Transforms based Tensor Robust PCA: Corrupted Low-Rank Tensors Recovery via Convex Optimization

Canyi Lu
Department of Electrical and Computer Engineering, Carnegie Mellon University
canyilu@gmail.com

Abstract

This work studies the Tensor Robust Principal Component Analysis (TRPCA) problem, which aims to exactly recover the low-rank and sparse components from their sum. Our model is motivated by the recently proposed linear transforms based tensor-tensor product and tensor SVD. We define a new transforms depended tensor rank and the corresponding tensor nuclear norm. Then we solve the TRPCA problem by convex optimization whose objective is a weighted combination of the new tensor nuclear norm and ℓ_1-norm. In theory, we prove that under some incoherence conditions, the convex program exactly recovers the underlying low-rank and sparse components with high probability. Our new TRPCA is much more general since it allows to use any invertible linear transforms. Thus, we have more choices in practice for different tasks and different type of data. Numerical experiments verify our results and the application on image recovery demonstrates the superiority of our method.

1. Introduction

Tensors are the higher-order generalization of vectors and matrices. They have many applications in many areas, and an in depth survey can be found in [12]. Tensor decompositions give a concise representation of the underlying structure of data, revealing the low-dimensional subspace of data. These decompositions are now widely used in many application areas such as computer vision [26], web data mining [6], and signal processing [23].

Tensor decomposition faces several challenges: arbitrary outliers, missing data/partial observations, and computational efficiency. Tensor decomposition resembles Principal Component Analysis (PCA) for matrices in many ways. The two commonly used decompositions are the CP and Tucker decomposition [12]. It is known that PCA is sensitive to outliers and gross corruptions, since the CP and Tucker decompositions are also based on least-squares approximation. Algorithms based on nonconvex formulations have been proposed to robustify tensor decompositions [5, 1]. However, they suffer from the lack of global optimality guarantee and statistical guarantee.

This work studies the Tensor Robust PCA (TRPCA) problem by convex optimization. Assume that a tensor $\boldsymbol{\mathcal{X}}$ can be decomposed as $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{L}}_0 + \boldsymbol{\mathcal{E}}_0$, where $\boldsymbol{\mathcal{L}}_0$ is low-rank and $\boldsymbol{\mathcal{E}}_0$ is sparse. TRPCA aims to recover $\boldsymbol{\mathcal{L}}_0$ and $\boldsymbol{\mathcal{E}}_0$ from $\boldsymbol{\mathcal{X}}$. We focus on the convex model which can be solved exactly and efficiently, and the solutions own the theoretical guarantee. TRPCA extends the known Robust PCA [3]

$$\min_{\boldsymbol{L},\boldsymbol{E}} \|\boldsymbol{L}\|_* + \lambda\|\boldsymbol{E}\|_1, \text{ s.t. } \boldsymbol{X} = \boldsymbol{L} + \boldsymbol{E}, \tag{1}$$

where $\|\boldsymbol{L}\|_*$ denotes the matrix nuclear norm, $\|\boldsymbol{E}\|_1$ denotes the ℓ_1-norm and $\lambda > 0$. It is proved that the solution to (1) exactly recovers the underlying low-rank and sparse components [3]. RPCA has many applications in image and video analysis [9, 21].

It is natural to consider the tensor extension of RPCA. However, existed tensor models have several limitations in theory or computation. The main issues lie in the definitions of tensor rank. There have different tensor SVD decompositions which lead to different tensor rank definitions. Then they lead to different tensor RPCA models. Compared with the matrix RPCA, the tensor RPCA models have several limitations. The tensor CP rank is defined as the smallest number of rank one tensors decomposition. However, the CP rank and its best convex relaxation are NP-hard to compute [7]. These issues make the low CP rank tensor recovery problem challenging. The tensor Tucker rank is more widely studied as it is tractable. The sum-of-nuclear-norms (SNN) is served as a simple convex surrogate for tensor Tucker rank. This idea has been successfully applied to various problems [13, 24]. The first theoretical guarantee for SNN minimization is given in [25]. It has been further enhanced in [20, 8]. However, these results are still suboptimal since SNN is not a tight convex relaxation of the Tucker rank. For exact tensor completion, the required sample complexity of SNN is much higher than the degrees

of freedom. This is different from the matrix nuclear norm minimization which leads to order optimal sampling complexity [4]. Intuitively, the limitation of SNN is that it is not a tight convex relaxation of the Tucker rank.

Recently, based on the tensor-tensor product (t-product) [11] which generalizes the matrix-matrix product, a new tensor tubal rank and Tensor Nuclear Norm (TNN) have been proposed and applied to tensor completion [17, 22, 18] and tensor RPCA [15, 16]. Compared with the Tucker rank based SNN model, the main advantage of the t-product induced TNN models is that they own the same tight recovery bound as the matrix cases. Also, unlike the CP rank, the tubal rank and TNN are computable.

