This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2020.3017672, IEEE Transactions on Pattern Analysis and Machine Intelligence

Enhanced Tensor RPCA and Its Application

Quanxue Gao, Pu Zhang, Wei Xia, Deyan Xie, Xinbo Gao, and Dacheng Tao, *Fellow,IEEE*

Abstract—Despite the promising results, tensor robust principal component analysis (TRPCA), which aims to recover underlying low-rank structure of clean tensor data corrupted with noise/outliers *by shrinking all singular values equally*, cannot well preserve the salient content of image. The major reason is that, in real applications, there is a salient difference information between all singular values of a tensor image, and the larger singular values are generally associated with some salient parts in the image. Thus, the singular values should be treated differently. Inspired by this observation, we investigate whether there is a better alternative solution when using tensor rank minimization. In this paper, we develop an enhanced TRPCA (ETRPCA) which explicitly considers the salient difference information between singular values of tensor data by the weighted tensor Schatten p-norm minimization, and then propose an efficient algorithm, which has a good convergence, to solve ETRPCA. Extensive experimental results reveal that the proposed method ETRPCA is superior to several state-of-the-art variant RPCA methods in terms of performance.

Index Terms—Tensor singular value decomposition, Robust principal component analysis, Multi-dimensional image denoising.

1 Introduction

Low-rank representation has been widely used for background/foreground detection in video surveillance and image denoising, where Robust Principal Component Analysis (RPCA) [1] is one of the most representative methods. It aims to separate the data matrix $\mathbf{Y}$ into a low-rank matrix $\mathbf{X} \in \mathbb{R}^{m\times n}$ and a sparse matrix $\mathbf{E}$, where $\mathbf{X}$ represents clean data, and $\mathbf{E}$ is error. Since the rank minimization is non-convex and discontinuous, researchers usually use nuclear norm minimization as a convex relaxation in real applications [2], [3].

Inspired by the impressive results of RPCA, variant RPCA methods have been developed by leveraging different regularized terms such as manifold regularization [4], [5] and noise regularized term [6], and achieved good experiments in some applications. Although their motivations are different, all of them regularize all singular values equally when solving nuclear norm minimization. This reduces the flexibility of algorithms and obtains the clean data whose rank does not approximate the target rank in practice. To solve these problems, two of the most representative nuclear norm minimization methods are: (1) Adaptive nuclear norm penalization [7], which is the same as weighted nuclear norm Minimization [8]. Using it instead of standard nuclear norm in RPCA, weighted RPCA (WRPCA) is described as

$$\min_{\mathbf{X},\mathbf{E}} ||\mathbf{X}||_{\boldsymbol{\omega},*} + \lambda||\mathbf{E}||_1 \qquad s.t.\ \mathbf{Y} = \mathbf{X} + \mathbf{E} \tag{1}$$

where $||\mathbf{X}||_{\boldsymbol{\omega},*} = \sum_j \omega_j * \sigma_j(\mathbf{X})$, ω_j denotes the j-th element of weighted vector $\boldsymbol{\omega}$, $\sigma_j(\bullet)$ denotes the j-th largest singular value of a matrix. (2) Nuclear norm minimization with Partial Sum of Singular Values (PSSV) [9], which is similar to truncated nuclear norm regularization [10]. Using it instead of standard nuclear norm, Oh *et al.* [9] proposed an alternative objective function which is defined as

$$\min_{\mathbf{X}} ||\mathbf{X}||_{p=r} + \lambda||\mathbf{E}||_1 \qquad s.t.\ \mathbf{Y} = \mathbf{X} + \mathbf{E} \tag{2}$$

where $||\mathbf{X}||_{p=r} = \sum_{j=r+1}^{\min(m,n)} \sigma_j(\mathbf{X})$.

One major shortcoming of the aforementioned nuclear norm minimization methods is that they cannot well handle the multi-dimensional data, also referred to tensor data which are ubiquitous in real applications [11], [12]. For example, a color image is a 3-way object with column, row and color modes; a gray scale video is indexed by two spatial variables and one temporal variable. Since existing nuclear norm methods first need to transform each multi-dimensional data into a matrix when learning the low-rank clean data, they ignore the information embedded in multi-dimensional structure. This would cause performance degradation. To alleviate this issue, inspired by tensor Singular Value Decomposition (t-SVD), Lu *et al.* [13] extended RPCA to tensor case and proposed Tensor Robust Principal Component (TRPCA) which achieves impressive results in the experiments. Motivated by TRPCA, Zhang *et al.* [14] developed tensor PSSV (TPSSV) that well exploits the target rank when learning the clean tensor data.

Although TRPCA and TPSSV have achieved impressive results for multi-dimensional data analysis, they still have some limitations. First, TRPCA explicitly considers each singular value equally and shrinks all singular values with the same parameter in solving the tensor nuclear norm minimization. In real applications, singular values have clear physical meanings, and the larger singular values are generally associated with some prominent information of the image. Thus, to preserve the prominent information embedding in image, we should make the large singular values shrink less, which was unfortunately not taken into account in TRPCA. Thus, it cannot well preserve some prominent information such as color information. Second, TPSSV does not

*Manuscript received Aug. 8, 2019; revised ****, ****.*

- *This work is supported by National Natural Science Foundation of China (Grant 61773302), Natural Science Basic Research Plan in Shaanxi Province (Grant 2020JZ-19). Corresponding author: Q. Gao, Email: qxgao@xidian.edu.cn.*

- *Q. Gao is with the Unmanned system research institute, Northwestern Polytechnical University, 710069, Xi'an China.*
- *Q. Gao, W. Xia, P. Zhang, D. Xie, and X. Gao are with Integrated services Networks, Xidian University, 710071, Xian China.*
- *D. Tao is with the UBTECH Sydney Artificial Intelligence Centre and the School of Information Technologies, Faculty of Engineering and Information Technologies, University of Sydney, Darlington, NSW 2008, Australia*

0162-8828 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Cornell University Library. Downloaded on August 22,2020 at 18:51:09 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2020.3017672, IEEE Transactions on Pattern Analysis and Machine Intelligence

shrink the first r largest singular values. Doing so implies that information, which are associated with the first r largest singular values, does not include nothing to do with content of the image. However, such an assumption restricts its capability and flexibility in dealing with many practical problems [15].

To handle the aforementioned problems and well exploit multi-dimensional structure embedded in tensor data, inspired by t-SVD based nuclear norm and TRPCA, we present an enhanced TRPCA (ETRPCA) that explicitly considers the prior knowledge of singular values of tensor data in solving tensor nuclear norm minimization, and then propose an efficient algorithm to solve the tensor low-rank matrix. Finally, we apply it to multi-dimensional data for image recovery/denosing and background modeling. The main contributions of our work are summarized as follows:

- We study the weighted tensor Schatten p-norm minimization (WTSNM) based on t-SVD and propose an efficient algorithm, which has a closed-form solution, to solve WTSNM. WTSNM explicitly considers the prior knowledge of singular values of tensor data.
- We apply WTSNM to solve our proposed ETRPCA and develop an efficient algorithm which has a good convergence. The obtained clean tensor data well preserve both the multi-dimensional structure and prominent information embedded in tensor data.
- Most existing nuclear norm minimization models can be viewed as special cases of our model.

2 Notations and Preliminaries

For convenience, we first introduce the notations and definitions used throughout the paper. We use bold calligraphy letters for third-order tensors, e.g., $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, bold upper case letters for matrices, e.g., $\mathbf{A}$, bold lower case letters for vectors, e.g., $\mathbf{a}$, and lower case letters such as a_{ijk} for the entries of $\boldsymbol{\mathcal{A}}$. Moreover, we denote $\mathbf{A}^{(i)}$ by the i-th frontal slice of $\boldsymbol{\mathcal{A}}$ and $\overline{\boldsymbol{\mathcal{A}}}$ the discrete Fast Fourier Transform (FFT) of $\boldsymbol{\mathcal{A}}$ along the third dimension, i.e., $\overline{\boldsymbol{\mathcal{A}}} = fft(\boldsymbol{\mathcal{A}}, [], 3)$. Thus, $\boldsymbol{\mathcal{A}} = ifft(\overline{\boldsymbol{\mathcal{A}}}, [], 3)$.

Definition 1. [16] For a 3-way tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, the Frobenius norm of $\boldsymbol{\mathcal{A}}$ is $||\boldsymbol{\mathcal{A}}||_F = \sqrt{\sum_{ijk} |a_{ijk}|^2}$, and conjugate transpose of $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is $\boldsymbol{\mathcal{A}}^T \in \mathbb{R}^{n_2 \times n_1 \times n_3}$.

