

In experiments, the proposed method and other state-of-the-art algorithms will be used to cope with two problems: matrix completion and vector completion. Numerical results on real datasets and artificial datasets demonstrate that our method can provide a significant performance advantage over alternative methods.

© 2018 Elsevier B.V. All rights reserved.

1. Introduction

The basic idea of matrix completion is to find a complete low rank matrix that is consistent with original matrix on known entries, which has been studied widely in many fields such as recommendation systems [1,2], background subtraction [3,4], Photometric Stereo [5], and structure from motion [6].

Given a matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n] \in \mathbb{R}^{m \times n}$, where n is the number of samples and m is the dimensionality of sample. The matrix \mathbf{X} is stacked by n column vectors, hence the matrix completion can be naturally seen as an accumulation of n vector completion problems. As described in Fig. 1, given a complete vector $\mathbf{m} = (x, y, z)$, one can exactly determinate its location in original space. Nevertheless, the vector (point) \mathbf{m} is confirmed only up to the line l when the value of z is absent. In practice, however, if the low dimensional subspace \mathcal{S} where \mathbf{m} lies is known in advance,

one can confirm the value of z by intersecting line l and subspace \mathcal{S} [7]. Hence, the vector completion problem can be further translated as a subspace learning problem.

Principal Component Analysis [8] is a pioneer for subspace learning. Over the past decades, a number of algorithms have been proposed to improve its robustness and practicability. Such as ℓ_1 -PCA [9], which is developed for handling sparse noise, PCE [10], a method that can automatically estimate the feature dimension of subspace, LRR [11,12] and its variants [13], two representative methods for coping with the setting that samples are drawn from a union of multiple subspaces. Nevertheless, both of them are based on the assumption that the input data is complete. Recently, research work focus on the problem that learning a subspace from highly incomplete information. Generally, most existing methods can be divided into two categories: batch methods [14–20] and on-line methods [21–23]. Compared with the latter, the former has received more attention in past ten years due to its conciseness and theoretical guarantee.

Batch methods. The batch methods are generally referred to robust matrix completion. Given a matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$, the batch

* Corresponding author at: School of computer science and Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xian, Shaanxi 710072, PR China.

E-mail addresses: liuzhe@nwpu.edu.cn (Z. Liu), huzhanxuan@mail.nwpu.edu.cn (Z. Hu), feipingnie@gmail.com (F. Nie).

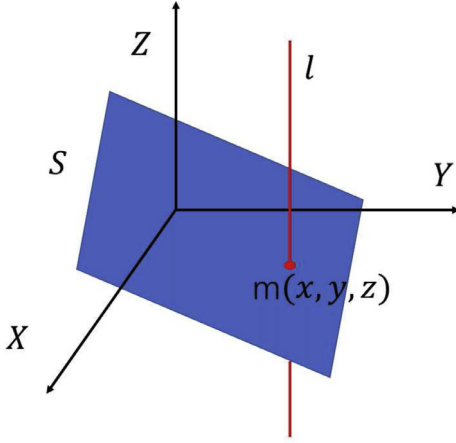


Fig. 1. A sample instance for vector completion. The point \mathbf{m} is confirmed only up to the line l when its coordinate on z -axis is absent. However, the locations of \mathbf{m} can be determined when the subspace S where \mathbf{m} lies is known.

algorithms aim at solving the following problem:

$$\min_L \text{rank}(\mathbf{L}) \quad \text{s.t.} \quad P_{\Omega}(\mathbf{L}) = P_{\Omega}(\mathbf{X}). \quad (1)$$

Since the problem (1) is NP-hard, many research works were devoted to solve its variants. Depending on the way of relaxing the rank minimization problem, all batch methods can be grouped into two categories: Bilinear Factorization (BF) [17,18,24] and Nuclear Norm Minimization (NNM) [25–28]. The most attractive advantage of NNM is that it can automatically determine the dimensionality of subspace. While the high time consuming [28] caused by SVD per iteration limits its application on large scale data. Comparing with the NNM, the BF based methods generally performs better in terms of computational efficiency and storage consumption. By utilizing the full rank decomposition when the rank r of desired matrix is known, most BF based methods reformulated the problem (1) as:

$$\min_{\mathbf{U}, \mathbf{V}} \|\mathbf{P}_{\Omega}(\mathbf{X} - \mathbf{UV})\|_{\ell_q}, \quad (2)$$

where $\mathbf{U} \in \mathbb{R}^{m \times r}$ and $\mathbf{V} \in \mathbb{R}^{r \times n}$ are two low rank matrices, ℓ_q denotes the norm used to measure loss function. It is obvious that the low dimensional subspace \mathcal{U} is fixed when matrix $\mathbf{L} = \mathbf{UV}$ is determined.

For problem (2), the ℓ_2 -norm based loss function is widely used due to its convexity and smoothness. Although ℓ_2 -norm based matrix completion algorithm returns a good estimation of the subspace when the known entries are damaged by Gaussian noise, it breaks down under sparse corruptions, even if that only a fraction of known entries are corrupted.

This undesirable issue has motivated the research of recovering a low rank matrix \mathbf{L} from a corrupted and incomplete matrix $P_{\Omega}(\mathbf{X}) = P_{\Omega}(\mathbf{L} + \mathbf{E})$, where \mathbf{E} represents a sparse matrix. To emphasize the sparseness of \mathbf{E} , many algorithms [29–32] utilize ℓ_1 -norm to build the loss function. Here ℓ_1 -norm is a convex relaxation of the ℓ_0 -quasi-norm. Unfortunately, the solution provided by ℓ_1 -norm may deviate from the original solution when the magnitude or number of corruptions is very large [33]. Recently, considering that some columns (samples) may be contaminated completely, Chen et al. [25] developed a robust method which emphasizes the sparseness of \mathbf{E} by $\ell_{2,1}$ -norm. However, its performance on real data is limited by the fragility to sparse corruptions. To reduce the ambiguity of notations, in the rest of this paper the entries corrupted by sparse noise are named as *corruptions*, and the columns (samples) contaminated completely by unknown noise are named as *outliers*. The [34] has considered the problem of recovering a low dimensional subspace from a matrix contaminated by

sparse corruptions and outliers simultaneously, but it can not deal with the setting that the matrix is incomplete.

In addition to the susceptibility to mixed noise, a significant drawback of existing batch methods is the low efficiency for dealing with the out-of-sample problem. Here, the out-of-sample denotes a fresh vector falling outside the training data. For instance, the new frame in video data. Actually, to estimate the missing values of a new sample the batch methods have to re-implement the algorithms on the entire new dataset, which is unacceptable for high dimensional data. The out-of-sample problem has been considered in [35], while it is developed for subspace clustering and can not be directly generalized to matrix completion.

Online methods. For the purpose of real-time in some practical applications, several online methods have been proposed. Compared with batch algorithms, the online methods have received little research attention, because it is susceptible to noise. Most online methods share the common program: first, parting the matrix \mathbf{X} as a series of column vectors $\{\mathbf{x}_i\}$; second, initializing a low rank matrix and then updating it by solving the following problem:

$$\min_{\mathbf{U}, \mathbf{v}_i} \|P_{\Omega}(\mathbf{x}_i - \mathbf{U}\mathbf{v}_i)\|_{\ell_q}. \quad (3)$$

where \mathbf{v}_i denotes the coordinate vector that projecting \mathbf{x}_i onto the current subspace. Finally, estimating the missing values of \mathbf{x}_i by $\mathbf{U}\mathbf{v}_i$, where the columns of \mathbf{U} are the basis of subspace. The main differences between existing methods are the way of updating \mathbf{U} and the norm used to measure the residual error, i.e., the value of q . Incremental Singular Value Decomposition (ISVD) [36] is a prior work for online learning subspace from incomplete information, which utilizes the SVD of $\mathbf{X}_n = \mathbf{U}_n \mathbf{S}_n \mathbf{V}_n^T$ to estimate the SVD of $\mathbf{X}_{n+1} = \mathbf{U}_{n+1} \mathbf{S}_{n+1} \mathbf{V}_{n+1}^T$, where $\mathbf{X}_{n+1} = [\mathbf{X}_n \quad \mathbf{x}_{n+1}]$. Then an online manifold learning algorithm GROUSE was proposed in [21], which conducts gradient descent on the Grassmannian manifold to update \mathbf{U} . The equivalency between ISVD and GROUSE has been proved in [37]. Subsequently, considering that the ℓ_2 -norm based objective functions are very sensitive to sparse corruptions, He et al. [22] substituted the ℓ_2 -norm by ℓ_1 -norm in GRATA. However, this method is still deficient due to the gap between ℓ_1 -norm and ℓ_0 -norm.

In addition to the approaches mentioned above, some online methods [23,38] have been developed, but most of them are vulnerable to outliers due to the way of updating subspace. As presented in (3), the online methods update the subspace by processing the data matrix, one column at a time, which makes them can estimate the missing values of one column vector (training data or out-of-sample) efficiently. However, a clear drawback is that the recovered subspace may deviate the intrinsic subspace significantly when some columns of training data are contaminated completely. Besides, although the existing online methods can be used to cope with the out-of-sample problem, none of them give a reasonable interpretation w.r.t this strategy. A natural problem is that why the coordinate vector \mathbf{v} learned from the known entries can be used to calculate the missing values. In Section 2, we will fill this blank via a sample formulation.