Motivated by a more general t-product definition performed on any invertible linear transforms [10], we propose a more general TRPCA model induced by the new t-product. We show that when the invertible linear transform given by the matrix $\boldsymbol{L}$ further satisfies

$$\boldsymbol{L}^\top \boldsymbol{L} = \boldsymbol{L}\boldsymbol{L}^\top = \ell \boldsymbol{I},$$

for some constant $\ell > 0$, we can define a new transform based tensor nuclear norm. Equipped with it, we then can solve the TRPCA problem by convex optimization

$$\min_{\boldsymbol{\mathcal{L}},\, \boldsymbol{\mathcal{E}}} \|\boldsymbol{\mathcal{L}}\|_* + \lambda \|\boldsymbol{\mathcal{S}}\|_1, \text{ s.t. } \boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{L}} + \boldsymbol{\mathcal{S}}. \tag{2}$$

Above, the tensor nuclear norm is defined in Section 3. In theory, we prove that, under certain incoherence conditions, the solution to (2) exactly recovers the underlying low-rank $\boldsymbol{\mathcal{L}}_0$ and sparse $\boldsymbol{\mathcal{S}}_0$ components with high probability. Different from the TRPCA in [16] which uses the discrete Fourier transform, our model is much more flexible as it is allowed to use any invertible linear transforms. Note that our TRPCA does not reduces to the model in [16] since we restrict $\boldsymbol{L}$ in $\mathbb{R}^{n_3 \times n_3}$ while the discrete Fourier transform matrix is complex. This also leads to several key differences in the theoretical proofs from [16].

The rest of this paper is structured as follows. Section 2 gives some notations and presents the new tensor nuclear norm induced by the t-product under linear transforms. Section 3 introduces the tensor nuclear norm induced by the t-product under linear transforms. Section 4 presents the theoretical guarantee of the new tensor nuclear norm based convex TRPCA model. Numerical experiments conducted on both synthesis and real world data are presented in Section 5. We finally conclude this work in Section 6.

2. Transforms based T-product

2.1. Notations

We introduce some notations and definitions used in [16]. We denote scalars by lowercase letters, e.g., a, vector by boldface lowercase letters, e.g., $\boldsymbol{a}$, matrices by boldface capital letters, e.g., $\boldsymbol{A}$, and tensors by boldface Euler script letters, e.g., $\boldsymbol{\mathcal{A}}$. For a 3-way tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, we denote its (i,j,k)-th entry as $\boldsymbol{\mathcal{A}}_{ijk}$ or a_{ijk} and use the Matlab notation $\boldsymbol{\mathcal{A}}(i,:,:)$, $\boldsymbol{\mathcal{A}}(:,i,:)$ and $\boldsymbol{\mathcal{A}}(:,:,i)$ to denote respectively the i-th horizontal, lateral and frontal slice [12]. More often, the frontal slice $\boldsymbol{\mathcal{A}}(:,:,i)$ is denoted compactly as $\boldsymbol{A}^{(i)}$. The tube is denoted as $\boldsymbol{\mathcal{A}}(i,j,:)$. The inner product between $\boldsymbol{A}$ and $\boldsymbol{B}$ in $\mathbb{R}^{n_1 \times n_2}$ is defined as $\langle \boldsymbol{A}, \boldsymbol{B} \rangle = \mathtt{Tr}(\boldsymbol{A}^\top \boldsymbol{B})$, where $\boldsymbol{A}^\top$ denotes the transpose and $\mathtt{Tr}(\cdot)$ denotes the matrix trace. The inner product between $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$ is defined as $\langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}} \rangle = \sum_{i=1}^{n_3} \left\langle \boldsymbol{A}^{(i)}, \boldsymbol{B}^{(i)} \right\rangle$. We denote $\boldsymbol{I}_n$ as the $n \times n$ sized identity matrix. We denote the Frobenius norm as $\|\boldsymbol{\mathcal{A}}\|_F = \sqrt{\sum_{ijk} a_{ijk}^2}$, the ℓ_1-norm as $\|\boldsymbol{\mathcal{A}}\|_1 = \sum_{ijk} |a_{ijk}|$, and the infinity norm as $\|\boldsymbol{\mathcal{A}}\|_\infty = \max_{ijk} |a_{ijk}|$, respectively. The above norms reduce to the vector or matrix norms if $\boldsymbol{\mathcal{A}}$ is a vector or a matrix. For $\boldsymbol{v} \in \mathbb{R}^n$, the ℓ_2-norm is $\|\boldsymbol{v}\|_2 = \sqrt{\sum_i v_i^2}$. The spectral norm of $\boldsymbol{A}$ is $\|\boldsymbol{A}\| = \max_i \sigma_i(\boldsymbol{A})$, where $\sigma_i(\boldsymbol{A})$'s are the singular values. The matrix nuclear norm is $\|\boldsymbol{A}\|_* = \sum_i \sigma_i(\boldsymbol{A})$.