Definition 2. [16] For a 3-way tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, we have

$$bdiag(\overline{\boldsymbol{\mathcal{A}}}) = \begin{bmatrix} \overline{\mathbf{A}}^{(1)} & & & \\ & \overline{\mathbf{A}}^{(2)} & & \\ & & \ddots & \\ & & & \overline{\mathbf{A}}^{(n_3)} \end{bmatrix} \tag{3}$$

Definition 3. [16] For a 3-way tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, its block circulant matrix is a matrix of $n_1 n_3 \times n_2 n_3$ having the following form:

$$bcirc(\boldsymbol{\mathcal{A}}) = \begin{bmatrix} \mathbf{A}^{(1)} & \mathbf{A}^{(n_3)} & \dots & \mathbf{A}^{(2)} \\ \mathbf{A}^{(2)} & \mathbf{A}^{(1)} & \dots & \mathbf{A}^{(3)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}^{(n_3)} & \mathbf{A}^{(n_3-1)} & \dots & \mathbf{A}^{(1)} \end{bmatrix} \tag{4}$$

Definition 4. [16] For a tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, we have

$$\begin{aligned} unfold(\boldsymbol{\mathcal{A}}) &= \left[\mathbf{A}^{(1)}; \mathbf{A}^{(2)}; \cdots; \mathbf{A}^{(n_3)}\right] \\ fold(unfold(\boldsymbol{\mathcal{A}})) &= \boldsymbol{\mathcal{A}} \end{aligned} \tag{5}$$

Definition 5. [16] Let $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ and $\boldsymbol{\mathcal{B}} \in \mathbb{R}^{n_2 \times l \times n_3}$, then the t-product between them is $\boldsymbol{\mathcal{C}} \in \mathbb{R}^{n_1 \times l \times n_3}$, i.e.,

$$\boldsymbol{\mathcal{C}} = \boldsymbol{\mathcal{A}} * \boldsymbol{\mathcal{B}} = fold(bcirc(\boldsymbol{\mathcal{A}}) \cdot unfold(\boldsymbol{\mathcal{B}})) \tag{6}$$

t-product between $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$ can be computed efficiently by

1) Calculate $\overline{\boldsymbol{\mathcal{A}}} = fft(\boldsymbol{\mathcal{A}}, [], 3)$ and $\overline{\boldsymbol{\mathcal{B}}} = fft(\boldsymbol{\mathcal{B}}, [], 3)$;
2) Multiply the each pair of the frontal slices of $\overline{\boldsymbol{\mathcal{A}}}$ and $\overline{\boldsymbol{\mathcal{B}}}$ to obtain $\overline{\boldsymbol{\mathcal{C}}}$;
3) Calculate $\boldsymbol{\mathcal{C}} = ifft(\overline{\boldsymbol{\mathcal{C}}}, [], 3)$;

Theorem 1. [16]. Block-circulant matrix can be block-diagonalized by

$$(\mathbf{F_{n_3}} \otimes \mathbf{I_{n_1}}) \cdot bcirc(\boldsymbol{\mathcal{A}}) \cdot (\mathbf{F_{n_3}}^{-1} \otimes \mathbf{I_{n_2}}) = \overline{\mathbf{A}} \tag{7}$$

where $\otimes$ denotes the Kronecker product, $\mathbf{F}_{n_3}$ is the $n_3 \times n_3$ Discrete Fourier Transform (DFT) matrix, $\mathbf{I}_{n_1}$ and $\mathbf{I}_{n_2}$ denote $n_1 \times n_1$ and $n_2 \times n_2$ identity matrices, respectively.

Theorem 2. [16](T-SVD) Let $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, then $\boldsymbol{\mathcal{A}}$ can be factored as

$$\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{U}} * \boldsymbol{\mathcal{S}} * \boldsymbol{\mathcal{V}}^T \tag{8}$$

where $\boldsymbol{\mathcal{U}} \in \mathbb{R}^{n_1 \times n_1 \times n_3}$ and $\boldsymbol{\mathcal{V}} \in \mathbb{R}^{n_2 \times n_2 \times n_3}$ are orthogonal, $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is a f-diagonal tensor.

Definition 6. [13], [17] (tensor nuclear norm). Given $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, $l = min(n_1, n_2)$, its nuclear norm is

$$||\boldsymbol{\mathcal{X}}||_{\circledast} = \sum_{i=1}^{n3} ||\overline{\mathbf{X}}^{(i)}||_* = \sum_{i=1}^{n3} \sum_{j=1}^{l} \sigma_j(\overline{\mathbf{X}}^{(i)}) \tag{9}$$

3 Robust TRPCA

3.1 Motivation and Objective

TRPCA [13] is a representative tensor low-rank representation method. It learns the clean low-rank tensor data $\boldsymbol{\mathcal{X}}$ from $\boldsymbol{\mathcal{Y}}$ by

$$\min_{\boldsymbol{\mathcal{E}}, \boldsymbol{\mathcal{X}}} \lambda ||\boldsymbol{\mathcal{E}}||_1 + ||\boldsymbol{\mathcal{X}}||_{\circledast} \quad s.t.\ \boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{X}} + \boldsymbol{\mathcal{E}} \tag{10}$$

According to Definition 6, we have that the model (10) regularizes all singular values of tensor data equally and shrinks all singular values with the same parameter in solving the tensor nuclear norm minimization. In real applications, for an arbitrary image, there is a large difference between its non-zero singular values, especially between the first several large singular values and the last several small singular values, and the larger singular values are generally associated with some prominent information such as color information of the image. To well exploit the prominent information embedding in image, we should make the large singular values shrink less, while TRPCA does not fully leverage this prior information about singular values in minimizing tensor nuclear norm. Thus, it cannot well preserve some prominent information such as color information. Moreover, the rank of the clean tensor data $\boldsymbol{\mathcal{X}}$ may not approximate the target rank in practice. To tackle this problem, one popular model is TPSSV whose objective function is [14]

$$\min_{\boldsymbol{\mathcal{E}}, \boldsymbol{\mathcal{X}}} \lambda ||\boldsymbol{\mathcal{E}}||_1 + ||\boldsymbol{\mathcal{X}}||_{p=r} \quad s.t.\ \boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{X}} + \boldsymbol{\mathcal{E}} \tag{11}$$

where $||\boldsymbol{\mathcal{X}}||_{p=r} = \sum_{i=1}^{n3} ||\overline{\mathbf{X}}^{(i)}||_{p=r}$. From the model (11), we have that it does not shrink the first r largest singular values in solving the tensor nuclear norm minimization. This indicates that the information, which are associated with the first r largest singular values, do not include nothing to do with content of the

0162-8828 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Cornell University Library. Downloaded on August 22,2020 at 18:51:09 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2020.3017672, IEEE Transactions on Pattern Analysis and Machine Intelligence

image. However, such an assumption is very strict and may not be reasonable in practice [15].

Inspired by the aforementioned insight analysis, to preserve the prominent information, we should make the large singular values shrink less in tensor nuclear norm minimization. Thus, we first introduce weighted tensor Schatten p-norm as

Definition 7. Given $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, $h = \min(n_1, n_2)$, weighted tensor Schatten p-norm of $\boldsymbol{\mathcal{X}}$ is defined as

$$\begin{aligned} ||\boldsymbol{\mathcal{X}}||_{\boldsymbol{\omega},S_p} &= (\sum\nolimits_{i=1}^{n_3} ||\overline{\mathbf{X}}^{(i)}||_{\boldsymbol{\omega},S_p}^p)^{\frac{1}{p}} \\ &= (\sum\nolimits_{i=1}^{n_3} \sum\nolimits_{j=1}^{h} \omega_j * \sigma_j(\overline{\mathbf{X}}^{(i)})^p)^{\frac{1}{p}} \end{aligned} \tag{12}$$

Then, we propose a Robust TRPCA whose objective function is

$$\min_{\boldsymbol{\mathcal{E}},\boldsymbol{\mathcal{X}}} \lambda||\boldsymbol{\mathcal{E}}||_1 + ||\boldsymbol{\mathcal{X}}||_{\boldsymbol{\omega},S_p}^p \quad s.t.\ \boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{X}} + \boldsymbol{\mathcal{E}} \tag{13}$$

3.2 Optimization

Inspired by Augmented Lagrange Multipliers, the model (13) can be solved by minimizing the model (14).

$$\begin{aligned} &\Gamma(\boldsymbol{\mathcal{E}}, \boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{L}}, \mu) = \lambda||\boldsymbol{\mathcal{E}}||_1 + \langle \boldsymbol{\mathcal{L}}, \boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{E}} \rangle \\ &||\boldsymbol{\mathcal{X}}||_{\boldsymbol{\omega},S_p}^p + \tfrac{\mu}{2}||\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{E}}||_F^2 \end{aligned} \tag{14}$$

where, $\boldsymbol{\mathcal{L}}$ is the Lagrange Multiplier and μ is a positive scaler. The main procedure includes the following several subproblems:

$\boldsymbol{\mathcal{E}}$-subproblem: To update $\boldsymbol{\mathcal{E}}$ (fixing the other variables), the model (14) becomes

$$\arg\min_{\boldsymbol{\mathcal{E}}} \tfrac{\lambda}{\mu_k}||\boldsymbol{\mathcal{E}}||_1 + \tfrac{1}{2}||\boldsymbol{\mathcal{E}} - \boldsymbol{\mathcal{H}}_k||_F^2 \tag{15}$$

where $\boldsymbol{\mathcal{H}}_k = \boldsymbol{\mathcal{Y}} + \mu_k^{-1}\boldsymbol{\mathcal{L}}_k - \boldsymbol{\mathcal{X}}_k$, inspired by soft-thresholding operator, we have

$$\boldsymbol{\mathcal{E}}_{k+1} = \mathrm{T}_{\frac{\lambda}{\mu_k}}(\boldsymbol{\mathcal{H}}_k) \tag{16}$$

where, the (i,j,k)-th element of $\mathrm{T}_{\frac{\lambda}{\mu_k}}(\boldsymbol{\mathcal{H}}_k)$ is $sign((\boldsymbol{\mathcal{H}}_k)_{i,j,k}) \bullet max(|(\boldsymbol{\mathcal{H}}_k)_{i,j,k}| - \lambda/\mu_k, 0)$.

$\boldsymbol{\mathcal{X}}$-subproblem: The model (14) becomes

$$\arg\min_{\boldsymbol{\mathcal{X}}} \mu_k^{-1}||\boldsymbol{\mathcal{X}}||_{\boldsymbol{\omega},S_p}^p + \tfrac{1}{2}||\boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{M}}_k||_F^2 \tag{17}$$

where $\boldsymbol{\mathcal{M}}_k = \boldsymbol{\mathcal{Y}} + \mu_k^{-1}\boldsymbol{\mathcal{L}}_k - \boldsymbol{\mathcal{E}}_{k+1}$ which is called WTSNM. To solve it, we first introduce the following Lemma and Theorems.

Lemma 1. [18]For the following optimization problem:

$$\min_{\delta \geq 0} \quad f(\delta) = \frac{1}{2}(\delta - \sigma)^2 + \omega\delta^p \tag{18}$$

with the given p and ω, there exists a specific threshold:

$$\tau_p^{GST}(\omega) = (2\omega(1-p))^{\frac{1}{2-p}} + \omega p(2\omega(1-p))^{\frac{p-1}{2-p}} \tag{19}$$

we have the following conclusion.