Paper innovations: As mentioned above, whether out-of-sample or matrix completion, each can be solved by vector completion under the condition that the intrinsic subspace is known. Hence, learning an optimal subspace from an incomplete matrix with simultaneous presence of sparse corruptions and outliers is the core of our method. To achieve this goal, we make the following innovations:

- First, we provide a formulation interpretation for Fig. 1 (a general case that \mathbf{x} contains multiple missing values), which has been omitted or ignored by many online methods. Significantly, the interpretation provides a guarantee for utilizing the subspace to solve the out-of-sample problem.

- Second, to learn the intrinsic subspace where **clean samples** lie we develop a robust batch method based on matrix Bilinear Factorization (BF). Particularly, we use the non-convex ℓ_p -norm rather than convex ℓ_1 -norm to built the objective function, which improves the robustness of algorithm to sparse corruptions. Note that the proposed non-convex optimization problem can be solved efficiently via an Inexact Augmented Lagrange Multiplier (IALM) method.
- Third, to remove the obstacle caused by outliers, we introduce a potential outliers tracking and removing strategy. More exactly, we learn the subspace by recovering only a subset of columns of original matrix. And, these columns have been determined as **inliers**. The experiment results present the motivation of this strategy and verify the advantage of it.
- Fourth, we propose a robust vector completion model which can utilize the subspace learned from **inliers** to estimate the missing values of out-of-sample accurately, even when the out-of-sample are contaminated by a great many of sparse corruptions.

Organization. The rest of the paper is organized as follows. We begin by describing the proposed method in Section 2. The optimization processes of proposed batch method and robust vector completion model are presented in Section 3. The performance evaluation and relevant experiment results are reported in Section 4. Finally, we conclude this paper with directions on potential improvements for future work in Section 5. The code of our method can be downloaded from in <https://github.com/sudalvxn/Matrix-completion.git>.

2. The proposed method

In this section, we begin, in Section 2.1, by summarizing the notations used throughout the paper. Then we provide an interpretation for vector completion in Section 2.2, which has been ignored by most online methods. A framework of our robust subspace learning method will be provided in Section 2.3. The robust batch based subspace learning method and robust vector completion model are described in Sections 2.4 and 2.5, respectively.

2.1. Notations

Matrices and vectors are bold capital and lowercase respectively. For example, $\mathbf{X} \in \mathbb{R}^{m \times n}$ denotes a matrix with n samples and each of them is m -dimension, and \mathbf{x}_i denotes the i th column (sample) of \mathbf{X} , x_j represents the j th element of \mathbf{x} , so the (i, j) th element of \mathbf{X} is denoted by $X_{i,j}$. For a vector \mathbf{x} , $\|\mathbf{x}\|_{\ell_p} = \|\mathbf{x}\|_p^p = (\sum_{i=1}^n |x_i|^p)^{1/p}$ denotes its ℓ_p -norm to power p ¹. Likewise, for matrix \mathbf{X} , we define its ℓ_p -norm to power p by $\|\mathbf{X}\|_{\ell_p} = \|\mathbf{X}\|_p^p = \sum_{i,j} |X_{i,j}|^p$. Particularly, $\|\mathbf{X}\|_2^2 = \|\mathbf{X}\|_F^2 = \langle \mathbf{X}, \mathbf{X} \rangle = \text{tr}(\mathbf{X}^T \mathbf{X})$, where $\|\mathbf{X}\|_F$ represents the Frobenius norm, and $\langle \cdot \rangle$ denotes inner product of matrix.

For a vector or matrix, the set of indices of known entries is denoted by Ω . Correspondingly, the P_Ω denotes a projection operator that projects a matrix \mathbf{X} to $P_\Omega(\mathbf{X})$, where $P_\Omega(\mathbf{X})$ is consistent with \mathbf{X} on Ω and equivalent to 0 on residual locations. U_Ω denotes a sub-matrix stacked by rows of \mathbf{U} whose indices corresponds to Ω . Besides, the cursive symbols \mathcal{U} represents a space spanned by the columns of \mathbf{U} .

2.2. An interpretation for vector completion

As mentioned in Section 1, we recast the matrix completion problem as a subspace learning system. By utilizing this

translation, actually, all existing batch methods can be generalized to cope with the out-of-sample problem. Nevertheless, this problem beyond the scope of this paper. Here, we present a formulation interpretation to demonstrate the reasonability of this translation, which is inspired by the work of [39] for PCA with missing values.

As presented in Fig. 1, for a vector \mathbf{x} with only one unknown entry, the missing value can be calculated by intersecting the line l and low dimensional subspace \mathcal{S} , where \mathcal{S} is determined. Consider a general case that the vector contains only k ($k \ll m$) known entries. Intersecting the vector \mathbf{x} and $\mathbf{U}\mathbf{v}$, where \mathbf{v} denotes the coordinate vector of \mathbf{x} on subspace \mathcal{U} , we have

$$\begin{bmatrix} \mathbf{x}_{\Omega^c} \\ \mathbf{x}_\Omega \end{bmatrix} = \begin{bmatrix} \mathbf{U}_{\Omega^c} \\ \mathbf{U}_\Omega \end{bmatrix} \mathbf{v}, \quad (4)$$

where $\mathbf{U} = [\mathbf{U}_{\Omega^c}; \mathbf{U}_\Omega]$ and $\mathbf{x} = [\mathbf{x}_{\Omega^c}; \mathbf{x}_\Omega]$, and Ω^c is the complement of Ω . Further, it can be rewritten as:

$$\begin{bmatrix} \mathbf{U}_{\Omega^c} & -\mathbf{I}_{m-k} \\ \mathbf{U}_\Omega & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \mathbf{x}_{\Omega^c} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{x}_\Omega \end{bmatrix}. \quad (5)$$

Obviously, A necessary condition for problem (5) to have a unique solution is that the structure of the matrix on the left-hand side being full column rank, i.e., $r + m - k \leq m$. This implies that $r \leq k$, i.e., we must be given at least r entries in order to estimate the missing values of a vector. Next, considering that the vector may be damaged by noise, we have the following theorem.

Theorem 1. Given a vector \mathbf{x} with k known entries and a low dimensional subspace \mathcal{U} where \mathbf{x} lies. Suppose the dimensionality of \mathcal{U} is r and $r \leq k$, the coordinate vector \mathbf{v} of \mathbf{x} in subspace \mathcal{U} can be calculated by $\mathbf{v} = (\mathbf{U}_\Omega^T \mathbf{U}_\Omega)^{-1} \mathbf{U}_\Omega^T \mathbf{x}_\Omega$. Besides, the missing values of \mathbf{x} can be calculated by $\mathbf{x}_{\Omega^c} = \mathbf{U}_{\Omega^c} \mathbf{v}$.

Proof. Suppose the projection of \mathbf{x} onto the subspace \mathcal{U} is $\mathbf{U}\mathbf{v}$, the optimal solution of \mathbf{v} and \mathbf{x}_{Ω^c} can be obtained by

$$\min_{\mathbf{v}, \mathbf{x}_{\Omega^c}} \|\mathbf{x} - \mathbf{U}\mathbf{v}\|_2^2, \quad (6)$$

further, it can be written as

$$\begin{aligned} & \min_{\mathbf{v}, \mathbf{x}_{\Omega^c}} \|\mathbf{x}_{\Omega^c} - \mathbf{x}_\Omega\|_2^2 + \|\mathbf{U}_{\Omega^c} \mathbf{v} - \mathbf{x}_\Omega\|_2^2 \\ & \Rightarrow \min_{\mathbf{v}, \mathbf{x}_{\Omega^c}} \|\mathbf{x}_{\Omega^c} - \mathbf{U}_{\Omega^c} \mathbf{v}\|_2^2 + \|\mathbf{x}_\Omega - \mathbf{U}_\Omega \mathbf{v}\|_2^2 \\ & \Rightarrow \min_{\mathbf{v}, \mathbf{x}_{\Omega^c}} L(\mathbf{v}, \mathbf{x}_{\Omega^c}) + L(\mathbf{v}) \end{aligned} \quad (7)$$

Taking the derivative w.r.t \mathbf{x}_{Ω^c} and setting it to zero, we have:

$$\mathbf{x}_{\Omega^c} = \mathbf{U}_{\Omega^c} \mathbf{v}. \quad (8)$$

Substituting it into Eq. (7), we can obtain:

$$\min_{\mathbf{v}} \|\mathbf{x}_\Omega - \mathbf{U}_\Omega \mathbf{v}\|_2^2. \quad (9)$$

Obviously, the closed-form solution of \mathbf{v} is:

$$\mathbf{v} = (\mathbf{U}_\Omega^T \mathbf{U}_\Omega)^{-1} \mathbf{U}_\Omega^T \mathbf{x}_\Omega, \quad (10)$$

and \mathbf{x}_{Ω^c} is given by

$$\mathbf{x}_{\Omega^c} = \mathbf{U}_{\Omega^c} \mathbf{v}. \quad (11)$$

In addition, since $\mathbf{U}_\Omega \in \mathbb{R}^{k \times r}$ and $\mathbf{U}_\Omega^T \mathbf{U}_\Omega \in \mathbb{R}^{r \times r}$, to ensure the matrix $\mathbf{U}_\Omega^T \mathbf{U}_\Omega$ is invertible, the number of known entries k must be larger than the dimensionality r of the low dimensional subspace \mathcal{U} . \square

Here we consider only the condition that \mathbf{x} is damaged by Gaussian noise. When it is also corrupted by sparse noise, i.e., $\mathbf{x} - \mathbf{e} = \mathbf{U}\mathbf{v}$, we need to estimate the sparse component \mathbf{e} at the same time.