2.2. T-product under Linear Transform

The t-product is defined based on the block circulant matrix and unfolded tensor [11]. It is equivalent to the matrix-matrix product under the discrete Fourier transform. We denote $\boldsymbol{\mathcal{R}} = \boldsymbol{\mathcal{P}} \odot \boldsymbol{\mathcal{Q}}$ as the frontal-slice-wise product (Definition 2.1 in [10]), i.e., $\boldsymbol{R}^{(i)} = \boldsymbol{P}^{(i)} \boldsymbol{Q}^{(i)}$, $i = 1, \cdots, n_3$. Let

$$\bar{\boldsymbol{\mathcal{A}}} = \boldsymbol{\mathcal{A}} \times_3 \boldsymbol{F}_{n_3}, \tag{3}$$

where $\times_3$ denotes the mode-3 product (see Definition 2.5 in [10]) and $\boldsymbol{F}_{n_3}$ is the Discrete Fourier Transform (DFT) matrix. We denote $\bar{\boldsymbol{A}} \in \mathbb{C}^{n_1 n_3 \times n_2 n_3}$ as

$$\bar{\boldsymbol{A}} = \mathtt{bdiag}(\bar{\boldsymbol{\mathcal{A}}}) = \begin{bmatrix} \bar{\boldsymbol{A}}^{(1)} & & & \\ & \bar{\boldsymbol{A}}^{(2)} & & \\ & & \ddots & \\ & & & \bar{\boldsymbol{A}}^{(n_3)} \end{bmatrix},$$

where $\mathtt{bdiag}$ is an operator which maps $\bar{\boldsymbol{\mathcal{A}}}$ to $\bar{\boldsymbol{A}}$. 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 [11] can be equivalently defined as

$$\boldsymbol{\mathcal{C}} = \boldsymbol{\mathcal{A}} * \boldsymbol{\mathcal{B}} \Leftrightarrow \bar{\boldsymbol{\mathcal{C}}} = \bar{\boldsymbol{\mathcal{A}}} \odot \bar{\boldsymbol{\mathcal{B}}} \Leftrightarrow \bar{\boldsymbol{C}} = \bar{\boldsymbol{A}} \bar{\boldsymbol{B}}. \tag{4}$$

Instead of using the specific discrete Fourier transform, a more general definition of t-product is proposed based on any invertible linear transform L [10]. In this work, we consider the linear transform $L : \mathbb{R}^{n_1 \times n_2 \times n_3} \to \mathbb{R}^{n_1 \times n_2 \times n_3}$ which gives $\bar{\boldsymbol{\mathcal{A}}}$ by performing a linear transform on $\boldsymbol{\mathcal{A}}$ along the 3-rd dimension, i.e.,

$$\bar{\boldsymbol{\mathcal{A}}} = L(\boldsymbol{\mathcal{A}}) = \boldsymbol{\mathcal{A}} \times_3 \boldsymbol{L}, \tag{5}$$

where the linear transform L is given by $\boldsymbol{L} \in \mathbb{R}^{n_3 \times n_3}$ which is invertible. We also have the inverse mapping given by

$$L^{-1}(\boldsymbol{\mathcal{A}}) = \boldsymbol{\mathcal{A}} \times_3 \boldsymbol{L}^{-1}. \tag{6}$$

Note that we restrict $\boldsymbol{L}$ to be a real matrix in this work for two reasons. First, in practice, the real linear transform guarantees that the t-product of two real tensors is also real. Thus, the solution/output of the TRPCA model (2) is real. This is important for real applications. Second, in theory, the proofs of the exact recovery guarantee in Section 4.2 is not applicable to the complex linear transforms. It will affect the definitions of the tensor basis. See more detailed discussions after Theorem 2.

We have the following t-product definition and other related concepts under linear transform L.

Definition 1 ***(T-product)*** *[10] Let L be any invertible linear transform in (5), $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ and $\boldsymbol{\mathcal{B}} \in \mathbb{R}^{n_2 \times l \times n_3}$. The t-product of $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$ under L, denoted as $\boldsymbol{\mathcal{C}} = \boldsymbol{\mathcal{A}} *_L \boldsymbol{\mathcal{B}}$, is defined such that $L(\boldsymbol{\mathcal{C}}) = L(\boldsymbol{\mathcal{A}}) \odot L(\boldsymbol{\mathcal{B}})$.*

For any invertible linear transform L in (5), we have the following tensor concepts extended from the matrix cases.