1) When $\sigma \leq \tau_p^{GST}(\omega)$, the optimal solution $T_p^{GST}(\sigma,\omega)$ of Eq. (18) is 0.
2) When $\sigma > \tau_p^{GST}(\omega)$, the optimal solution is $T_p^{GST}(\sigma,\omega) = sign(\sigma)S_p^{GST}(\sigma,\omega)$ and $S_p^{GST}(\sigma,\omega)$ can be obtain by solving $S_p^{GST}(\sigma,\omega) - \sigma + \omega p(S_p^{GST}(\sigma,\omega))^{p-1} = 0$.

Theorem 3. [18] Let $\mathbf{Y} = \mathbf{U}_Y\mathbf{D}_Y\mathbf{V}_Y^T$ be the SVD of $\mathbf{Y} \in \mathbb{R}^{m\times n}$, $\tau > 0$, $l = \min(m,n)$, $0 \leq \omega_1 \leq \omega_2 \leq ... \leq \omega_l$, a global optimal solution of the following model

$$\arg\min_{\mathbf{X}} \frac{1}{2}||\mathbf{X} - \mathbf{Y}||_F^2 + \tau||\mathbf{X}||_{\boldsymbol{\omega},S_p}^p \tag{20}$$

is

$$\Upsilon_{\tau*\boldsymbol{\omega}}[\mathbf{Y}] = \mathbf{U}_Y\mathbf{P}_{\tau*\boldsymbol{\omega}}(\mathbf{Y})\mathbf{V}_Y^T \tag{21}$$

where, $\mathbf{P}_{\tau*\boldsymbol{\omega}}(\mathbf{Y}) = diag(\gamma_1, \gamma_2, \cdots, \gamma_l)$ and $\gamma_i = T_p^{GST}(\sigma_i(\mathbf{Y}), \tau * \omega_i)$ which can be obtained by Lemma 1.

The fact that a closed-form global minimizer can be found comes from von Neumann's trace inequality [19], $\{\sigma_i(\mathbf{Y})\}$ is in nonincreasing order, while $\{\omega_i\}$ is in nondecreasing order.

Theorem 4. Suppose $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1\times n_2\times n_3}$, $l = \min(n_1,n_2)$,$0 \leq \omega_1 \leq \omega_2 \leq ... \leq \omega_l$, let $\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{U}} * \boldsymbol{\mathcal{S}} * \boldsymbol{\mathcal{V}}^T$ Given the model

$$\arg\min_{\boldsymbol{\mathcal{X}}} \frac{1}{2}||\boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{A}}||_F^2 + \tau||\boldsymbol{\mathcal{X}}||_{\boldsymbol{\omega},S_p}^p \tag{22}$$

Then, a global optimal solution to the model (22) is

$$\boldsymbol{\mathcal{X}}^* = \Upsilon_{\tau*\boldsymbol{\omega}}(\boldsymbol{\mathcal{A}}) = \boldsymbol{\mathcal{U}} * ifft(\mathbf{P}_{\tau*\boldsymbol{\omega}}(\overline{\boldsymbol{\mathcal{A}}})) * \boldsymbol{\mathcal{V}}^T \tag{23}$$

where, $\mathbf{P}_{\tau*\boldsymbol{\omega}}(\overline{\boldsymbol{\mathcal{A}}})$ is a tensor and $\mathbf{P}_{\tau*\boldsymbol{\omega}}(\overline{\mathbf{A}}^{(i)})$ is the i-th frontal slice of $\mathbf{P}_{\tau*\boldsymbol{\omega}}(\overline{\boldsymbol{\mathcal{A}}})$.

Proof: In Fourier domain, the model (22) becomes

$$\arg\min_{\overline{\boldsymbol{\mathcal{X}}}} \frac{1}{2}||\overline{\boldsymbol{\mathcal{X}}} - \overline{\boldsymbol{\mathcal{A}}}||_F^2 + \sum\nolimits_{i=1}^{n3} \tau * ||\bar{\mathbf{X}}^{(i)}||_{\boldsymbol{\omega},S_p}^p \tag{24}$$

According to Definition 1, we have

$$\arg\min_{\overline{\boldsymbol{\mathcal{X}}}} \sum_{i=1}^{n_3} (\tfrac{1}{2}||\overline{\mathbf{X}}^{(i)} - \overline{\mathbf{A}}^{(i)}||_F^2 + \tau * ||\bar{\mathbf{X}}^{(i)}||_{\boldsymbol{\omega},S_p}^p) \tag{25}$$

where, $\overline{\mathbf{X}}^{(i)}$ is the i-th frontal slice of $\overline{\boldsymbol{\mathcal{X}}}$.

In Eq. (25), each variable $\overline{\mathbf{X}}^{(i)}$ is independent. Thus, it can be divided into n_3 independent subproblems. For the i-th ($i = 1, 2, \cdots, n_3$) subproblem, we have

$$\arg\min_{\overline{\mathbf{X}}^{(i)}} \tfrac{1}{2}||\overline{\mathbf{X}}^{(i)} - \overline{\mathbf{A}}^{(i)}||_F^2 + \tau * ||\bar{\mathbf{X}}^{(i)}||_{\boldsymbol{\omega},S_p}^p \tag{26}$$

According to Theorem 3, the global optimal solution of Eq. (26) is $\overline{\mathbf{X}}^{(i)*} = \Upsilon_{\tau*\boldsymbol{\omega}}[\overline{\mathbf{A}}^{(i)}] = \overline{\mathbf{U}}^{(i)}\mathbf{P}_{\tau*\boldsymbol{\omega}}(\overline{\mathbf{A}}^{(i)})\overline{\mathbf{V}}^{(i)T}$, which is the i-th frontal slice of $\overline{\boldsymbol{\mathcal{X}}}^*$. Since we get global solutions of all subproblems, according to Definition 5, we can easily get the global solution of the optimization problem (22), i.e.,

$$\boldsymbol{\mathcal{X}}^* = \Upsilon_{\tau*\boldsymbol{\omega}}[\boldsymbol{\mathcal{A}}] = \boldsymbol{\mathcal{U}} * ifft(\mathbf{P}_{\tau*\boldsymbol{\omega}}(\overline{\boldsymbol{\mathcal{A}}})) * \boldsymbol{\mathcal{V}}^T \tag{27}$$

where $\boldsymbol{\mathcal{U}} = ifft(\overline{\boldsymbol{\mathcal{U}}}, [], 3)$ and $\boldsymbol{\mathcal{V}} = ifft(\overline{\boldsymbol{\mathcal{V}}}, [], 3)$. ■

Now, we consider how to solve the model (17). According to Theorem 4, the solution is

$$\boldsymbol{\mathcal{X}}_{k+1} = \Upsilon_{\mu_k^{-1}*\boldsymbol{\omega}}(\boldsymbol{\mathcal{M}}_k) \tag{28}$$

The pseudo code is summarized in Algorithm 1.

4 Convergence Analysis

To prove the convergence of Algorithm 1, we first prove the bounded of sequences $\{\boldsymbol{\mathcal{L}}_k\}$, $\{\boldsymbol{\mathcal{E}}_k\}$ and $\{\boldsymbol{\mathcal{X}}_k\}$ (See Theorems 5 and 6), then show convergence (See Theorem 8).

Theorem 5. The sequence $\{\boldsymbol{\mathcal{L}}_k\}$, which is generated by Algorithm 1, is bounded.

Proof: In the $(k+1)$-th iteration in Algorithm 1, we have

$$\begin{aligned} &||\boldsymbol{\mathcal{L}}_{k+1}||_F = ||\boldsymbol{\mathcal{L}}_k + \mu_k(\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{X}}_{k+1} - \boldsymbol{\mathcal{E}}_{k+1})||_F \\ &= \mu_k \times ||\mu_k^{-1}\boldsymbol{\mathcal{L}}_k + \boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{X}}_{k+1} - \boldsymbol{\mathcal{E}}_{k+1}||_F \\ &= \tfrac{\mu_k}{\sqrt{n_3}}||bdiag(\overline{\boldsymbol{\mathcal{Y}} + \mu_k^{-1}\boldsymbol{\mathcal{L}}_k - \boldsymbol{\mathcal{E}}_{k+1} - \boldsymbol{\mathcal{X}}_{k+1}})||_F \\ &= \tfrac{\mu_k}{\sqrt{n_3}}||bdiag(\overline{\boldsymbol{\mathcal{M}}_k}) - bdiag(\overline{\boldsymbol{\mathcal{X}}_{k+1}})||_F \end{aligned} \tag{29}$$

0162-8828 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Cornell University Library. Downloaded on August 22,2020 at 18:51:09 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2020.3017672, IEEE Transactions on Pattern Analysis and Machine Intelligence

$$\boldsymbol{\mathcal{X}}_{k+1}=\boldsymbol{\mathcal{U}}_k * ifft(\mathbf{P}_{\mu_k^{-1}*\boldsymbol{\omega}}(\overline{\boldsymbol{\mathcal{M}}_k})) * \boldsymbol{\mathcal{V}}_k^T \tag{30}$$