¹ Note that when $0 < p < 1$, the $\|\mathbf{x}\|_{\ell_p}$ is a quasi-norm that violates the triangle inequality.

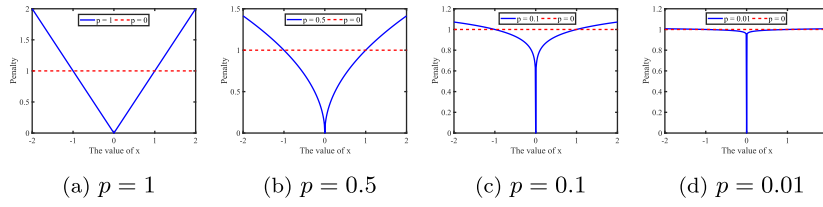


Fig. 2. Illustration of 1-dimensional penalty functions $y = |x|^p$ with different p .

2.3. The framework of proposed method

The Theorem 1 demonstrates that the missing values of a vector \mathbf{x} are computable when the subspace where \mathbf{x} lies is determined. Now, the question is how to learn a subspace from an incomplete matrix with corruptions and outliers.

Given an incomplete and contaminated matrix \mathbf{X} , we can utilize the robust matrix completion model (12) to weaken the impact caused by sparse corruptions. For outliers, a natural question is whether can we first determine the locations of them, and then remove them from input data? In this case, the low dimensional subspace can be recovered by utilizing the residual *inliers*. The answer is yes. In this paper, an outliers tracking strategy will be introduced into the robust subspace learning process. The framework of proposed method is outlined in Algorithm 1.

Algorithm 1 The framework of proposed robust subspace learning method.

Input: the data matrix \mathbf{X} and relevant information.

1. Tracking the potential outliers via proposed robust matrix completion model, and recording them into the matrix \mathbf{X}_{out} ;
2. Removing the potential outliers from \mathbf{X} and preserving the residual data as a new matrix \mathbf{X}_{new} ;
3. Learning the subspace \mathcal{U} where the *inliers* lie by recovering \mathbf{X}_{new} .
4. Estimating the missing values of \mathbf{X}_{out} via proposed robust vector completion model (13), where \mathbf{U} is provided by \mathbf{X}_{new} .

Output: the low rank matrix \mathbf{U} , i.e., the low dimensional subspace \mathcal{U} , the recovered matrix \mathbf{X}_{new}^* and matrix \mathbf{X}_{out}^* .

Since some columns of original matrix \mathbf{X} may be mistaken for potential outliers, in step 4 of Algorithm 1 we utilize the subspace \mathcal{U} learned by recovering matrix \mathbf{X}_{new} , to estimate the missing values of \mathbf{X}_{out} . The real outliers of \mathbf{X}_{out} fall outside the subspace \mathcal{U} , hence the final predicted results of them are in general inexact. Nevertheless, the main goal of us is to recover the clean data of original matrix.

Obviously, the robust matrix completion model is the core of proposed method as presented in Algorithm 1. Solving this problem efficiently and precisely is vital for subspace learning and subsequent missing values estimation. Next, we present our robust matrix completion model.

2.4. The proposed matrix completion model

Inspired by the works [40,41] for robust PCA, in this paper we utilize the ℓ_p -norm to measure the reconstruction error. Fig. 2 shows four examples of different penalties including three non-convex penalties corresponding to $p = 0.01$, $p = 0.1$, and $p = 0.5$, respectively, and one convex penalty corresponding to $p = 1$. Obviously, the ℓ_p -norm ($0 < p < 1$) offers better sparse approximation than ℓ_1 -norm, and when $p \rightarrow 0$, ℓ_p -norm $\rightarrow \ell_0$ -norm. It is obvious that the ℓ_p -norm can efficiently bridge the gap between ℓ_0 -norm and ℓ_1 -norm. Without specific illustration, in rest of this paper $0 < p < 1$. In our robust matrix completion model, the following

non-convex model will be solved:

$$\min_{\mathbf{U}^T \mathbf{U} = \mathbf{I}, \mathbf{V}, \mathbf{L} = \mathbf{UV}} \|\mathbf{P}_{\Omega}(\mathbf{X} - \mathbf{L})\|_{\ell_p}, \quad (12)$$

where $\mathbf{U}^T \mathbf{U} = \mathbf{I}$ is the orthogonal constraints w.r.t \mathbf{U} , which can shrink the solution space and provide a standard orthogonal basis for recovered subspace.

2.5. The proposed robust vector completion model

After recovering the intrinsic subspace \mathcal{U} from *inliers*, we can utilize the Eq. (13) to calculate the coordinate vector \mathbf{v} that projecting \mathbf{x} onto \mathcal{U} , and further estimate the missing values of \mathbf{x} by $\mathbf{U}\mathbf{v}$.

$$\min_{\mathbf{v}} \|\mathbf{P}_{\Omega^c}(\mathbf{x} - \mathbf{U}\mathbf{v})\|_{\ell_p}, \quad (13)$$

Here the vector \mathbf{x} represents the column of \mathbf{X}_{out} or the out-of-sample.

3. The optimization of proposed models

3.1. Optimize the matrix completion model via IALM

We in this paper utilize the IALM method to optimize proposed model, which has been used to solve many convex [27] and non-convex [30] optimization problems. The augmented Lagrange function of problem (12) is:

$$\mathcal{L}(\mathbf{L}, \mathbf{U}, \mathbf{V}, \mathbf{Y}, \mu) = \|\mathbf{P}_{\Omega}(\mathbf{X} - \mathbf{L})\|_p^p + \langle \mathbf{Y}, \mathbf{L} - \mathbf{UV} \rangle + \frac{\mu}{2} \|\mathbf{L} - \mathbf{UV}\|_F^2. \quad (14)$$

where \mathbf{Y} and μ are Lagrange multiplier and penalty parameter, respectively. Since the problem (14) is a non-convex optimization problem, we adopt the Alternating Direction Minimization (ADM) method to solve it. More exactly, we aim to solve the following subproblems in each iteration.

$$\mathbf{U}_{n+1} = \min_{\mathbf{U}^T \mathbf{U} = \mathbf{I}} \mathcal{L}(\mathbf{L}_n, \mathbf{U}, \mathbf{V}_n) \quad (15)$$

$$\mathbf{V}_{n+1} = \min_{\mathbf{V}} \mathcal{L}(\mathbf{L}_n, \mathbf{U}_{n+1}, \mathbf{V}) \quad (16)$$

$$\mathbf{L}_{n+1} = \min_{\mathbf{L}} \mathcal{L}(\mathbf{L}, \mathbf{U}_{n+1}, \mathbf{V}_{n+1}) \quad (17)$$

Obviously, these three variables are updated by minimizing each one while fixing the residuals.

3.1.1. Updating \mathbf{U}

Fixing the variables \mathbf{V} and \mathbf{L} , and removing the terms that do not involve the \mathbf{U} , we update \mathbf{U} by solving the following problem:

$$\min_{\mathbf{U}^T \mathbf{U} = \mathbf{I}} \|\mathbf{L}_n - \mathbf{UV}_n + \mathbf{Y}_n / \mu_n\|_F^2. \quad (18)$$

As presented in [30], this is a famous orthogonal procrustes problem, and the optimal solution of \mathbf{U} is given by conducting SVD for matrix $(\mathbf{L} - \mathbf{Y}/\mu)\mathbf{V}^T$. Particularly, we update \mathbf{U} by

$$\mathbf{U}_{n+1} = \mathbf{PQ}^T, \quad \mathbf{P}\Sigma\mathbf{Q}^T = \mathbf{M}. \quad (19)$$

where $\mathbf{M} = (\mathbf{L}_n + \mathbf{Y}_n/\mu_n)\mathbf{V}_n^T$, and $\mathbf{P}\Sigma\mathbf{Q}^T = \mathbf{M}$ is the SVD of it. Since \mathbf{M} is a $m \times r$ matrix and r is small in general, this step is efficient.