Definition 2 ***(Tensor transpose)*** *[10] Let L be any invertible linear transform in (5), and $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$. Then the tensor transpose of $\boldsymbol{\mathcal{A}}$ under L, denoted as $\boldsymbol{\mathcal{A}}^\top$, satisfies $L(\boldsymbol{\mathcal{A}}^\top)^{(i)} = (L(\boldsymbol{\mathcal{A}})^{(i)})^\top$, $i = 1, \cdots, n_3$.*

Definition 3 ***(Identity tensor)*** *[10] Let L be any invertible linear transform in (5). Let $\boldsymbol{\mathcal{I}} \in \mathbb{R}^{n \times n \times n_3}$ so that each frontal slice of $L(\boldsymbol{\mathcal{I}}) = \bar{\boldsymbol{\mathcal{I}}}$ is a $n \times n$ sized identity matrix. Then $\boldsymbol{\mathcal{I}} = L^{-1}(\bar{\boldsymbol{\mathcal{I}}})$ gives the identity tensor under L.*

It is clear that $\boldsymbol{\mathcal{A}} *_L \boldsymbol{\mathcal{I}} = \boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{I}} *_L \boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{A}}$ with appropriate dimensions. The tensor $\bar{\boldsymbol{\mathcal{I}}} = L(\boldsymbol{\mathcal{I}})$ is a tensor with each frontal slice being the identity matrix.

Definition 4 ***(Orthogonal tensor)*** *[10] Let L be any invertible linear transform in (5). A tensor $\boldsymbol{\mathcal{Q}} \in \mathbb{R}^{n \times n \times n_3}$ is orthogonal under L if it satisfies $\boldsymbol{\mathcal{Q}}^\top *_L \boldsymbol{\mathcal{Q}} = \boldsymbol{\mathcal{Q}} *_L \boldsymbol{\mathcal{Q}}^\top = \boldsymbol{\mathcal{I}}$.*

Definition 5 ***(F-diagonal Tensor)*** *A tensor is called f-diagonal if each of its frontal slices is a diagonal matrix.*

Theorem 1 ***(T-SVD)*** *[10] Let L be any invertible linear transform in (5), and $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$. Then it can be factorized as*

$$\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{U}} *_L \boldsymbol{\mathcal{S}} *_L \boldsymbol{\mathcal{V}}^\top, \tag{7}$$

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

Definition 6 ***(Tensor tubal rank)*** *Let L be any invertible linear transform in (5), and $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$. The tensor tubal rank of $\boldsymbol{\mathcal{A}}$ under L, denoted as $\text{rank}_t(\boldsymbol{\mathcal{A}})$, is defined as the number of nonzero tubes of $\boldsymbol{\mathcal{S}}$, where $\boldsymbol{\mathcal{S}}$ is from the t-SVD of $\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{U}} *_L \boldsymbol{\mathcal{S}} *_L \boldsymbol{\mathcal{V}}^\top$. We can write*

$$\text{rank}_t(\boldsymbol{\mathcal{A}}) = \#\{i, \boldsymbol{\mathcal{S}}(i,i,:) \neq 0\}.$$

For $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ with tubal rank r, we also have the skinny t-SVD, i.e., $\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{U}} *_L \boldsymbol{\mathcal{S}} *_L \boldsymbol{\mathcal{V}}^\top$, where $\boldsymbol{\mathcal{U}} \in \mathbb{R}^{n_1 \times r \times n_3}$, $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{r \times r \times n_3}$, and $\boldsymbol{\mathcal{V}} \in \mathbb{R}^{n_2 \times r \times n_3}$, in which $\boldsymbol{\mathcal{U}}^\top *_L \boldsymbol{\mathcal{U}} = \boldsymbol{\mathcal{I}}$ and $\boldsymbol{\mathcal{V}}^\top *_L \boldsymbol{\mathcal{V}} = \boldsymbol{\mathcal{I}}$. We use the skinny t-SVD throughout this paper.

3. Transform based Tensor Nuclear Norm

At the following, we show how to define the convex tensor nuclear norm induced by the t-product under L. We can first define the tensor spectral norm as in [15, 16].