Denote by $\boldsymbol{\mathcal{U}}_k * \boldsymbol{\mathcal{S}}_k * \boldsymbol{\mathcal{V}}_k^T$ the t-SVD of $\boldsymbol{\mathcal{M}}_k$, and substituting it and Eq. (30) into Eq. (29), we have

$$\begin{aligned}
&||\boldsymbol{\mathcal{L}}_{k+1}||_F = \mu_k \times \tfrac{1}{\sqrt{n_3}}||bdiag(\overline{\boldsymbol{\mathcal{U}}_k}) \bullet (bdiag(\overline{\boldsymbol{\mathcal{S}}_k}) \\
&-bdiag(\mathbf{P}_{\mu_k^{-1}*\boldsymbol{\omega}}(\overline{\boldsymbol{\mathcal{M}}_k})))\bullet bdiag(\overline{\boldsymbol{\mathcal{V}}_k}^T)||_F \\
&=\mu_k \times \tfrac{1}{\sqrt{n_3}}||bdiag(\overline{\boldsymbol{\mathcal{S}}_k})-bdiag(\mathbf{P}_{\mu_k^{-1}*\boldsymbol{\omega}}(\overline{\boldsymbol{\mathcal{M}}_k}))||_F \\
&\leq \mu_k \times \tfrac{1}{\sqrt{n_3}}\sqrt{\sum_{j=1}^{n_3}\sum_i\left(\tfrac{J\omega_i}{\mu_k}\right)^2} = \tfrac{1}{\sqrt{n_3}}\sqrt{\sum_{j=1}^{n_3}\sum_i J^2{\omega_i}^2}
\end{aligned} \tag{31}$$

where, J is the number of iterations in solving the model (18). Thus, $\{\boldsymbol{\mathcal{L}}_k\}$ is bounded. ■

Algorithm 1: Solving Eq. (13)

Input: Observation data $\boldsymbol{\mathcal{Y}}$, weight vector $\boldsymbol{\omega}$.
Initialize: $\mu_0 = 1e-4$, $\rho = 1.1$, $k = 0$, μ_{max}=$1e10$, ε=$1e-8$, $\boldsymbol{\mathcal{X}}_0 = \boldsymbol{\mathcal{E}}_0 = \mathbf{0}$, $\boldsymbol{\mathcal{L}}_0 = \mathbf{0}$.
while not converged **do**
1. Update $\boldsymbol{\mathcal{E}}_{k+1}$ by the model (16).
2. Update$\boldsymbol{\mathcal{X}}_{k+1}$ by the model (28).
3. Update $\boldsymbol{\mathcal{L}}_{k+1} = \boldsymbol{\mathcal{L}}_k + \mu_k(\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{X}}_{k+1} - \boldsymbol{\mathcal{E}}_{k+1})$.
4. Update μ_{k+1} by $\mu_{k+1} = \min(\rho\mu_k, \mu_{max})$.
5. Check the convergence conditions
$||\boldsymbol{\mathcal{E}}_{k+1} - \boldsymbol{\mathcal{E}}_k||_F \leq \varepsilon$, $||\boldsymbol{\mathcal{X}}_{k+1} - \boldsymbol{\mathcal{X}}_k||_F \leq \varepsilon$
$||\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{X}}_{k+1} - \boldsymbol{\mathcal{E}}_{k+1}||_F \leq \varepsilon$
end while
6. Output $\boldsymbol{\mathcal{E}}$, $\boldsymbol{\mathcal{X}}$

Theorem 6. The sequences $\{\boldsymbol{\mathcal{E}}_k\}$ and $\{\boldsymbol{\mathcal{X}}_k\}$, which are generated by Algorithm 1, are bounded.

Proof: In step 1 and step 2 in Algorithm 1, since both $\boldsymbol{\mathcal{E}}$ and $\mathbf{X}$ subproblems have optimal solution, we have

$$\Gamma(\boldsymbol{\mathcal{E}}_{k+1}, \mathbf{X}_{k+1}, \boldsymbol{\mathcal{L}}_k, \mu_k) \leq \Gamma(\boldsymbol{\mathcal{E}}_k, \mathbf{X}_k, \boldsymbol{\mathcal{L}}_k, \mu_k) \tag{32}$$

According to step 3 in Algorithm 1, we have

$$\boldsymbol{\mathcal{L}}_{k+1} = \boldsymbol{\mathcal{L}}_k + \mu_k(\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{X}}_{k+1} - \boldsymbol{\mathcal{E}}_{k+1}) \tag{33}$$

Then,

$$\begin{aligned}
&\Gamma(\boldsymbol{\mathcal{E}}_k, \boldsymbol{\mathcal{X}}_k, \boldsymbol{\mathcal{L}}_k, \mu_k)=\Gamma(\boldsymbol{\mathcal{E}}_k, \boldsymbol{\mathcal{X}}_k, \boldsymbol{\mathcal{L}}_{k-1}, \mu_{k-1}) \\
&+\tfrac{\mu_k-\mu_{k-1}}{2}||\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{X}}_k - \boldsymbol{\mathcal{E}}_k||_F^2 + \langle \boldsymbol{\mathcal{L}}_k - \boldsymbol{\mathcal{L}}_{k-1}, \boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{X}}_k - \boldsymbol{\mathcal{E}}_k\rangle \\
&= \Gamma(\boldsymbol{\mathcal{E}}_k, \boldsymbol{\mathcal{X}}_k, \boldsymbol{\mathcal{L}}_{k-1}, \mu_{k-1}) + \tfrac{\mu_k-\mu_{k-1}}{2}||\mu_{k-1}^{-1}(\boldsymbol{\mathcal{L}}_k - \boldsymbol{\mathcal{L}}_{k-1})||_F^2 \\
&+\left\langle \boldsymbol{\mathcal{L}}_k - \boldsymbol{\mathcal{L}}_{k-1}, \mu_{k-1}^{-1}(\boldsymbol{\mathcal{L}}_k - \boldsymbol{\mathcal{L}}_{k-1})\right\rangle \\
&= \Gamma(\boldsymbol{\mathcal{E}}_k, \boldsymbol{\mathcal{X}}_k, \boldsymbol{\mathcal{L}}_{k-1}, \mu_{k-1}) + \tfrac{\mu_k+\mu_{k-1}}{2\mu_{k-1}^2}||\boldsymbol{\mathcal{L}}_k - \boldsymbol{\mathcal{L}}_{k-1}||_F^2
\end{aligned} \tag{34}$$

Denote by Θ the bound of $||\boldsymbol{\mathcal{L}}_k - \boldsymbol{\mathcal{L}}_{k-1}||_F^2$ for all $\{k = 1, ..., \infty\}$, and combining Eq. (32), we have

$$\begin{aligned}
\Gamma(\boldsymbol{\mathcal{E}}_{k+1}, \boldsymbol{\mathcal{X}}_{k+1}, \boldsymbol{\mathcal{L}}_k, \mu_k) &\leq \Gamma(\boldsymbol{\mathcal{E}}_1, \boldsymbol{\mathcal{X}}_1, \boldsymbol{\mathcal{L}}_0, \mu_0) \\
&+ \Theta\sum_{k=1}^{\infty}\frac{\mu_k + \mu_{k-1}}{2\mu_{k-1}^2}
\end{aligned} \tag{35}$$

According to [20], we have that the penalty parameter $\{\mu_k\}$ satisfies $\sum_{k=1}^{\infty}\frac{\mu_{k+1}}{\mu_k^2} < +\infty$, then,

$$\sum_{k=1}^{\infty}\frac{\mu_k + \mu_{k-1}}{2\mu_{k-1}^2} \leq \sum_{k=1}^{\infty}\frac{\mu_k}{\mu_{k-1}^2} < +\infty \tag{36}$$

Thus, $\Gamma(\boldsymbol{\mathcal{E}}_{k+1}, \boldsymbol{\mathcal{X}}_{k+1}, \boldsymbol{\mathcal{L}}_k, \mu_k)$ is upper bounded. The boundedness of sequences $\{\boldsymbol{\mathcal{X}}_{k+1}\}$ and $\{\boldsymbol{\mathcal{E}}_{k+1}\}$ can be deduced as follows:

$$\begin{aligned}
&\lambda||\boldsymbol{\mathcal{E}}_k||_1 + ||\boldsymbol{\mathcal{X}}_k||_{\boldsymbol{\omega},S_p}^p \\
&= \Gamma(\boldsymbol{\mathcal{E}}_k, \boldsymbol{\mathcal{X}}_k, \boldsymbol{\mathcal{L}}_{k-1}, \mu_{k-1}) + \tfrac{\mu_{k-1}}{2}(\tfrac{1}{\mu_{k-1}^2}||\boldsymbol{\mathcal{L}}_{k-1}||_F^2 \\
&-||\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{X}}_k - \boldsymbol{\mathcal{E}}_k + \tfrac{1}{\mu_{k-1}}\boldsymbol{\mathcal{L}}_{k-1}||_F^2) \\
&= \Gamma(\boldsymbol{\mathcal{E}}_k, \boldsymbol{\mathcal{X}}_k, \boldsymbol{\mathcal{L}}_{k-1}, \mu_{k-1}) - \tfrac{1}{2\mu_{k-1}}(||\boldsymbol{\mathcal{L}}_k||_F^2 - ||\boldsymbol{\mathcal{L}}_{k-1}||_F^2)
\end{aligned} \tag{37}$$

Since $\{\boldsymbol{\mathcal{L}}_k\}$, which is generated by Algorithm 1, and $\Gamma(\boldsymbol{\mathcal{E}}_{k+1}, \boldsymbol{\mathcal{X}}_{k+1}, \boldsymbol{\mathcal{L}}_k, \mu_k)$ are all bounded, in addition, $||\boldsymbol{\mathcal{E}}_k||_1$ and $||\boldsymbol{\mathcal{X}}_k||_{\boldsymbol{\omega},S_p}^p$ are all nonnegative, thus $\{\boldsymbol{\mathcal{X}}_k\}$, and $\{\boldsymbol{\mathcal{E}}_k\}$ are all bounded. ■

Theorem 7. (**Bolzano-Weierstrass theorem**) [21] Every bounded sequence of real numbers has a convergent subsequence.