3.1.2. Updating \mathbf{V}

Similarly, fixing the variables \mathbf{U} and \mathbf{L} as the latest values. The \mathbf{V} is updated by solving

$$\min_{\mathbf{V}} \|\mathbf{L}_n - \mathbf{U}_{n+1}\mathbf{V} + \mathbf{Y}_n/\mu_n\|_F^2. \quad (20)$$

As matrix \mathbf{U}_{n+1} is orthogonal, we can obtain

$$\mathbf{V}_{n+1} = \mathbf{U}_{n+1}^T (\mathbf{L}_n + \mathbf{Y}_n/\mu_n). \quad (21)$$

3.1.3. Updating \mathbf{L}

Provided \mathbf{U}_{n+1} and \mathbf{V}_{n+1} , the variable \mathbf{L} can be updated by solving:

$$\min_{\mathbf{L}} \|\mathbf{P}_{\Omega}(\mathbf{X} - \mathbf{L})\|_p^p + \frac{\mu_n}{2} \|\mathbf{L} - \mathbf{U}_{n+1}\mathbf{V}_{n+1} + \mathbf{Y}_n/\mu_n\|_F^2. \quad (22)$$

Further, it can be rewritten as:

$$\begin{aligned} \min_{\mathbf{L}} \|\mathbf{P}_{\Omega}(\mathbf{X} - \mathbf{L})\|_p^p + \frac{\mu_n}{2} \|\mathbf{P}_{\Omega}(\mathbf{L} - \mathbf{U}_{n+1}\mathbf{V}_{n+1} + \mathbf{Y}_n/\mu_n)\|_F^2 \\ + \frac{\mu_n}{2} \|\mathbf{P}_{\Omega^c}(\mathbf{L} - \mathbf{U}_{n+1}\mathbf{V}_{n+1} + \mathbf{Y}_n/\mu_n)\|_F^2, \end{aligned} \quad (23)$$

where Ω^c denotes the complement of Ω , i.e., a set that records the indices of unknown entries. Since the elements of \mathbf{L} are separable, we can first get the optimal solution of $\mathbf{P}_{\Omega^c}(\mathbf{L})$:

$$\mathbf{P}_{\Omega^c}(\mathbf{L}) = \mathbf{P}_{\Omega^c}(\mathbf{U}_{n+1}\mathbf{V}_{n+1} - \mathbf{Y}_n/\mu_n). \quad (24)$$

And, for each element $\{L_{i,j} | (i,j) \in \Omega\}$, we can update it by solving the following problem.

$$\min_x \frac{1}{2}(x - a)^2 + \lambda|x|^p, \quad (25)$$

where $\lambda = 1/\mu_n$. Before coping with this non-convex problem, we introduce the Theorem 2 [42].

Theorem 2. Suppose $0 < p < 1$, and $c_1 = [2\lambda(1-p)]^{\frac{1}{2-p}}$ as well as $c_2 = c_1 + \lambda p c_1^{p-1}$ are two constants, then the solutions of problem(25) satisfy:

$$x^* = \tau(a) = \begin{cases} 0, & |a| < c_1 \\ \{0, \text{sgn}(a)c_1\}, & |a| = c_1 \\ \text{sgn}(a)x_*, & |a| > c_1 \end{cases}$$

where x^* is the solution of

$$x + \lambda p x^{p-1} = |a|, \quad (26)$$

which can be computed by the following iteration:

$$x_{k+1} = |a| - \lambda p x_k^{p-1} \quad (27)$$

with the initial condition $x_0 \in [c_1, |a|]$.

In practice the iteration (27) generally converges within two times [42]. According to Theorem 2, we update $\mathbf{P}_{\Omega}(\mathbf{L})$ by:

$$L_{i,j}^{n+1} = \tau(T_{i,j}), \quad (i,j) \in \Omega, \quad (28)$$

where $\mathbf{T} = \mathbf{U}_{n+1}\mathbf{V}_{n+1} - \mathbf{Y}_n/\mu_n$, and $L_{i,j}^{n+1}$ denotes the (i,j) th entry of \mathbf{L}_{n+1} .

The entire optimization process including the updating of \mathbf{Y} and μ is outlined in Step 1 of Algorithm 2.

3.1.4. Outliers tracking and removing

Considering the setting that the matrix \mathbf{X} is contaminated by corruptions and outliers simultaneously, we denote a matrix \mathbf{X}_{out} which is stacked by the potential corrupted columns of \mathbf{X} . Naturally, the \mathbf{X}_{out} violates the assumption that $\mathbf{X} = \mathbf{L} + \mathbf{S}$. A reasonable assumption is that the columns of \mathbf{X}_{out} , i.e., the outliers depart

Algorithm 2 The details of robust subspace learning.

Input: the matrix \mathbf{X} , rank r , parameter $\rho = 1.2$, and the value of K denoting the number of potential outliers.

Initialization: Initializing relevant variables.

while no converged **do**

1. // **Robust matrix completion**

- Update \mathbf{U} and \mathbf{V} by (19) and (21), respectively;

- Update \mathbf{L} by (24) as well as (28);

- Update \mathbf{Y} by $\mathbf{Y}_{n+1} = \mu_n(\mathbf{L} - \mathbf{U}\mathbf{V})$;

- Update μ by $\mu_{n+1} = \min(\rho_1\mu_n, 1e20)$;

- Convergence checking.

end while

2. // **Potential outliers tracking and removing**

Let $out = \max_{\kappa}(\mathbf{e})$, where $\max_{\kappa}(\mathbf{e})$ means extracting the indices corresponding to the κ largest elements of vector \mathbf{e} . Then, remove \mathbf{X}_{out} from \mathbf{X} and preserve the residual data as \mathbf{X}_{new} ;

3. // **Robust subspace learning**

Return Step 1 to learn the subspace \mathcal{U} by completing the matrix \mathbf{X}_{new} .

4. // **Estimate the missing values of \mathbf{X}_{out} .**

Complete each column of \mathbf{X}_{out} by Algorithm 3.

Output: the matrix \mathbf{U} , i.e., the subspace \mathcal{U} , and the recovered matrix \mathbf{L}^* .

even further from the recovered subspace compared with the *inliers*. Therefore, we provide a vector \mathbf{e} to measure the level of deviation of each column from recovered subspace. The i th element of \mathbf{e} is defined by:

$$e_i = \|\mathbf{P}_{\Omega^c}(\mathbf{I}_i - \mathbf{x}_i)\|_{\ell_p}, \quad (29)$$

In practice, the columns of \mathbf{X} corresponding to the K largest elements of \mathbf{e} will be considered as potential outliers and be removed after conducting Step 1. Therefore, the subspace will be learned by recovering the residual *clean data*. Note that a large value of K within a reasonable range will improve the probability of removing all potential outliers but mistake the *clean data* for *outliers*. Hence, we utilize the subspace learned from \mathbf{X}_{new} to estimate the missing values of \mathbf{X}_{out} via robust vector completion. The details w.r.t robust vector completion will be described in next section. The entire procedure of robust subspace learning is described in Algorithm 2.

3.2. Optimize the vector completion model via IALM

According to Theorem 1, we know that the missing values of a vector \mathbf{x} can be estimated when the subspace \mathcal{U} is known. The core is to estimate the coordinate vector \mathbf{v} . In practice, however, the vector \mathbf{x} may be corrupted by sparse noise, i.e., $\mathbf{x} - \mathbf{e} \in \mathcal{U}$ rather than $\mathbf{x} \in \mathcal{U}$, where \mathbf{e} denotes a sparse vector. Under this situation, we have to estimate the sparse vector \mathbf{e} and coordinate vector \mathbf{v} simultaneously. By defining $\mathbf{e} = \mathbf{x} - \mathbf{U}\mathbf{v}$, the problem (13) can be rewritten as:

$$\min_{\mathbf{v}, \mathbf{x} - \mathbf{U}\mathbf{v} = \mathbf{e}} \|\mathbf{P}_{\Omega^c}(\mathbf{e})\|_{\ell_p}. \quad (30)$$

Its Augmented Lagrange Function is:

$$\mathcal{L}(\mathbf{v}, \mathbf{e}, \mathbf{y}, \mu) = \|\mathbf{P}_{\Omega^c}(\mathbf{e})\|_p^p + \langle \mathbf{y}, \mathbf{x} - \mathbf{U}\mathbf{v} - \mathbf{e} \rangle + \frac{\mu}{2} \|\mathbf{x} - \mathbf{U}\mathbf{v} - \mathbf{e}\|_2^2. \quad (31)$$

Similarly, in $(k+1)$ th iteration we have:

$$\mathbf{v}_{k+1} = \mathbf{U}^T(\mathbf{x} - \mathbf{e}_k + \mathbf{y}_k/\mu_k), \quad (32)$$

and

$$\mathbf{e}_{k+1} = \begin{cases} \mathbf{e}_{i,j} = \tau(\mathbf{m}_{i,j}) & (i,j) \in \Omega\mathbf{x} \\ \mathbf{e}_{i,j} = \mathbf{m}_{i,j} & (i,j) \in \Omega\mathbf{x}^c \end{cases} \quad (33)$$

where $\mathbf{m} = \mathbf{x} - \mathbf{U}\mathbf{v}_{k+1} + \mathbf{y}_k/\mu_k$. The IALM approach for solving the robust vector completion problem is described in Algorithm 3.

Algorithm 3 Robust vector completion.

Input: the vector \mathbf{x} , matrix \mathbf{U} , parameter $\rho_2 = 1.8$.

Initialization: Initializing relevant variables.

while no converged **do**

1. // IALM for robust vector completion

- Update \mathbf{v} by (32);

- Update \mathbf{e} by (33);

- Update \mathbf{y} by $\mathbf{y}_{k+1} = \mu_k(\mathbf{x} - \mathbf{U}\mathbf{v} - \mathbf{e})$;

- Update μ by $\mu_{k+1} = \min(\rho_2\mu_k, 1e20)$;

- Convergence checking.

end while

2. Estimate the missing values of \mathbf{x} by $\mathbf{x}_{\Omega^c} = \mathbf{U}_{\Omega^c}\mathbf{v}$.

Output: the recovered vector \mathbf{x}^* .

Actually, a out-of-sample can be regarded as a vector that lies in the subspace \mathcal{U} . For instance, the background component of a new frame in video data is generally similar to the background recovered by the training data. Hence, the missing background pixels of the new frame can be estimated by utilizing the subspace \mathcal{U} learned from previous frames. In addition, in some practical applications such as recommender system, a user who changes its ratings on some items can also be considered as a out-of-sample. Naturally, one can provide a new estimation for it by conducting robust vector completion.