Definition 7 ***(Tensor spectral norm)*** *Let L be any invertible linear transform in (5), and $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$. The tensor spectral norm of $\boldsymbol{\mathcal{A}}$ under L is defined as $\|\boldsymbol{\mathcal{A}}\| := \|\bar{\boldsymbol{A}}\|$.*

Then the tensor nuclear norm can be defined as the dual norm of the tensor spectral norm. To this end, we further need the following assumption on L given in (5), i.e.,

$$\boldsymbol{L}^\top \boldsymbol{L} = \boldsymbol{L}\boldsymbol{L}^\top = \ell \boldsymbol{I}_{n_3}, \tag{8}$$

where $\ell > 0$ is a constant. Using (8), we have

$$\langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}} \rangle = \frac{1}{\ell} \langle \bar{\boldsymbol{A}}, \bar{\boldsymbol{B}} \rangle. \tag{9}$$

$$\|\boldsymbol{\mathcal{A}}\|_F = \frac{1}{\sqrt{\ell}} \|\bar{\boldsymbol{A}}\|_F. \tag{10}$$

For any $\boldsymbol{\mathcal{B}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ and $\tilde{\boldsymbol{B}} \in \mathbb{R}^{n_1 n_3 \times n_2 n_3}$, we have

$$\|\boldsymbol{\mathcal{A}}\|_* := \sup_{\|\boldsymbol{\mathcal{B}}\| \leq 1} \langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}} \rangle = \sup_{\|\bar{\boldsymbol{B}}\| \leq 1} \frac{1}{\ell} \langle \bar{\boldsymbol{A}}, \bar{\boldsymbol{B}} \rangle \tag{11}$$

$$\leq \frac{1}{\ell} \sup_{\|\tilde{\boldsymbol{B}}\| \leq 1} \langle \bar{\boldsymbol{A}}, \tilde{\boldsymbol{B}} \rangle = \frac{1}{\ell} \|\bar{\boldsymbol{A}}\|_*, \tag{12}$$

where (11) uses (9), the inequality uses the fact that $\bar{\boldsymbol{B}}$ is a block diagonal matrix in $\mathbb{R}^{n_1 n_3 \times n_2 n_3}$ while $\tilde{\boldsymbol{B}}$ is an arbitrary matrix in $\mathbb{R}^{n_1 n_3 \times n_2 n_3}$, and (12) uses the fact that the matrix nuclear norm is the dual norm of the matrix spectral norm. On the other hand, let $\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{U}} *_L \boldsymbol{\mathcal{S}} *_L \boldsymbol{\mathcal{V}}^\top$ be the t-SVD of $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}} = \boldsymbol{\mathcal{U}} *_L \boldsymbol{\mathcal{V}}^\top$. We have

$$\begin{aligned} \|\boldsymbol{\mathcal{A}}\|_* &= \sup_{\|\boldsymbol{\mathcal{B}}\| \leq 1} \langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}} \rangle \geq \langle \boldsymbol{\mathcal{U}} *_L \boldsymbol{\mathcal{S}} *_L \boldsymbol{\mathcal{V}}^\top, \boldsymbol{\mathcal{U}} *_L \boldsymbol{\mathcal{V}}^\top \rangle \\ &= \langle \boldsymbol{\mathcal{U}}^\top *_L \boldsymbol{\mathcal{U}} *_L \boldsymbol{\mathcal{S}}, \boldsymbol{\mathcal{V}}^\top *_L \boldsymbol{\mathcal{V}} \rangle \\ &= \langle \boldsymbol{\mathcal{S}}, \boldsymbol{\mathcal{I}} \rangle = \frac{1}{\ell} \langle \bar{\boldsymbol{S}}, \bar{\boldsymbol{I}} \rangle = \frac{1}{\ell} \text{Tr}(\bar{\boldsymbol{S}}) = \frac{1}{\ell} \|\bar{\boldsymbol{A}}\|_*. \end{aligned} \tag{13}$$

Combining (11)-(12) and (13), we then have the following definition of tensor nuclear norm.

Definition 8 ***(Tensor nuclear norm)*** *Let L be any invertible linear transform in (5) and it satisfies (8), and $\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{U}} *_L \boldsymbol{\mathcal{S}} *_L \boldsymbol{\mathcal{V}}^\top$ be the t-SVD of $\boldsymbol{\mathcal{A}}$. The tensor nuclear norm of $\boldsymbol{\mathcal{A}}$ under L is defined as $\|\boldsymbol{\mathcal{A}}\|_* := \langle \boldsymbol{\mathcal{S}}, \boldsymbol{\mathcal{I}} \rangle = \frac{1}{\ell}\|\bar{\boldsymbol{A}}\|_*$.*

We would like to emphasize that our condition for the linear transform L in (8) is new for defining the tensor nuclear norm. In contrast, t-product only requires L to be invertible.