Theorem 8. If the weights are sorted in non-descending order, then $\{\boldsymbol{\mathcal{E}}_k\}$ and $\{\boldsymbol{\mathcal{X}}_k\}$ generated by Algorithm 1 satisfy

1) $\lim_{k\to\infty} ||\boldsymbol{\mathcal{X}}_{k+1} - \boldsymbol{\mathcal{X}}_k||_F = 0$
2) $\lim_{k\to\infty} ||\boldsymbol{\mathcal{E}}_{k+1} - \boldsymbol{\mathcal{E}}_k||_F = 0$
3) $\lim_{k\to\infty} ||\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{X}}_{k+1} - \boldsymbol{\mathcal{E}}_{k+1}||_F = 0$

Proof: According to Theorem 5 and Theorem 6, we know $\{\boldsymbol{\mathcal{X}}_k\}$, $\{\boldsymbol{\mathcal{E}}_k\}$ and $\{\boldsymbol{\mathcal{L}}_k\}$ are all bounded. According to Theorem 7, $\{\boldsymbol{\mathcal{X}}_k, \boldsymbol{\mathcal{E}}_k, \boldsymbol{\mathcal{L}}_k\}$ have at least one convergent subsequence, respectively, so there exists at least one accumulation point for $\{\boldsymbol{\mathcal{X}}_k, \boldsymbol{\mathcal{E}}_k, \boldsymbol{\mathcal{L}}_k\}$. Specifically, we have

$$\lim_{k\to\infty} ||\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{X}}_{k+1} - \boldsymbol{\mathcal{E}}_{k+1}||_F = \lim_{k\to\infty}\tfrac{1}{\mu_k}||\boldsymbol{\mathcal{L}}_{k+1} - \boldsymbol{\mathcal{L}}_k||_F = 0 \tag{38}$$

Thus, the accumulation point is a feasible solution of the model (13). Then, for the $\boldsymbol{\mathcal{E}}$ subproblem in Algorithm 1, we have

$$\begin{aligned}
&\lim_{k\to\infty} ||\boldsymbol{\mathcal{E}}_{k+1} - \boldsymbol{\mathcal{E}}_k||_F = \lim_{k\to\infty} ||\mathrm{T}_{\frac{\lambda}{\mu_k}}(\boldsymbol{\mathcal{H}}_k) - \boldsymbol{\mathcal{H}}_k \\
&+\mu_k^{-1}\boldsymbol{\mathcal{L}}_k+\mu_{k-1}^{-1}(\boldsymbol{\mathcal{L}}_k - \boldsymbol{\mathcal{L}}_{k-1})||_F \leq \lim_{k\to\infty}\tfrac{\lambda n_1 n_2 n_3}{\mu_k} \\
&+||\mu_k^{-1}\boldsymbol{\mathcal{L}}_k + \mu_{k-1}^{-1}(\boldsymbol{\mathcal{L}}_k - \boldsymbol{\mathcal{L}}_{k-1})||_F = 0
\end{aligned} \tag{39}$$

where n_1, n_2 and n_3 are size of $\boldsymbol{\mathcal{Y}}$.
In the k-th iteration, we have

$$\boldsymbol{\mathcal{X}}_k=\boldsymbol{\mathcal{U}}_{k-1} * ifft(\mathbf{P}_{\mu_{k-1}^{-1}*\boldsymbol{\omega}}(\boldsymbol{\mathcal{M}}_{k-1})) * \boldsymbol{\mathcal{V}}_{k-1}^T \tag{40}$$

According to step 3 in Algorithm 1, we have

$$\boldsymbol{\mathcal{X}}_{k+1} = \boldsymbol{\mathcal{Y}}+\mu_k^{-1}(\boldsymbol{\mathcal{L}}_{k+1} - \boldsymbol{\mathcal{L}}_k) - \boldsymbol{\mathcal{E}}_{k+1} \tag{41}$$

Denote by $\boldsymbol{\mathcal{U}}_{k-1} * \boldsymbol{\mathcal{S}}_{k-1} * \boldsymbol{\mathcal{V}}_{k-1}^T$ the t-SVD of $\boldsymbol{\mathcal{M}}_{k-1}$, then,

$$\begin{aligned}
&\lim_{k\to\infty} ||\boldsymbol{\mathcal{X}}_{k+1} - \boldsymbol{\mathcal{X}}_k||_F \\
&= \lim_{k\to\infty} ||\boldsymbol{\mathcal{Y}}+\mu_k^{-1}(\boldsymbol{\mathcal{L}}_k - \boldsymbol{\mathcal{L}}_{k+1}) - \boldsymbol{\mathcal{E}}_{k+1} - \boldsymbol{\mathcal{X}}_k||_F \\
&= \lim_{k\to\infty} ||(\boldsymbol{\mathcal{Y}}+\mu_k^{-1}\boldsymbol{\mathcal{L}}_k - \mu_k^{-1}\boldsymbol{\mathcal{L}}_{k+1} - \boldsymbol{\mathcal{E}}_{k+1}) - \boldsymbol{\mathcal{X}}_k \\
&+(\boldsymbol{\mathcal{E}}_k + \mu_{k-1}^{-1}\boldsymbol{\mathcal{L}}_{k-1}) - (\boldsymbol{\mathcal{E}}_k + \mu_{k-1}^{-1}\boldsymbol{\mathcal{L}}_{k-1})||_F \leq \\
&\lim_{k\to\infty} ||\boldsymbol{\mathcal{M}}_{k-1} - \boldsymbol{\mathcal{X}}_k||_F + ||\boldsymbol{\mathcal{E}}_k - \boldsymbol{\mathcal{E}}_{k+1}||_F + ||\mu_k^{-1}\boldsymbol{\mathcal{L}}_k \\
&- \mu_k^{-1}\boldsymbol{\mathcal{L}}_{k+1} - \mu_{k-1}^{-1}\boldsymbol{\mathcal{L}}_{k-1}||_F = \lim_{k\to\infty}(||\boldsymbol{\mathcal{E}}_k - \boldsymbol{\mathcal{E}}_{k+1}||_F \\
&+ \frac{1}{\sqrt{n_3}}||bdiag(\overline{\boldsymbol{\mathcal{S}}_{k-1}})-bdiag(\mathbf{P}_{\mu_{k-1}^{-1}*\boldsymbol{\omega}}(\overline{\boldsymbol{\mathcal{M}}_{k-1}}))||_F \\
&+ ||\mu_k^{-1}\boldsymbol{\mathcal{L}}_k - \mu_k^{-1}\boldsymbol{\mathcal{L}}_{k+1} - \mu_{k-1}^{-1}\boldsymbol{\mathcal{L}}_{k-1}||_F) \\
&= \lim_{k\to\infty}\frac{1}{\sqrt{n_3}\mu_k}\sqrt{\sum_{j=1}^{n_3}\sum_i J^2{\omega_i}^2}=0
\end{aligned} \tag{42}$$

0162-8828 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Cornell University Library. Downloaded on August 22,2020 at 18:51:09 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2020.3017672, IEEE Transactions on Pattern Analysis and Machine Intelligence

TABLE 1: Correct recovery for random problems of varying sizes ($r = rank_t(\mathcal{D}_0) = 0.1n, m = ||\mathcal{E}_0||_0 = 0.1n^3$).

n	r	m	Algorithm	$rank_t(\widehat{\mathcal{D}})$	$\|\widehat{\mathcal{E}}\|_0$	$\frac{\|\widehat{\mathcal{D}}-\mathcal{D}_0\|_F}{\|\mathcal{D}_0\|_F}$	$\frac{\|\widehat{\mathcal{E}}-\mathcal{E}_0\|_F}{\|\mathcal{E}_0\|_F}$
100	10	1e5	RPCA	100	564266	6.98 e-01	7.00 e-03
			TRPCA	10	101827	2.27e-07	9.97e-10
			TPSSV	10	869417	3.21e+01	3.21e-01
			ETRPCA	10	100000	7.48e-08	4.82e-10
200	20	8e5	RPCA	200	4521630	7.03e-01	3.52e-03
			TRPCA	20	816203	5.57e-07	1.04e-09
			TPSSV	20	7054878	5.51e+01	2.76e-01
			ETRPCA	20	800002	5.27e-08	9.15e-11
300	30	27e5	RPCA	300	15274692	7.05e-01	2.35e-03
			TRPCA	30	2754349	5.69e-07	7.04e-10
			TPSSV	30	24118681	6.19e+01	2.06e-01
			ETRPCA	30	2700000	1.17e-07	5.46e-11

5 Relationship with Related Work

This section introduces the relationships of our model with the related work. Fig. 1 shows that the connections between our method and most representative methods such as RPCA [1], TRPCA [13], PSSV [9], TPSSV [14], TRPCA-WTNN [11], [12], WRPCA [7], [8], and RPCA-WSNM [18]. Due to the space limit, we herein prove the relationship between RPCA and our model as follows.

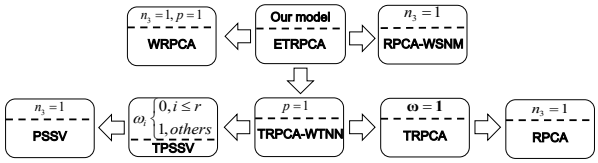

Fig. 1: The relationship graph of our model with the related work

Theorem 9. Our model (13) reduces to RPCA when $n_3 = 1$, $\omega = \mathbf{1}$ and $p = 1$.

Proof: When $n_3 = 1$, then $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{Y}}$ become second-order matrices, denoting $\mathbf{X}$ and $\mathbf{Y}$, respectively.

From Definition 3, we have

$$bcirc(\boldsymbol{\mathcal{X}}) = \left[\mathbf{X}^{(1)}\right] = \mathbf{X} \tag{43}$$

According to Theorem 1 and combing Eq. (43), we have

$$\begin{aligned} bdiag(\overline{\boldsymbol{\mathcal{X}}}) &= \left[\overline{\mathbf{X}}^{(1)}\right] \\ &= (\mathbf{F}_1 \otimes \mathbf{I}_{n_1}) \cdot bcirc(\boldsymbol{\mathcal{X}}) \cdot (\mathbf{F}_1^{-1} \otimes \mathbf{I}_{n_2}) \\ &= (\mathbf{F}_1 \otimes \mathbf{I}_{n_1}) \cdot \mathbf{X} \cdot (\mathbf{F}_1^{-1} \otimes \mathbf{I}_{n_2}) \end{aligned} \tag{44}$$

where $\mathbf{F}_1$, denoting the 1×1 Discrete Fourier Transform(DFT) matrix, is a constant 1. Then, $\overline{\mathbf{X}}^{(1)} = \mathbf{X}$.