3.3. Complexity analysis

In this section, we investigate the time complexity of proposed methods, and the investigation consists of the following two fractions.

The time complexity of subspace learning: Before recovering the subspace, we have to estimate the locations of outliers via proposed batch based matrix completion model, i.e., the step 1 of Algorithm 1. In this process, the main time consuming is updating \mathbf{U} which needs conducting SVD for a matrix with size $m \times r$, and the time complexity is $\mathcal{O}(mr^2)$; in addition, the time complexity of updating \mathbf{V} is $\mathcal{O}(mnr)$ and updating \mathbf{L} is $\mathcal{O}(2|\Omega| + |\Omega^c|)$, where $|\Omega|$ is the number of known entries. As, we have to implement the step 1 two times, the total time complexity of subspace learning is $\mathcal{O}(2(mr^2 + mnr + 2|\Omega| + |\Omega^c|))$.

The time complexity of vector completion: Observing the Algorithm 2, we can find that it mainly involves some sample matrix multiplications, and the time complexity of updating \mathbf{v} is $\mathcal{O}(mr)$, updating \mathbf{e} is $\mathcal{O}(2|\Omega_x| + |\Omega_x^c|)$, where $|\Omega_x|$ is the number of known entries in \mathbf{x} . The total time complexity is $\mathcal{O}(mr + 2|\Omega_x| + |\Omega_x^c|)$.

4. Numerical experimental results

To verify the effectiveness of proposed robust subspace learning method, we implement numerous experiments on synthetic and real data. The information of used datasets and corresponding evaluation criterions are described in Section 4.1. The choices of parameters are discussed in Section 4.2. Besides, in this subsection, we report some experiment results to support our motivation w.r.t outliers determination. The comparisons between RMC-Lp and some alternative state-of-the-art robust algorithms on matrix completion problem will be reported in Section 4.3. And the experimental results on out-of-sample problem will be presented in Section 4.4.

Table 1

The information of real data.

| Dataset | Original frames | Used frames | Resolution |
|--------------|-----------------|-------------|------------|
| Hall-Monitor | 0–295 | 0–150 | 320 × 240 |
| HighwayI | 0–439 | 0–150 | 320 × 240 |
| HighwayII | 0–499 | 0–150 | 320 × 240 |
| IBMtest2 | 0–89 | 0–89 | 320 × 240 |

4.1. The information of used datasets

4.1.1. The details of synthetic data

The synthetic data matrix used in this paper for simulations follows one of the following models:

Model 1: The original noiseless matrix $\mathbf{L} \in \mathbb{R}^{m \times n}$. Each low rank matrix \mathbf{L} with rank r is generated as follows. Firstly, we generate two full rank matrices $\mathbf{U} \in \mathbb{R}^{m \times r}$ and $\mathbf{V} \in \mathbb{R}^{r \times n}$ with i.i.d. standard Gaussian entries, and then let $\mathbf{L} = \mathbf{UV}$.

Model 2: The matrix \mathbf{X}_s : the matrix \mathbf{X}_s represents a matrix corrupted only by sparse noise. According to [43], we select $\alpha\%mn$ elements from \mathbf{L} and corrupt them by the following non-Gaussian noise.

$$S = \kappa_{\pm 1}N(\mu, \sigma^2), \quad (34)$$

where $\kappa_{\pm 1} = 1$ or $\kappa_{\pm 1} - 1$ with equal probability, while $N(\mu, \sigma^2)$ denotes a random variable sampled from a Gaussian distribution.

Model 3: The matrix \mathbf{X} : The matrix corrupted by sparse noise and outliers simultaneously is denoted by \mathbf{X} . To compound matrix \mathbf{X} with c column outliers, we first generate a matrix $\mathbf{C} \in \mathbb{R}^{m \times c}$ with i.i.d. Gaussian-entries, where c denotes the number of outliers, and then utilize the columns of \mathbf{C} to replace c columns of \mathbf{X}_s randomly.

In each test, we extract $P = \rho \times mn$ entries from generated matrix as known entries, where ρ is the sample ratio. Since the ground-truth of test matrix is known, we utilize the Log-relative error (LRE) to evaluate the quality of recovered matrix \mathbf{X}^* . The LRE is defined by

$$LRE = \log_{10} \left(\frac{\|\mathbf{X}_I^* - \mathbf{L}_I\|_F}{\|\mathbf{L}_I\|_F} \right), \quad (35)$$

where I denotes a set recording the indices of inliers. The Eq. (35) means that we measure only the error w.r.t inliers, i.e., the columns damaged by corruptions only.

4.1.2. The information of real data

In [22], the authors have utilized their online robust matrix completion model to deal with the problem of background and foreground separation in video. The [4] present a detailed review w.r.t this problem. In this paper, four datasets selected from SBI² [44] will be used to evaluate the performance of proposed method. The information of these videos are described in Table 1. In addition, the evaluation criterions are described as follows:

– **AGE** (Average Gray-level Error): the average absolute error on gray image, which is defined as:

$$AGE = \frac{\|\mathbf{R} - \mathbf{G}\|_{\ell_1}}{mn}, \quad (36)$$

where mn is the total number of pixels of entire image, \mathbf{R} and \mathbf{G} represent the recovered background and ground-truth, respectively. Both of them are gray image.

– **PEP** (Percentage of Error Pixels): A pixel of recovered background whose value deviates from ground-truth more than $\tau = 20$ will be considered as a error pixel. $PEP = EP/N$, where EP is the total number of error pixels, and $N = mn$.

² Available at <http://sbmi2015.na.icar.cnr.it>.

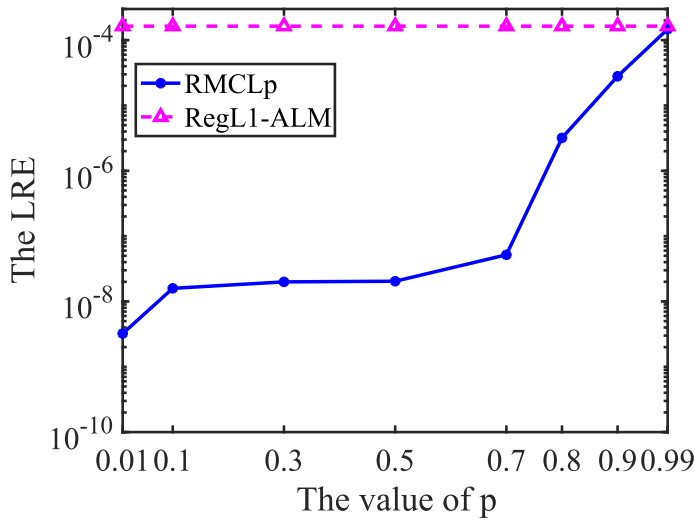


Fig. 3. The performance of our robust matrix model with different values of p . RegL1-ALM is an alternative method based on ℓ_1 -norm for comparison.

4.2. Parameter choice

Without considering the rank r , there are two important parameters in proposed method: the norm parameter p , and the number of potential outliers K . Now, we investigate the choice of them for the performance of proposed method.

The parameter p : As presented in Fig. 2, the gap between ℓ_p -norm and ℓ_0 -norm depends on the value of p . We conduct only the Step. 1 of Algorithm 2 to investigate the impact of p to our robust matrix model. In this experiment, the test matrix \mathbf{X}_s follows the Model 2 with $m = n = 500$, $r = 10$, and $\rho = 0.23$. In addition, for corruptions we set $\alpha = 10$, $\mu = 2$ and $\sigma = 10$. For RMC-Lp, We change the value of p in the range of $\{0.01, 0.1, 0.3, 0.5, 0.7, 0.8, 0.9, 0.99\}$. The RegL1-ALM method [30] will be conducted for comparison. The final results, the average of 20 times, are reported in Fig. 3 which demonstrates that the performance of our method decreases with the increasing of p , and this phenomenon is prominent when $p > 0.5$. In particular, the performance of our approach is very close to RegL1-ALM when $p = 0.99$, because the hard thresholding used in RegL1-ALM is the limit point of $\tau(z)$ as $p \rightarrow 1^-$. In practice, however, we fix $p = 0.1$ rather than $p = 0.01$ to avoid overfitting.

The motivation of outliers removing strategy: To verify the reasonability of Step 4 in Algorithm 2, in this section we implement our robust matrix completion model on synthetic datasets. The test matrix follows the Model 3. Except for parameter c , the residual parameters are selected as mentioned in Section 4.2. Four test cases were generated at the column outliers number c equal to 10, 50, 100, and 200, respectively. For visualization the final results, in each case we select the first c columns of \mathbf{X}_s , and replace them by generated outliers. Note that we also conduct only the Step.1 of Algorithm 2 in this test. The error of recovered vector \mathbf{x}_i^* is measured by

$$e_i = \|P_{\Omega_i}(\mathbf{x}_i - \mathbf{x}_i^*)\|_{\ell_p} \quad (37)$$

where \mathbf{x}_i denotes the i th column of input matrix \mathbf{X} . The error of each column with different number of outliers in training data are reported in Fig. 4. It is obvious that the outliers are clearly distinguishable when the value of c is small. In addition, the Fig. 4d shows that the recovered subspace will deviate the intrinsic subspace seriously when input data contains overmuch outliers.