4. TRPCA under Linear Transform with Exact Recovery Guarantee

We can solve the convex TRPCA model in (2) by Alternating Direction Method of Multipliers [2] (see details in the supplementary material). The per-iteration cost is $O(n_1 n_2 n_3^2 + n_{(1)} n_{(2)}^2 n_3)$. Assume that we are given $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{L}}_0 + \boldsymbol{\mathcal{S}}_0$, where $\boldsymbol{\mathcal{L}}_0$ is of low tubal rank and $\boldsymbol{\mathcal{S}}_0$ is sparse. In this section, we will theoretically show that the solution to (2) exactly recovers both $\boldsymbol{\mathcal{L}}_0$ and $\boldsymbol{\mathcal{S}}_0$ with overwhelming probability.

4.1. Tensor Incoherence Conditions

Some incoherence conditions are required to avoid some pathological situations that the recovery is impossible. We need to assume that the low-rank component $\boldsymbol{\mathcal{L}}_0$ is not sparse. To this end, we assume $\boldsymbol{\mathcal{L}}_0$ to satisfy some incoherence conditions. Another identifiability issue arises if the sparse tensor has low tubal rank. This can be avoided by assuming that the support of $\boldsymbol{\mathcal{S}}_0$ is uniformly distributed. We need the following tensor basis concept for defining the tensor incoherence conditions.

Definition 9 ***(Standard tensor basis)*** *Let L be any invertible linear transform in (5) and it satisfies (8). We denote $\mathring{\mathbf{e}}_i$ as the tensor **column basis**, which is a tensor of size $n \times 1 \times n_3$ with the entries of the $(i,1,:)$ tube of $L(\mathring{\mathbf{e}}_i)$ equaling 1 and the rest equaling 0. Naturally its transpose $\mathring{\mathbf{e}}_i^\top$ is called **row basis**. The other tensor basis is called **tube basis** $\dot{\mathbf{e}}_k$, which is a tensor of size $1 \times 1 \times n_3$ with the $(1,1,k)$-th entry of $L(\dot{\mathbf{e}}_k)$ equaling 1 and the rest equaling 0.*

Definition 10 ***(Tensor incoherence conditions)*** *Let L be any invertible linear transform in (5) and it satisfies (8). For $\boldsymbol{\mathcal{L}}_0 \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ with $\text{rank}_t(\boldsymbol{\mathcal{L}}_0) = r$ and the skinny t-SVD $\boldsymbol{\mathcal{L}}_0 = \boldsymbol{\mathcal{U}} *_L \boldsymbol{\mathcal{S}} *_L \boldsymbol{\mathcal{V}}^\top$. Then $\boldsymbol{\mathcal{L}}_0$ is said to satisfy the tensor incoherence conditions with parameter μ if*

$$\max_{i=1,\cdots,n_1} \max_{k=1,\cdots,n_3} \|\boldsymbol{\mathcal{U}}^\top *_L \mathring{\mathbf{e}}_i *_L L(\dot{\mathbf{e}}_k)\|_F \le \sqrt{\frac{\mu r}{n_1}}, \quad (14)$$

$$\max_{j=1,\cdots,n_2} \max_{k=1,\cdots,n_3} \|\boldsymbol{\mathcal{V}}^\top *_L \mathring{\mathbf{e}}_j *_L L(\dot{\mathbf{e}}_k)\|_F \le \sqrt{\frac{\mu r}{n_2}}, \quad (15)$$

and

$$\|\boldsymbol{\mathcal{U}} *_L \boldsymbol{\mathcal{V}}^\top\|_\infty \le \sqrt{\frac{\mu r}{n_1 n_2 \ell}}. \quad (16)$$

Proposition 1 *With the same notations in Definition 10, if the following conditions hold,*

$$\max_{i=1,\cdots,n_1} \|\boldsymbol{\mathcal{U}}^\top *_L \mathring{\mathbf{e}}_i\|_F \le \sqrt{\frac{\mu r}{n_1 \ell}}, \quad (17)$$

$$\max_{j=1,\cdots,n_2} \|\boldsymbol{\mathcal{V}}^\top *_L \mathring{\mathbf{e}}_j\|_F \le \sqrt{\frac{\mu r}{n_2 \ell}}, \quad (18)$$

then (14)-(15) hold.

Proposition 1 shows that our new conditions (14)-(15) are less restrictive than (17)-(18). And our proofs of main results only need to use the new conditions (14)-(15).

4.2. Main Results

Define $n_{(1)} = \max(n_1, n_2)$ and $n_{(2)} = \min(n_1, n_2)$. We have the following exact recovery guarantee for convex program (2).