According to Definition 7, we have

$$||\boldsymbol{\mathcal{X}}||^p_{\boldsymbol{\omega},S_p} = ||\overline{\mathbf{X}}^{(1)}||_{\boldsymbol{\omega},S_p} = ||\mathbf{X}||_{\boldsymbol{\omega},S_p} \tag{45}$$

Thus, if $n_3 = 1$, $p = 1$, $\boldsymbol{\omega} = \mathbf{1}$, our model reduces to RPCA. ■

6 Experimental Results

In this section, we first compare our proposed method ETRPCA with some classical and recently proposed methods such as RPCA [1], WRPCA [7], [8], PSSV [9], TRPCA [13] and TPSSV [14] under synthetic data sets, and then apply them for image recovery and background modelling in the following subsections.

TABLE 2: Correct recovery for random problems of varying sizes ($r = rank_t(\mathcal{D}_0) = 0.1n, m = ||\mathcal{E}_0||_0 = 0.2n^3$).

n	r	m	Algorithm	$rank_t(\widehat{\mathcal{D}})$	$\|\widehat{\mathcal{E}}\|_0$	$\frac{\|\widehat{\mathcal{D}}-\mathcal{D}_0\|_F}{\|\mathcal{D}_0\|_F}$	$\frac{\|\widehat{\mathcal{E}}-\mathcal{E}_0\|_F}{\|\mathcal{E}_0\|_F}$
100	10	2e5	RPCA	100	567144	8.00e-01	5.65e-03
			TRPCA	10	200009	5.48e-07	2.98e-09
			TPSSV	10	880479	4.05e+01	2.86e-01
			ETRPCA	10	200000	1.65e-07	8.62e-10
200	20	16e5	RPCA	200	4541146	8.01e-01	2.84e-03
			TRPCA	20	1600218	6.24e-07	1.68e-09
			TPSSV	20	7083855	7.23e+01	2.56e-01
			ETRPCA	20	1600000	2.99e-07	7.49e-10
300	30	54e5	RPCA	300	15335876	8.03e-01	1.89e-03
			TRPCA	30	5401678	7.74e-07	1.37e-09
			TPSSV	30	24174049	8.28e+01	1.95e-01
			ETRPCA	30	5400000	1.23e-07	2.17e-09

TABLE 3: Correct recovery with varying p and ω ($n = 100$, $r = rank_t(\mathcal{D}_0) = 0.1n, m = ||\mathcal{E}_0||_0 = 0.2n^3$).

m	p	ω	$rank_t(\widehat{\mathcal{D}})$	$\|\widehat{\mathcal{E}}\|_0$	$\frac{\|\widehat{\mathcal{D}}-\mathcal{D}_0\|_F}{\|\mathcal{D}_0\|_F}$	$\frac{\|\widehat{\mathcal{E}}-\mathcal{E}_0\|_F}{\|\mathcal{E}_0\|_F}$
2e5	1	$1, 1, 1$	10	200009	5.48e-07	2.98
	1	$0.001, 0.5, 1$	15	189646	32.2191	0.2275
	1	$0.1, 0.5, 1$	14	191667	28.9027	0.2041
	1	$0.5, 0.5, 1$	10	200000	1.79e-07	9.49e-10
	1	$0.8, 1, 10$	10	200000	1.71e-07	9.05e-10
2e5	0.8	$0.8, 1, 10$	10	200000	1.89e-07	9.99e-10
	0.85	$0.8, 1, 10$	10	200000	1.55e-07	8.12e-10
	0.9	$0.8, 1, 10$	10	200000	4.25e-07	2.29e-09

TABLE 4: Correct recovery under different distributions ($r = rank_t(\mathcal{D}_0) = 0.1n, m = ||\mathcal{E}_0||_0 = 0.1n^3, n = 100$).

Distribution	Method	$rank_t(\widehat{\mathcal{D}})\|\widehat{\mathcal{E}}\|_0$		$\frac{\|\widehat{\mathcal{D}}-\mathcal{D}_0\|_F}{\|\mathcal{D}_0\|_F}$	$\frac{\|\widehat{\mathcal{E}}-\mathcal{E}_0\|_F}{\|\mathcal{E}_0\|_F}$
$\mathcal{P}, \mathcal{Q} \sim N\left(0, \frac{1}{n}\right)$ $\mathcal{E} \sim B(0.5)$	RPCA	100	564266	6.98e-01	7.00e-03
	TRPCA	10	101827	2.27e-07	9.97e-10
	TPSSV	10	869417	3.21e+01	3.21e-01
	ETRPCA	10	100000	7.48e-08	4.82e-10
$\mathcal{P}, \mathcal{Q} \sim N(0, 1)$ $\mathcal{E} \sim B(0.5)$	RPCA	100	561772	6.24e-01	6.26e+01
	TRPCA	10	100000	3.15e-11	2.16e-09
	TPSSV	10	100000	2.04e-11	1.41e-09
	ETRPCA	10	100000	1.43e-11	9.78e-10
$\mathcal{P}, \mathcal{Q} \sim N(0, 1)$ $\mathcal{E} \sim La(0, 1)$	RPCA	100	561576	0.626	19.812
	TRPCA	54	537670	0.019	0.600
	TPSSV	11	910456	0.013	0.410
	ETRPCA	10	912788	0.011	0.410

6.1 Exact recovery from varying fractions of error

We verify the correct recovery of low-rank and sparse components on random data with different fractions of error. First, we generate a low rank tensor $\boldsymbol{\mathcal{D}}_0 = \boldsymbol{\mathcal{P}} * \boldsymbol{\mathcal{Q}}$ with tubal rank r according to Theorem 4.1 in [13], where $\boldsymbol{\mathcal{P}} \in \mathbf{R}^{n \times r \times n}$ and $\boldsymbol{\mathcal{Q}} \in \mathbf{R}^{r \times n \times n}$ are independently sampled from $N(0, 1/n)$ distribution. The size of tensor $\boldsymbol{\mathcal{D}}_0$ is $n \times n \times n$ with varying choices of dimension (n =100, 200 and 300). Then, we generate a sparse tensor $\boldsymbol{\mathcal{E}}$ ($||\boldsymbol{\mathcal{E}}_0||_0 = m$) whose non-zero entries satisfy independent Bernoulli distribution ± 1. We test on two settings. In the first experiments, we set $r = rank_t(\boldsymbol{\mathcal{D}}_0) = 0.1n$ and $m = ||\boldsymbol{\mathcal{E}}_0||_0 = 0.1n^3$. In the second experiments, we set $r = rank_t(\boldsymbol{\mathcal{D}}_0) = 0.1n$ and $m = ||\boldsymbol{\mathcal{E}}_0||_0 = 0.2n^3$. Tab. 1 and Tab. 2 list the recovery results of several methods on varying choices of n, respectively.

From Tab. 1 and Tab. 2, we can see that: (1) Tensor methods

0162-8828 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Cornell University Library. Downloaded on August 22,2020 at 18:51:09 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2020.3017672, IEEE Transactions on Pattern Analysis and Machine Intelligence

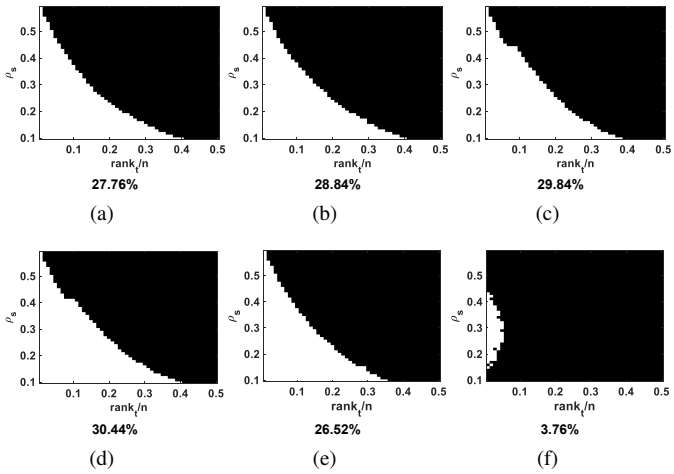

Fig. 2: Correct recovery for varying rank and sparsity. (a)-(d)are ETRPCA under different weights; (e) is TRPCA, (f) is TPSSV;

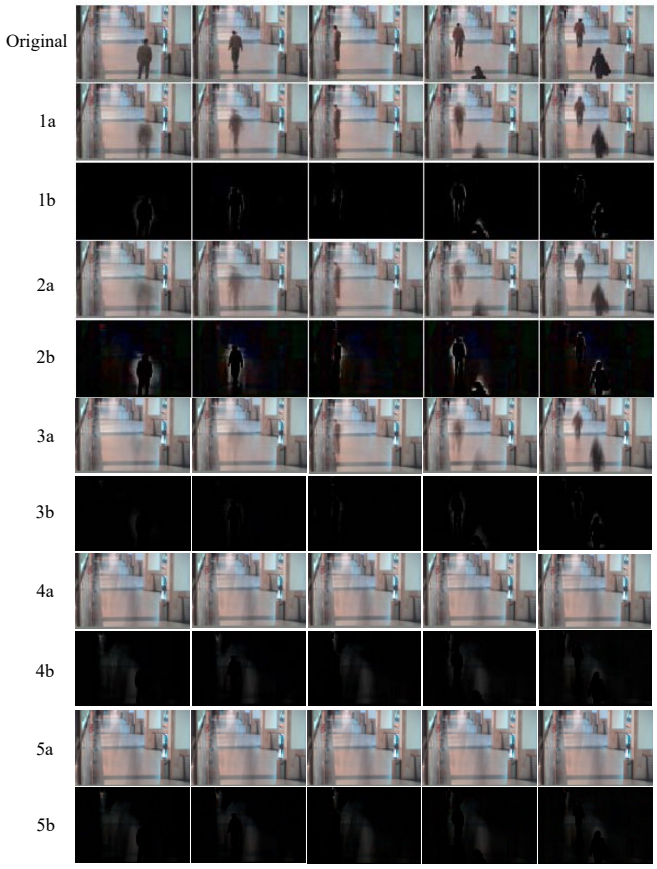

Fig. 3: Background modeling from videos. Original frames(first row), RPCA(1a and 1b), TRPCA(2a and 2b), WRPCA (3a and 3b),TPSSV (4a and 4b), and ETRPCA (5a and 5b).