The choice of parameter K : In practical applications, generally, the number of outliers is relatively few compared with the input data. In this case, it can be easily distinguished and removed from

Table 2

Numerical comparisons on synthetic data.

| Corruptions | Outliers | GRASTA | RegL1-ALM | RPCA-GD | RMC-Lp |
|---------------|---------------|----------|-----------|----------|-----------------|
| $\alpha = 10$ | $c = 10$ | 2.01e-06 | 4.90e-05 | 6.72e-02 | 2.13e-08 |
| | $c = 30$ | 6.83e-05 | 4.41e-05 | 7.02e-02 | 2.56e-08 |
| | $c = 50$ | 5.71e-04 | 1.72e-03 | 2.37e-01 | 3.32e-08 |
| | $c = 100$ | 3.29e-01 | 5.21e-02 | 3.54e-01 | 2.47e-08 |
| $c = 50$ | $\alpha = 5$ | 1.14e-05 | 1.84e-05 | 2.02e-03 | 3.59e-09 |
| | $\alpha = 10$ | 2.01e-04 | 1.93e-04 | 2.54e-02 | 2.13e-08 |
| | $\alpha = 15$ | 2.68e-03 | 3.54e-03 | 6.89e-02 | 3.32e-08 |
| | $\alpha = 20$ | 2.12e-02 | 8.24e-03 | 2.71e-01 | 2.76e-08 |

the original matrix. Hence, the outliers removing strategy introduced by Algorithm 2 is reasonable. After removing the potential outliers, the subspace where the clean data lies can be recovered exactly from the residual data. As the number of real outliers is unknown in real scenarios, we generally set $K = 0.2n$.

Besides, for Algorithms 2 and 3 we fix $\mu_1 = 1e - 6$. And the convergence condition of them follow $\|\mathbf{L} - \mathbf{U}\mathbf{V}\|_F \leq 1e - 8$ and $\|\mathbf{x} - \mathbf{U}\mathbf{v} - \mathbf{e}\|_2 \leq 1e - 8$, respectively. The parameter settings of all alternative methods are provided by original paper or authors suggestion.

4.3. Experiments on matrix completion

4.3.1. Matrix completion on synthetic data

In this section we aim to compare proposed method RMC-Lp to several state-of-the-art algorithms, including:

-GRASTA [22]³: A representative method for online subspace learning, which can be used to deal with the out-of-sample problem.

-RegL1-ALM [30]⁴: A method based on ℓ_1 -norm, which can be seen as an particular case of our method that $p \rightarrow 1$. In addition, the authors have added an nuclear norm regularization term w.r.t \mathbf{V} to accelerate the speed of convergence.

-RPCA-GD [45]⁵: A fast method for RPCA and robust matrix completion. The authors have introduced a corruption estimation operator to remove the potential sparse corruptions from observed entries.

The advantage of ℓ_p -norm: In this test, we conduct all methods on \mathbf{X}_s which follows the Model 2 with $m = n = 1000$, $r = 5$, $\rho = 0.20$. To verify the advantage of ℓ_p -norm for sparse corruptions with high magnitude, we change the value of σ . Particularly, in each case of σ the percentage of sparse corruptions will gradually increase. We present the final results in Fig. 5, which shows that the performance of GRASTA and RegL1-ALM, two ℓ_1 -norm based algorithms will descend rapidly with the increasing of sparse corruptions. Besides, in each case the final recovery result of RPCA-GD is inexact, because numerous sparse corruptions with high magnitude reduce the precision of low rank approximation, and further many corruptions will be mistaken as normal observed entries. The algorithm finally trapped into a vicious circle. Furthermore, it is obvious that the percentage and magnitude of sparse corruptions have no significant effect on our method.

The advantage of outliers removing strategy: Next, we compare our approach to alternative methods on Model 3. We vary the number of column outliers and the percentage of corruptions. For RMC-Lp, we also set $K = 0.2n$ although the number of real outliers is known. Table 2 shows that the number of outliers have an significant impact on GRASTA, and the performance of RegL1-ALM is poor when input data contains numerous sparse corruptions. In

³ Available at <https://sites.google.com/site/hejunzz/grasta>.

⁴ Available at <https://sites.google.com/site/yinqiangzheng/>.

⁵ Available at <https://www.yixinyang.org/code/RPCA-GD.zip>.

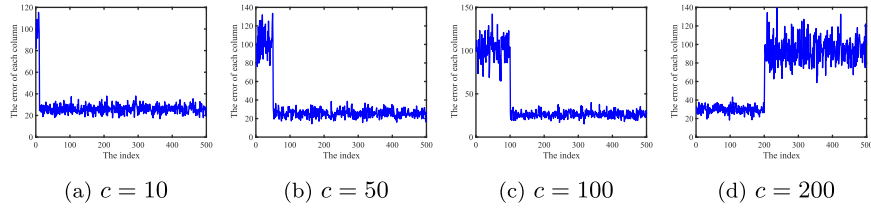


Fig. 4. The errors of all columns with different number of outliers in training data. In each case, the first c columns are **outliers**, and the residual are **inliers**. Obviously, the error between them are distinguishable. In Case 4, the recovered subspace deviates the intrinsic subspace seriously, because training data contains too many outliers.

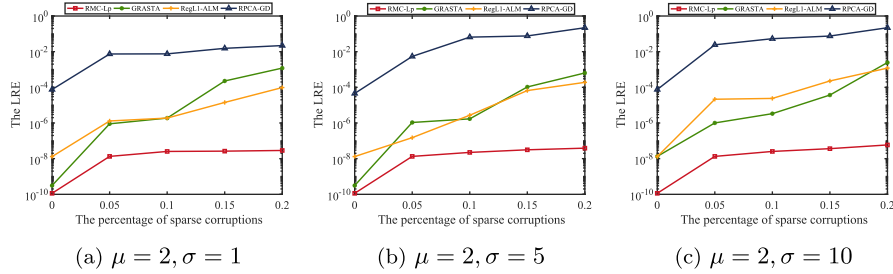


Fig. 5. The RE of all methods on synthetic datasets which follow the **Model 2**. We vary the value of σ to change the magnitude of corruptions. Obviously, the percentage and magnitude of sparse corruptions have no significant effect on our method.

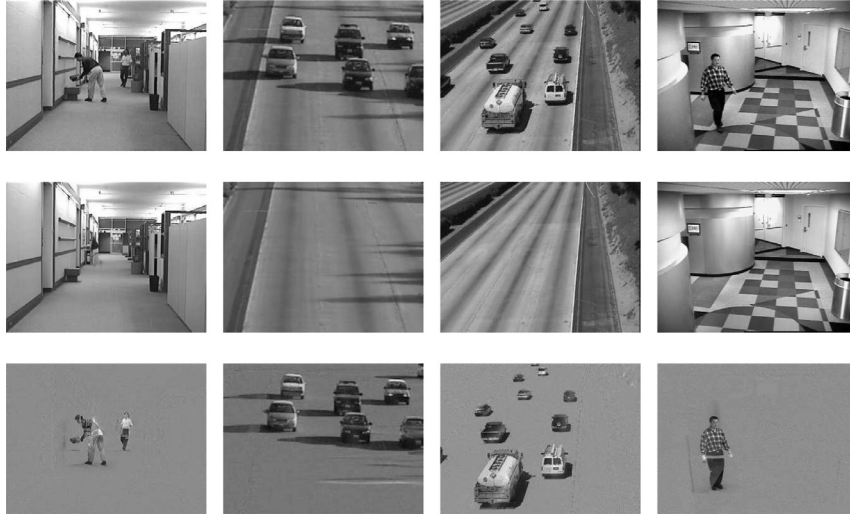


Fig. 6. Background subtraction and foreground separation results provided by RMC-Lp. The first row is the original frame of test video. The frame number are 100, 141, 16, and 117, respectively. The middle row represent the recovered background from 50% information. The last row is the foreground separation result. Visual difference with alternative methods is very slim, so only our results are presented.

contrast, the performance of our method almost maintains stable with the varying of outliers and corruptions. The reason is that the Step 2 of **Algorithm 2** helps us remove the columns that are corrupted completely. In this case, the subspace is recovered from inliers which only contaminated by sparse noise. The superiority of RMC-Lp for solving this issue has been proved in above.