Theorem 2 *Let L be any invertible linear transform in (5) and it satisfies (8). Suppose $\boldsymbol{\mathcal{L}}_0 \in \mathbb{R}^{n \times n \times n_3}$ obeys (14)-(16). Fix any $n \times n \times n_3$ tensor $\boldsymbol{\mathcal{M}}$ of signs. Suppose that the support set $\boldsymbol{\Omega}$ of $\boldsymbol{\mathcal{S}}_0$ is uniformly distributed among all sets of cardinality m, and that $\text{sgn}([\boldsymbol{\mathcal{S}}_0]_{ijk}) = [\boldsymbol{\mathcal{M}}]_{ijk}$ for all $(i,j,k) \in \boldsymbol{\Omega}$. Then, there exist universal constants $c_1, c_2 > 0$ such that with probability at least $1 - c_1(nn_3)^{-c_2}$ (over the choice of support of $\boldsymbol{\mathcal{S}}_0$), $\{\boldsymbol{\mathcal{L}}_0, \boldsymbol{\mathcal{S}}_0\}$ is the unique minimizer to (2) with $\lambda = 1/\sqrt{n\ell}$, provided that*

$$\text{rank}_t(\boldsymbol{\mathcal{L}}_0) \le \frac{\rho_r n}{\mu(\log(nn_3))^2} \text{ and } m \le \rho_s n^2 n_3, \quad (19)$$

where ρ_r and ρ_s are positive constants. If $\boldsymbol{\mathcal{L}}_0 \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ has rectangular frontal slices, TRPCA with $\lambda = 1/\sqrt{n_{(1)}\ell}$ succeeds with probability at least $1 - c_1(n_{(1)}n_3)^{-c_2}$, provided that $\text{rank}_t(\boldsymbol{\mathcal{L}}_0) \le \frac{\rho_r n_{(2)}}{\mu(\log(n_{(1)}n_3))^2}$ and $m \le \rho_s n_1 n_2 n_3$.

Theorem 2 gives the exact recovery guarantee for convex model (2) under certain incoherence conditions. It says that the incoherent $\boldsymbol{\mathcal{L}}_0$ can be recovered for $\text{rank}_t(\boldsymbol{\mathcal{L}}_0)$ on the order of $n/(\mu(\log nn_3)^2)$ and a number of nonzero entries in $\boldsymbol{\mathcal{S}}_0$ on the order of $n^2 n_3$. For $\boldsymbol{\mathcal{S}}_0$, we make only an assumption on the random location distribution, but no assumption about the magnitudes or signs of the nonzero entries.

We would like to emphasize that TRPCA in [16] which uses the discrete Fourier transform cannot be regarded a special case of ours, especially from the perspective of the theoretical result and its proofs. The key difference is that we restrict the linear transform within the real domain $L : \mathbb{R}^{n_1 \times n_2 \times n_3} \to \mathbb{R}^{n_1 \times n_2 \times n_3}$ while the discrete Fourier transform used in [16] is a mapping from the real domain to the complex domain $L : \mathbb{R}^{n_1 \times n_2 \times n_3} \to \mathbb{C}^{n_1 \times n_2 \times n_3}$. Note that this domain difference is crucial in the proofs. The standard tensor basis and tensor incoherence conditions are defined depended on the properties of L. Our proofs are not

Table 1: Correct recovery for random problems of varying size. The Discrete Cosine Transform (DCT) is used as the invertible linear transform L.

$r = \text{rank}_t(\boldsymbol{\mathcal{L}}_0) = 0.1n, m = \|\boldsymbol{\mathcal{S}}_0\|_0 = 0.1n^3$

n	r	m	$\text{rank}_t(\hat{\boldsymbol{\mathcal{L}}})$	$\|\hat{\boldsymbol{\mathcal{S}}}\|_0$	$\frac{\|\hat{\boldsymbol{\mathcal{L}}}-\boldsymbol{\mathcal{L}}_0\|_F}{\|\boldsymbol{\mathcal{L}}_0\|_F}$	$\frac{\|\hat{\boldsymbol{\mathcal{S}}}-\boldsymbol{\mathcal{S}}_0\|_F}{\|\boldsymbol{\mathcal{S}}_0\|_F}$
100	10	1e5	10	102,921	2.4e−6	8.7e−10
200	20	8e5	20	833,088	6.8e−6	8.7e−10
300	30	27e5	30	2,753,084	1.8e−5	1.4e−9

$r = \text{rank}_t(\boldsymbol{\mathcal{L}}_0) = 0.1n, m = \|\boldsymbol{\mathcal{S}}_0\|_0 = 0.2n^3$

n	r	m	$\text{rank}_t(\hat{\boldsymbol{\mathcal{L}}})$	$\|\hat{\boldsymbol{\mathcal{S}}}\|_0$	$\frac{\|\hat{\boldsymbol{\mathcal{L}}}-\boldsymbol{\mathcal{L}}_0\|_F}{\|\boldsymbol{\mathcal{L}}_0\|_F}$	$\frac{\|\hat{\boldsymbol{\mathcal{S}}}-\boldsymbol{\mathcal{S}}_0\|_F}{\|\boldsymbol{\mathcal{S}}_0\|_F}$
100	10	2e5	10	201,090	4.0e−6	2.1e−10
200	20	16e5	20	1,600,491	5.3e−6	9.8e−10
300	30	54e5	30	5,460,221	1.8e−5	1.8e−9

Table 2: Correct recovery for random problems of varying size. A Random Orthogonal Matrix (ROM) is used as the invertible linear transform L.