can give the correct rank estimation, while RPCA does not. The reason may be that tensor methods well exploit spatial structure due to the fact that they do not need to transform each tensor into a matrix. (2) Compared with the other methods, our method ETRPCA achieves the smallest relative errors about $\|\widehat{\boldsymbol{\mathcal{D}}} - \boldsymbol{\mathcal{D}}_0\|_F / \|\boldsymbol{\mathcal{D}}_0\|_F$ with the correct estimation of rank. Moreover, ETRPCA overall gives the correct estimation of the sparsity of $\boldsymbol{\mathcal{E}}_0$ with the smallest relative errors $\|\widehat{\boldsymbol{\mathcal{E}}} - \boldsymbol{\mathcal{E}}_0\|_F / \|\boldsymbol{\mathcal{E}}_0\|_F$. The reason may be that ETRPCA explicitly considers the prominent different information between singular values. (3) TPSSV is inferior to RPCA. This is probably because that TPSSV assumes that the information, which are associated with the first r largest singular values, do not include nothing to do with content of the image. This is unreasonable in practice.

Tab. 3 shows the results of ETRPCA under different values of ω and p. In Tab. 3, we divide ω into three groups, the first group ranges from 1 to 5, the second group ranges from 6 to 10, the third group ranges from 11 to 100, and the specific parameters of each group are shown in Tab. 3. We have that ETRPCA achieves the good results with $\boldsymbol{\omega} = [0.8, 1, 10]$ when $p = 1$ is fixed,and gets the best results with $p = 0.85$ when $\boldsymbol{\omega} = [0.8, 1, 10]$ is fixed. It means that performance of ETRPCA depends on the setting of ω and p, which affect singular values. Thus, prominent different information of singular values is important in practice.

To further analyze the performance of ETRPCA, by fixing $n = 100$, $r = 10$, and $m = ||\boldsymbol{\mathcal{E}}_0||_0 = 0.1n^3$, we generate tensors $\boldsymbol{\mathcal{P}} \in \mathbb{R}^{n\times r\times n}$ and $\boldsymbol{\mathcal{Q}} \in \mathbb{R}^{r\times n\times n}$ whose elements are independently sampled from two different distributions $N(0, 1/n)$ and $N(0, 1)$, and sparse tensor $\boldsymbol{\mathcal{E}}$ ($||\boldsymbol{\mathcal{E}}_0||_0 = m$) whose non-zero entries satisfy independent Bernoulli distribution ± 1 and Laplacian distribution $La(0, 1)$, respectively. Tab. 4 lists the recovery results of ETRPCA, RPCA, TRPCA and TPSSV, respectively. It can be seen that RPCA is inferior to the tensor methods in estimating the tubal rank of $\boldsymbol{\mathcal{D}}_0$ and sparsity of $\boldsymbol{\mathcal{E}}_0$ under different distributions. Compared with the other methods, ETRPCA has the smallest relative errors about $\|\widehat{\boldsymbol{\mathcal{D}}} - \boldsymbol{\mathcal{D}}_0\|_F / \|\boldsymbol{\mathcal{D}}_0\|_F$ and $\|\widehat{\boldsymbol{\mathcal{E}}} - \boldsymbol{\mathcal{E}}_0\|_F / \|\boldsymbol{\mathcal{E}}_0\|_F$ with the exact recovery of tubal rank. These conclusions are consistent with the above analysis. All methods have the large relative errors when $\boldsymbol{\mathcal{E}}_0$ satisfies Laplacian distribution. The reason may be that sparse constraint is unreasonable for Laplacian distribution.

6.2 Phase Transition in Tubal Rank and Sparsity

In this section, we evaluate the recovery phenomenon of several methods with varying tubal rank of $\boldsymbol{\mathcal{D}}_0$ and varying sparsity of $\boldsymbol{\mathcal{E}}_0$. In the experiments, we set $n = 100$ and $n_3 = 50$, and generate a low rank tensor $\boldsymbol{\mathcal{D}}_0 = \boldsymbol{\mathcal{P}} * \boldsymbol{\mathcal{Q}}$ of tubal rank r, where $\boldsymbol{\mathcal{P}} \in \mathbf{R}^{n\times r\times n_3}$ and $\boldsymbol{\mathcal{Q}} \in \mathbf{R}^{r\times n\times n_3}$. The entries of $\boldsymbol{\mathcal{P}}$ and $\boldsymbol{\mathcal{Q}}$ are independently sampled from $N(0, 1/n)$ distribution. We also generate a sparse tensor $\boldsymbol{\mathcal{E}}_0$ which satisfies the Bernoulli distribution $B(\rho_s)$. Its elements are defined as

$$[\boldsymbol{\mathcal{E}}_0]_{ijk} = \begin{cases} 1, & w.p.\ \rho_s/2 \\ 0, & w.p.\ 1-\rho_s \\ -1, & w.p.\ \rho_s/2 \end{cases} \tag{46}$$

where, ρ_s is a probability to control sparsity, we select ρ_s and r/n as in [0.01: 0.01 : 0.5] and [0.01 : 0.01 : 0.5], respectively. For each data generated by (r, ρ_s), if the recovered tensor $\widehat{\boldsymbol{\mathcal{D}}}$ satisfies $||\widehat{\boldsymbol{\mathcal{D}}} - \boldsymbol{\mathcal{D}}_0||_F / ||\boldsymbol{\mathcal{D}}_0||_F < 1e-3$, then the recovery of low rank is successful, otherwise recovery fails. Fig. 2 shows the recovery results of several tensor methods (ETRPCA, TRPCA and TPSSV) (black and white denote 0% and 100%, respectively). Fig. 2 (a)-(d) denote the recovery results of our method with varying weight vector $\boldsymbol{\omega}$ and p. In the experiments, singular values, which are in decreasing order, are divided into three groups, the first group ranges from 1 to 50, the second group ranges from 51 to 75, the third group ranges from 76 to 100, then $\boldsymbol{\omega} = [0.9, 0.9, 1.1]$ and $p = 1$ for (a), $\boldsymbol{\omega} = [0.9, 0.9, 1.1]$ and $p = 0.98$ for (b), $\boldsymbol{\omega} =$

0162-8828 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Cornell University Library. Downloaded on August 22,2020 at 18:51:09 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2020.3017672, IEEE Transactions on Pattern Analysis and Machine Intelligence

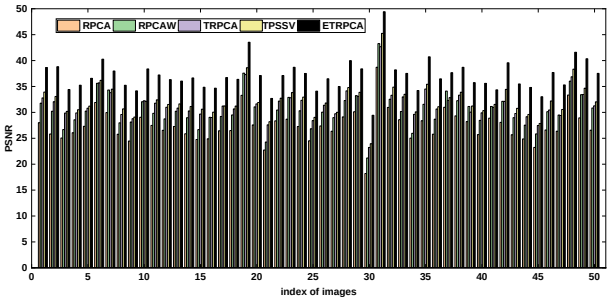

Fig. 4: Comparison of the PSNR values of RPCA, WRPCA, TRPCA, TPSSV and ETRPCA for image denoising on 50 images.

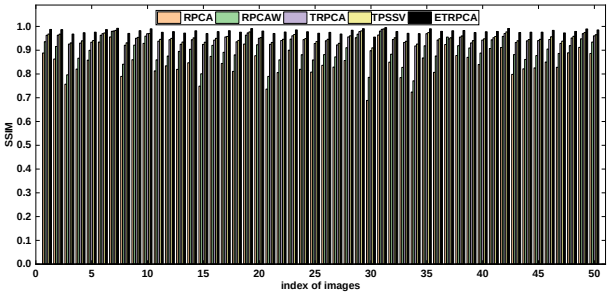

Fig. 5: Comparison of the SSIM values of RPCA, WRPCA, TRPCA, TPSSV and ETRPCA for image denoising on 50 images.

$[0.95, 0.95, 1.05]$ and $p = 0.95$ for (c), and $\boldsymbol{\omega} = [0.9, 0.9, 1.1]$ and $p = 0.95$ for (d). (e) and (f) in Fig. 2 denote TRPCA and TPSSV, respectively. From Fig. 2, we have that ETRPCA and TRPCA are superior to TPSSV. It is may be that TPSSV does not shrink the first r largest singular values which include nothing to do with the content of images. ETRPCA is superior to the other tensor methods, and has a large region in which the recovery is correct. From (a)-(d) in Fig. 2, we have that the recovery is correct when the tubal rank of $\boldsymbol{\mathcal{D}}_0$ is relatively low and the errors $\boldsymbol{\mathcal{E}}_0$ is relatively sparse. Moreover, p and $\boldsymbol{\omega}$ affect the performance of ETRPCA, it indicates that the prior information of singular values is important.