4.3.2. Background subtraction by robust matrix completion

In this subsection, we apply RMC-Lp and alternative methods to subtract the background from a video, which is a basic problem in many computer vision fields. Following [22], we first capture the first L frames of a video, and recast them as a data matrix $\mathbf{X} \in \mathbb{R}^{mn \times L}$, where $mn = m \times n$ denotes the resolution of frame. Then, we extract 50% entries at random from \mathbf{X} as known entries. In each test, we set $r = 1$ for RMC-Lp as well as RegL1-ALM, while $r = 5$ for RPCA-GD as well as GRASTA. Besides, we can utilize the recovered image to separate the foreground object from original video. Due to the space limitation, we report only the final

Table 3

Numerical comparisons on background subtraction.

| Methods | Criteria | GRASTA | RegL1-ALM | RPCA-GD | RMC-Lp |
|----------------|----------|--------|-----------|---------|--------------|
| HallAndMonitor | AEG | 6.92 | 6.74 | 7.78 | 6.24 |
| | PEP | 0.045 | 0.042 | 0.054 | 0.036 |
| HighwayI | AEG | 5.65 | 5.55 | 6.21 | 5.15 |
| | PEP | 0.017 | 0.016 | 0.027 | 0.013 |
| HighwayII | AEG | 3.82 | 3.73 | 4.21 | 3.55 |
| | PEP | 0.016 | 0.014 | 0.021 | 0.011 |
| IBMtest2 | AEG | 3.76 | 3.51 | 4.07 | 3.21 |
| | PEP | 0.009 | 0.007 | 0.013 | 0.003 |

results of RMC-Lp in Fig. 6. The numerical comparisons of all methods are reported in Table 3. The Fig. 6 shows that RMC-Lp performs better in the condition that the background is static (HighwayI, HighwayII, IBMtest2). In HallAndMonitor data set, the partial information of **people** (Black T-shirt) imbed the back-

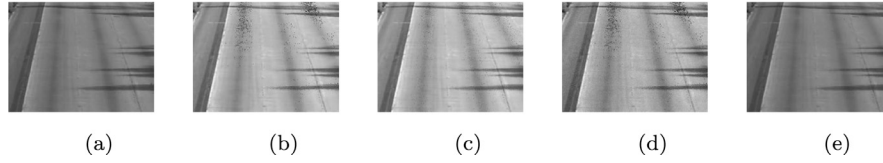


Fig. 7. Visual comparison for background subtraction on **Synthetic HighwayI** dataset with numerous outliers. Here we substitute 20 frames of original video by the images of residual datasets. (a). Ground truth; (b). RPCA-GD; (c). RegL1-ALM; (d). GRASTA; (e). RMC-Lp.

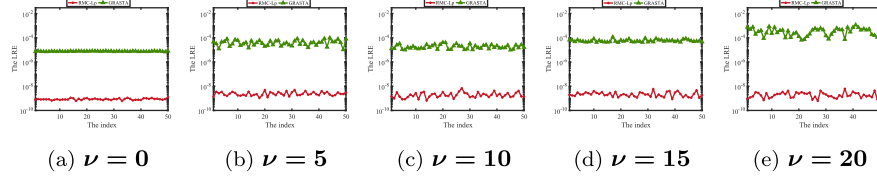


Fig. 8. Comparisons on out-of-sample with sparse noise. Where ν is the percentage of corruptions on known entries.

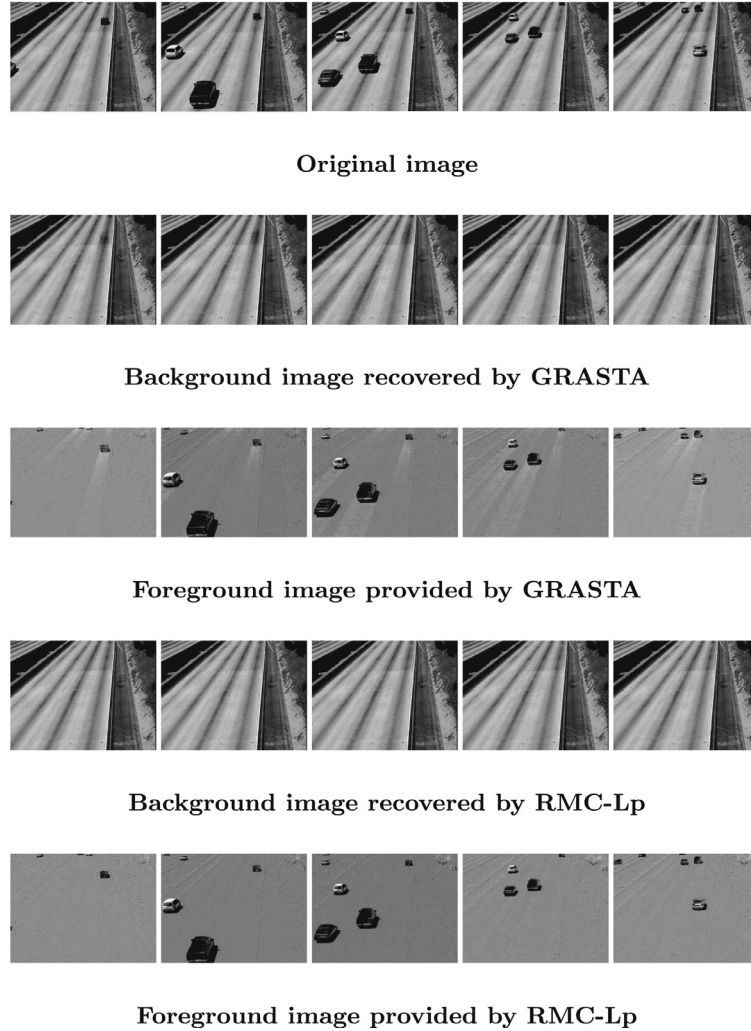


Fig. 9. Background subtraction and foreground separation results provided by GRASTA and RMC-Lp respectively. Here, the first 100 frames of HighwayII (including 20 outliers) serve as training data to learn the subspace, and the 103, 108, 114, 126, and 147 frames are selected as test data. The first row is the original frame of test video. The 2–3 rows represent the recovered results provided by GRASTA. The 4–5 rows represent the recovered results provided by RMC-Lp.

ground from frame 40 to frame 150, which leads to the error recovery of our method. In addition, we consider the situation that some frames of the test video are destroyed by foreground object completely. To generate such a dataset, we select 20 frames randomly from HighwayII, IBMtest, and HallAndMonitor as outliers,

and arbitrarily insert them into HighwayI. Then we implement all methods to process this synthetic video data. Fig. 7 clearly reveals that our method outperforms all alternative methods due to the advantage of outliers removing.

4.4. Experiments on out-of-sample

In this section we utilize the subspace learned from training data to cope with the out-of-sample, i.e., vector completion problem. All experiments are based on the Eq. (30). Note that we compare RMC-Lp to GRAFTA which is the sole online approach in alternative methods. Although many online methods [21,23,38] have been developed recently, most of them are based on ℓ_2 -norm and sensitive to corruptions and outliers.

4.4.1. Numerous results on synthetic data

Given a out-of-sample, a significant assumption is that it belongs to the subspace where the inliers of training data lie. Therefore, we simulate out-of-samples through the following way. We first generate a matrix \mathbf{L} which follows the **Model 1** with $m = 500$, $n = 550$, $r = 10$, $\rho = 0.20$, and then select 50 columns randomly from \mathbf{L} as out-of-samples. We denote the residual data matrix by \mathbf{L}_r . Subsequently, the matrix \mathbf{L}_r will be damaged by corruptions and outliers with $\alpha = 10$ ($\mu = 2$, $\sigma = 10$) and $c = 50$. In the meantime, $v\%$ known entries of each out-of-sample will be arbitrarily corrupted by sparse noise. A summary of test results is presented in Fig. 8, which demonstrates that RMC-Lp provides a significant advantage over GRAFTA in each case. There are two reasons relate to this result: first, the subspace recovered by RMC-Lp is closer to the intrinsic subspace of \mathbf{L} ; second, the robust vector completion model based on ℓ_p -norm further improve the robustness of algorithm to sparse corruptions.

4.4.2. Real-time background subtraction

To date, numerical researches have been devoted to the topic that subtracting the background of a new frame efficiently, and a large body of algorithms have been developed. However, most of them are based on RPCA and require the data being complete. GRAFTA is an representative method for robust subspace learning from incompleteness information with corruptions. In this section, we utilize it and RMC-Lp to recover the background of new video frames. Following to Section 4.4.1, we first extract 20 frames randomly from the first 100 frames of **Highway11** and substitute them by outliers selected from other video data. This synthetic video data will be used to learn the subspace. Then, we choose 5 frames at random from the residual **Highway11** data as out-of-sample. In theory, only one pixel of a new frame \mathbf{x} can help us to recover its background. In practice, however, we also sample 50% entries from \mathbf{x} as known entries to improve the robustness of algorithm to noise. Fig. 9 shows the recovered image. It is obvious that our method performs better than GRAFTA in this case.

5. Conclusion

This paper provides a novel approach for estimating the missing values of input data matrix. We translate the traditional matrix completion problem as a subspace learning system, and provide a simple formulation proof for its reasonability. To learn the optimal subspace from a matrix contaminated by sparse noise and column outliers simultaneously, we propose a robust matrix completion model and introduce a potential outliers removing strategy. The former can improve the robustness of algorithm to sparse noise, and provide a preliminary estimation for potential outliers. The final subspace learning procedure is conducted on the residual inliers.

Moreover, we present a robust vector completion method, which can be naturally used to estimate the missing values of out-of-sample. More importantly, our method provides a direction for generalizing the existing batch methods to cope with this issue.

It should be noted that we in this paper consider only the situation that the subspace of streaming data is constant or changes

slowly. Nevertheless, it may be inapplicable in some practical applications. For instance, the background of a parking lot generally changes over time due to the existing of illumination variation and moving objects. Actually, we can follow the online methods such as ISVD [36] and GRAFTA [22] to track the changes of subspace. The relevant investigation will be presented in our future work.

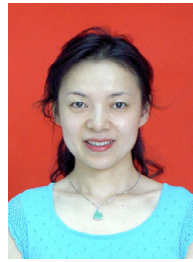
Acknowledgments

The research leading to these results has received funding from the **National Science Foundation of China** under Grant no. 61772427.