6.3 Image Recovery

We evaluate ETRPCA on the corrupted color images for the image recovery. In the experiments, we select Berkeley Segmentation dataset [22] as gallery, and randomly choose 300 color images for the test. The size of each image is $481 \times 321 \times 3$. For each image, 10% of pixels are randomly set as random values in [0, 255], and the positions of the corrupted pixels are unknown. All the 3 channels of the images are corrupted at the same positions (the corruptions are on the whole tubes). This problem is more challenging than the corruptions on 3 channels at different positions [13]. For the recovered image, we evaluate its quality by the Peak Signal-to-Noise Ratio (PSNR) value and structural similarity (SSIM) value. The higher PSNR and SSIM values indicate better recovery performance. Fig. 4 and Fig. 5 list PSNR and SSIM values of several methods on 50 images, respectively, due to the space limit. Fig. 6 shows some images and the corresponding denoised images that are recovered by our method ETRPCA, RPCA, WRPCA, TRPCA and TPSSV, respectively. We set parameter $\lambda = 1\Big/\sqrt{3\max(n_1, n_2)}$ in ETRPCA. As can be seen in Figs. 4-6, RPCA and WRPCA are inferior to the tensor methods (TRPCA, TPSSV and ETRPCA), the reason is that RPCA and WRPCA, which perform the image recovery on each channel independently, do not exploit the complementary information and high-order information embedded in different channels, while the tensor methods improve the performance by taking the advantage of the multi-dimensional structure of data. Our method ETRPCA is superior to TRPCA and TPSSV, and achieves best results with $p \neq 1$. This is probably because that each singular value has different meaning and our method considers this fact by choosing different shrink thresholds for different singular values, while TRPCA and TPSSV do not.

6.4 Application to Background Modeling

We apply ETRPCA to background modeling that aims to extract the foreground objects from the background. We select CAVIAR1 image sequences from Scene Background Initialization (SBI) database [23] to do experiments. For CAVIAR1 image sequences, which contain 610 frames with a resolution of $384 \times 256 \times 3$, we use the first 600 frames and resize each frame to $96 \times 64 \times 3$ pixels. To use RPCA and WPRCA, we reshape it to a 18432×600 matrix. To use TRPCA, TPSSV and ETRPCA, we reshape it to a $6144 \times 3 \times 600$ tensor. Fig. 3 shows some original images and the corresponding experimental results by ETRPCA, RPCA, WPRCA, TRPCA and TPSSV, respectively. As can be seen that, ETRPCA is superior to the other methods. This is probably because that ETRPCA well preserves the salient information such as color information and filters out noise by assigning different weights for all singular values.

7 Conclusion

In this paper, we study the weighted tensor Schatten p-norm minimization, and leverage it to present an ETRPCA which explicitly considers the priori information of singular values. Thus, our method well preserves the prominent information of image, especially color images. We develop an efficient algorithm, which has a good convergence, to solve ETRPCA. Extensive experimental results on several applications indicate the efficiency of our method.

Acknowledgment

The authors would like to thank the anonymous reviewers and AE for their constructive comments and suggestions.

References

[1] E. J. Candes, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *Journal of the ACM*, vol. 58, no. 3, p. 11, 2011.

[2] N. Vaswani, T. Bouwmans, S. Javed, and P. Narayanamurthy, "Robust subspace learning: Robust pca, robust subspace tracking, and robust subspace recovery," *IEEE Signal Processing Magazine*, vol. 35, no. 4, pp. 32–55, 2018.

[3] T. Bouwmans, S. Javed, H. Zhang, Z. Lin, and R. Otazo, "On the applications of robust pca in image and video processing," *Proceedings of the IEEE*, vol. 106, no. 8, pp. 1427–1457, 2018.

[4] B. Jiang, C. Ding, B. Luo, and J. Tang, "Graph-Laplacian PCA: Closed-Form Solution and Robustness," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 3492–3498.

[5] N. Shahid, V. Kalofolias, X. Bresson, M. Bronstein, and P. Vandergheynst, "Robust principal component analysis on graphs," in *IEEE Int. Conf. Comput. Vis.*, 2015, pp. 2812–2820.

0162-8828 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Cornell University Library. Downloaded on August 22,2020 at 18:51:09 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2020.3017672, IEEE Transactions on Pattern Analysis and Machine Intelligence

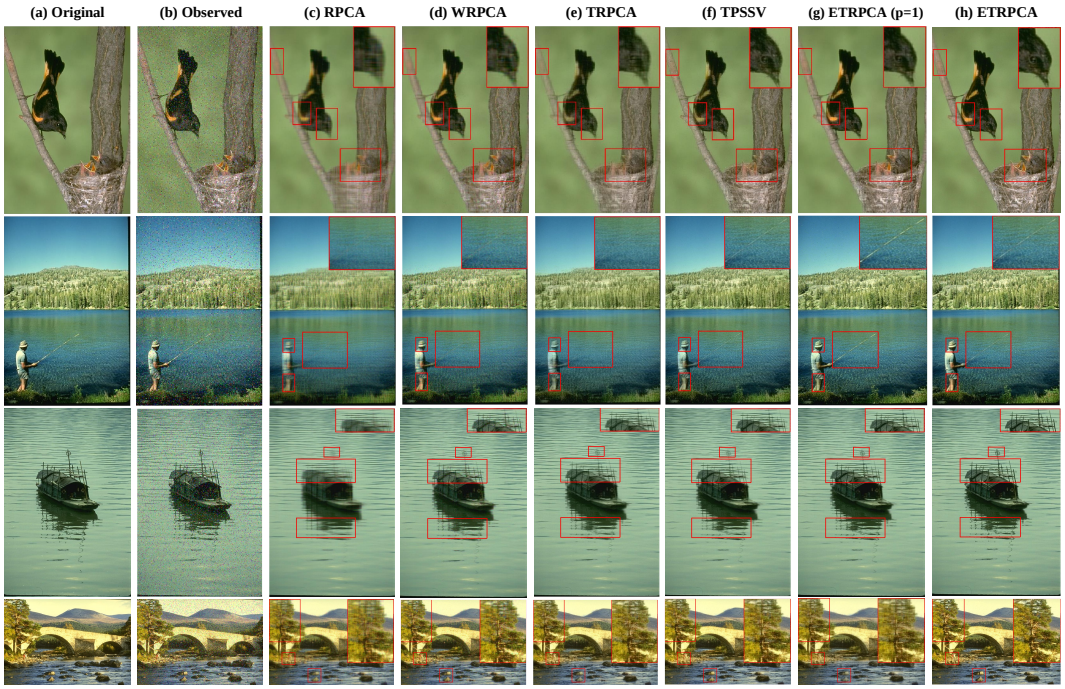

Fig. 6: Recovery performance comparison on the Berkeley images. (a) Original image; (b) observed image; (c)-(h) recovered images by RPCA, WRPCA, TRPCA, TPSSV and ETRPCA, respectively.

[6] Z. Zhou, X. Li, J. Wright, E. Candes, and Y. Ma, "Stable principal component pursuit," in *IEEE international symposium on information theory*. IEEE, 2010, pp. 1518–1522.

[7] K. Chen, H. Dong, and K.-S. Chan, "Reduced rank regression via adaptive nuclear norm penalization," *Biometrika*, vol. 100, no. 4, pp. 901–920, 2013.

[8] S. Gu, Q. Xie, D. Meng, W. Zuo, X. Feng, and L. Zhang, "Weighted nuclear norm minimization and its applications to low level vision," *Int. J. Comput. Vis.*, vol. 121, no. 2, pp. 183–208, 2017.

[9] T. H. Oh, Y. W. Tai, J. C. Bazin, H. Kim, and I. S. Kweon, "Partial sum minimization of singular values in robust pca: Algorithm and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 4, pp. 744–758, 2016.

[10] Y. Hu, D. Zhang, J. Ye, X. Li, and X. He, "Fast and accurate matrix completion via truncated nuclear norm regularization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 9, pp. 2117–2130, 2013.

[11] Q. Gao, W. Xia, Z. Wan, D. Xie, and P. Zhang, "Tensor-svd based graph learning for multi-view subspace clustering," in *The Thirty-Fourth AAAI Conference on Artificial Intelligence, 2020, New York, USA, February 7-12, 2020*, 2020, pp. 3930–3937.

[12] Y. Nu, P. Wang, L. Lu, X. Zhang, and L. Qi, "Weighted tensor nuclear norm minimization for tensor completion using tensor-svd," *Pattern Recognit. Lett.*, vol. 130, pp. 4–11, 2020.

[13] C. Lu, J. Feng, W. Liu, Z. Lin, and S. Yan, "Tensor robust principal component analysis with a new tensor nuclear norm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 4, pp. 925–938, 2020.

[14] L. Zhang and Z. Peng, "Infrared small target detection based on partial sum of the tensor nuclear norm," *Remote Sensing*, vol. 11, no. 4, pp. 382–392, 2019.

[15] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, 2002.

[16] M. E. Kilmer and C. D. Martin, "Factorization strategies for third-order tensors," *Linear Algebra and Its Applications*, vol. 435, no. 3, pp. 641–658, 2011.

[17] C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, and S. Yan, "Tensor robust principal component analysis: Exact recovery of corrupted low-rank tensors via convex optimization," in *IEEE Conf. Comput. Vis. Pattern Recognit.,*, 2016, pp. 5249–5257.

[18] Y. Xie, S. Gu, Y. Liu, W. Zuo, W. Zhang, and L. Zhang, "Weighted schatten p-norm minimization for image denoising and background subtraction," *IEEE Trans. Image Processing*, vol. 25, no. 10, pp. 4842–4857, 2016.

[19] L. Mirsky, "A trace inequality of john von neumann," *Monatshefte für mathematik*, vol. 79, no. 4, pp. 303–306, 1975.

[20] Z. Lin, R. Liu, and H. Li, "Linearized alternating direction method with parallel splitting and adaptive penalty for separable convex programs in machine learning," *Machine Learning*, vol. 99, no. 2, pp. 287–325, 2015.

[21] J. E. Kimber and Jr., "Two extended bolzano-weierstrass theorems," *American Mathematical Monthly*, vol. 72, no. 9, pp. 1007–1012, 1965.

[22] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *IEEE Int. Conf. Comput. Vis.,*, 2001, pp. 416–423.

[23] L. Maddalena and A. Petrosino, "Towards benchmarking scene background initialization," in *ICIAP*, 2015, pp. 469–476.

0162-8828 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Cornell University Library. Downloaded on August 22,2020 at 18:51:09 UTC from IEEE Xplore. Restrictions apply.