References

- [1] J.L. Herlocker, J.A. Konstan, A. Borchers, J. Riedl, An algorithmic framework for performing collaborative filtering, in: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, Berkeley, CA, USA, 1999, pp. 230–237.
- [2] Z. Zhang, K. Zhao, H. Zha, Inducible regularization for low-rank matrix factorizations for collaborative filtering, *Neurocomputing* 97 (2012) 52–62.
- [3] Q. Dong, L. Li, Smooth incomplete matrix factorization and its applications in image/video denoising, *Neurocomputing* 122 (2013) 458–469.
- [4] 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-scaledataset, *Comput. Sci. Rev.* 23 (2017) 1–71.
- [5] L. Wu, A. Ganesh, B. Shi, Y. Matsushita, Y. Wang, Y. Ma, Robust photometric stereo via low-rank matrix completion and recovery, in: Asian Conference on Computer Vision, Springer, 2010, pp. 703–717.
- [6] P.F.U. Gotardo, A.M. Martinez, Computing smooth time trajectories for camera and deformable shape in structure from motion with occlusion, *IEEE Trans. Pattern Anal. Mach. Intell.* 33 (10) (2011) 2051–2065.
- [7] R. Vidal, Y. Ma, S. Sastry, Generalized Principal Component Analysis, Springer, 2016.
- [8] S. Wold, K. Esbensen, P. Geladi, Principal component analysis, *Chemom. Intell. Lab. Syst.* 2 (1–3) (1987) 37–52.
- [9] N. Kwak, Principal component analysis based on ℓ_1 -norm maximization, *IEEE Trans. Pattern Anal. Mach. Intell.* 30 (9) (2008) 1672–1680.
- [10] X. Peng, J. Lu, Z. Yi, R. Yan, Automatic subspace learning via principal coefficients embedding, *IEEE Trans. Cybern.* 47 (11) (2017) 3583–3596.
- [11] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, Y. Ma, Robust recovery of subspace structures by low-rank representation, *IEEE Trans. Pattern Anal. Mach. Intell.* 35 (1) (2013) 171–184.
- [12] G. Liu, Q. Liu, P. Li, Blessing of dimensionality: recovering mixture data via dictionary pursuit, *IEEE Trans. Pattern Anal. Mach. Intell.* 39 (1) (2017) 47–60.
- [13] S. Li, Y. Fu, Learning robust and discriminative subspace with low-rank constraints, *IEEE Trans. Neural Netw. Learn. Syst.* 27 (11) (2016) 2160–2173.
- [14] J.F. Cai, Candes, E.J. S. Z. Shen, A singular value thresholding algorithm for matrix completion, *SIAM J. Optim.* 20 (4) (2010) 1956–1982.
- [15] F. Nie, H. Huang, C.H.Q. Ding, Low-rank matrix recovery via efficient Schatten p -norm minimization, in: Proceedings of the Twenty Sixth Conference on Artificial Intelligence, AAAI, 2012.
- [16] E. Kim, S. Oh, Robust orthogonal matrix factorization for efficient subspace learning, *Neurocomputing* 167 (2015) 218–229.
- [17] J. Tanner, K. Wei, Low rank matrix completion by alternating steepest descent methods, *Appl. Comput. Harmon. Anal.* 40 (2) (2016) 417–429.
- [18] Z. Wen, W. Yin, Y. Zhang, Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm, *Math. Progr. Comput.* 4 (4) (2012) 333–361.
- [19] G. Liu, Q. Liu, X. Yuan, A new theory for matrix completion, in: Proceedings of the Advances in Neural Information Processing Systems, 2017, pp. 785–794.
- [20] G. Liu, P. Li, Low-rank matrix completion in the presence of high coherence, *IEEE Trans. Signal Process.* 64 (21) (2016) 5623–5633.
- [21] L. Balzano, R. Nowak, B. Recht, Online identification and tracking of subspaces from highly incomplete information, in: Proceedings of the Forty Eighth Annual Allerton Conference on Communication, Control, and Computing, Allerton, 2010, pp. 704–711.
- [22] J. He, L. Balzano, A. Szlam, Incremental gradient on the Grassmannian for on-line foreground and background separation in subsampled video, in: Conference on Computer Vision and Pattern Recognition, IEEE, Providence, RI, USA, 2012, pp. 1568–1575.
- [23] C. Jin, S.M. Kakade, P. Netrapalli, Provable efficient online matrix completion via non-convex stochastic gradient descent, *Advances in Neural Information Processing Systems* (2016) 4520–4528.
- [24] R.H. Keshavan, S. Oh, A gradient descent algorithm on the grassman manifold for matrix completion, *CoRR* (2009) abs/0910.5260.
- [25] Y. Chen, H. Xu, C. Caramanis, S. Sanghavi, Matrix completion with column manipulation: near-optimal sample-robustness-rank tradeoffs, *IEEE Trans. Inf. Theory* 62 (1) (2015) 503–526.
- [26] S. Ma, D. Goldfarb, L. Chen, Fixed point and Bregman iterative methods for matrix rank minimization, *Math. Progr.* 128 (1–2) (2011) 321–353.

- [27] Z. Lin, M. Chen, Y. Ma, The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices, *CoRR* (2010) [abs/1009.5055](#).
- [28] Candes, E.J. S, X. Li, Y. Ma, J. Wright, Robust principal component analysis? *J. ACM* 58 (3) (2009) 1–73.
- [29] Q. Ke, T. Kanade, Robust ℓ_1 -norm factorization in the presence of outliers and missing data by alternative convex programming, in: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR*, v, IEEE, 2005, pp. 739–746.
- [30] Y. Zheng, G. Liu, S. Sugimoto, S. Yan, M. Okutomi, Practical low-rank matrix approximation under robust ℓ_1 -norm, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2012, pp. 1410–1417.
- [31] A. Eriksson, A. Van Den Hengel, Efficient computation of robust low-rank matrix approximations in the presence of missing data using the ℓ_1 -norm, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2010, pp. 771–778.
- [32] S. Chouvardas, M.A. Abdullah, L. Claude, M. Draief, Robust online matrix completion on graphs, in: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, (ICASSP)*, IEEE, 2017, pp. 4019–4023.
- [33] G. Gasso, A. Rakotomamonjy, S. Canu, Recovering sparse signals with a certain family of nonconvex penalties and DC programming, *IEEE Trans. Signal Process.* 57 (12) (2009) 4686–4698.
- [34] M. Rahmani, G. Atia, Low rank matrix recovery with simultaneous presence of outliers and sparse corruption (2017). *arXiv preprint arXiv:1702.01847*
- [35] X. Peng, H. Tang, L. Zhang, Z. Yi, S. Xiao, A unified framework for representation-based subspace clustering of out-of-sample and large-scale data, *IEEE Trans. Neural Netw. Learn. Syst.* 27 (12) (2016) 2499–2512.
- [36] M. Brand, Incremental singular value decomposition of uncertain data with missing values 2350 (2002) 707–720. *Lecture Notes in Computer Science*.
- [37] L. Balzano, S.J. Wright, On GROUSE and incremental SVD, in: *Proceedings of the IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, 2013, pp. 1–4.
- [38] D. Zhang, L. Balzano, Convergence of a Grassmannian gradient descent algorithm for subspace estimation from undersampled data(2016).
- [39] Y. Ma, S. Sastry, R. Vidal, Generalized principal component analysis, *Interdiscip. Appl. Math.* 43 (2016) 22–23.
- [40] F. Seidel, C. Hage, M. Kleinsteuber, pROST: a smoothed ℓ_p -norm robust online subspace tracking method for background subtraction in video, *Mach. Vis. Appl.* 25 (5) (2014) 1227–1240.
- [41] J. Wang, M. Wang, X. Hu, S. Yan, Visual data denoising with a unified Schatten-p norm and ℓ_q norm regularized principal component pursuit, *Pattern Recognit.* 48 (10) (2015) 3135–3144.
- [42] G. Marjanovic, V. Solo, On l_q optimization and matrix completion, *IEEE Trans. Signal Process.* 60 (11) (2012) 5714–5724.
- [43] L. Cambier, P.A. Absil, Robust low-rank matrix completion by riemannian optimization, *SIAM J. Sci. Comput.* 38 (5) (2016) S440–S460.
- [44] L. Maddalena, A. Petrosino, Towards Benchmarking Scene Background Initialization, *CoRR* (2015) 469–476 [abs/1506.04051](#).
- [45] X. Yi, D. Park, Y. Chen, C. Caramanis, Fast algorithms for robust PCA via gradient descent, *Advances in Neural Information Processing Systems* (2016) 4152–4160.



Zhe Liu received the Ph.D. degree in computer science from Northwestern Polytechnical University, China, in 2002. She is currently a full professor at Northwestern Polytechnical University, Xian, China. Her research interests are compressed sensing and dimensionality reduction.



Zhanxuan Hu is currently pursuing the Ph.D. degree in computer science with Northwestern Polytechnical University, Xian 710072, Shaanxi, PR China. His research interests include matrix completion, RPCA, rigid (Non-rigid) structure from motion and background subtraction.



Feiping Nie received the Ph.D. degree in computer science from Tsinghua University, China, in 2009. He is currently a full professor at Northwestern Polytechnical University, Xian, China. His research interests are machine learning and its application fields, such as pattern recognition, data mining, computer vision, image processing, and information retrieval. He has published more than 100 papers in the prestigious journals and conferences like TIP, TPAMI, TNNLS, TKDE, ICML, NIPS, KDD, etc. He is currently serving as an associate editor or a PC member for several prestigious journals and conferences in the related fields.