$r = \text{rank}_t(\boldsymbol{\mathcal{L}}_0) = 0.1n, m = \|\boldsymbol{\mathcal{S}}_0\|_0 = 0.1n^3$

n	r	m	$\text{rank}_t(\hat{\boldsymbol{\mathcal{L}}})$	$\|\hat{\boldsymbol{\mathcal{S}}}\|_0$	$\frac{\|\hat{\boldsymbol{\mathcal{L}}}-\boldsymbol{\mathcal{L}}_0\|_F}{\|\boldsymbol{\mathcal{L}}_0\|_F}$	$\frac{\|\hat{\boldsymbol{\mathcal{S}}}-\boldsymbol{\mathcal{S}}_0\|_F}{\|\boldsymbol{\mathcal{S}}_0\|_F}$
100	10	1e5	10	103,034	2.7e−6	9.6e−9
200	20	8e5	20	833,601	7.0e−6	9.0e−10
300	30	27e5	30	2,852,933	1.8e−5	1.3e−9

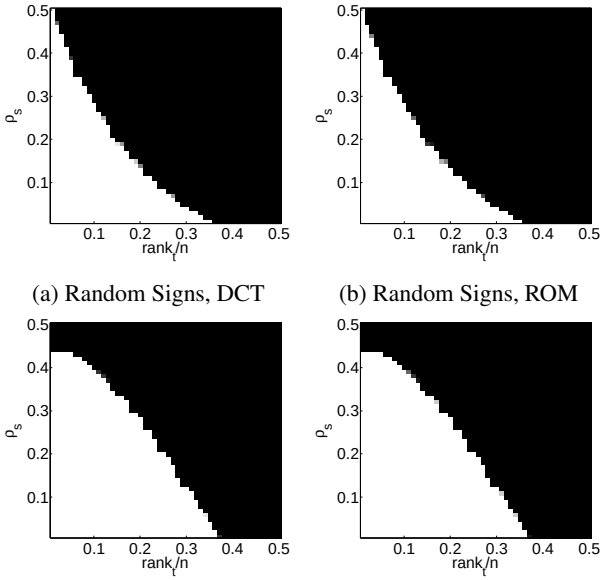
$r = \text{rank}_t(\boldsymbol{\mathcal{L}}_0) = 0.1n, m = \|\boldsymbol{\mathcal{S}}_0\|_0 = 0.2n^3$

n	r	m	$\text{rank}_t(\hat{\boldsymbol{\mathcal{L}}})$	$\|\hat{\boldsymbol{\mathcal{S}}}\|_0$	$\frac{\|\hat{\boldsymbol{\mathcal{L}}}-\boldsymbol{\mathcal{L}}_0\|_F}{\|\boldsymbol{\mathcal{L}}_0\|_F}$	$\frac{\|\hat{\boldsymbol{\mathcal{S}}}-\boldsymbol{\mathcal{S}}_0\|_F}{\|\boldsymbol{\mathcal{S}}_0\|_F}$
100	10	2e5	10	201,070	4.3e−6	2.3e−9
200	20	16e5	20	1,614,206	5.4e−6	9.9e−10
300	30	54e5	30	5,457,874	9.6e−6	9.5e−10

applicable to the case using $L : \mathbb{R}^{n_1\times n_2\times n_3} \to \mathbb{C}^{n_1\times n_2\times n_3}$. Even the proofs in [16] cannot handle this general case either. Their used discrete Fourier transform has some special properties which are crucial in their proofs. In particular, in [16], $\boldsymbol{\mathfrak{e}}_{ijk} = \mathring{\boldsymbol{\mathfrak{e}}}_i * \dot{\boldsymbol{\mathfrak{e}}}_k * \mathring{\boldsymbol{\mathfrak{e}}}_j^\top$ is the unit tensor and it satisfies $\dot{\boldsymbol{\mathfrak{e}}}_k * \mathring{\boldsymbol{\mathfrak{e}}}_j^* * \mathring{\boldsymbol{\mathfrak{e}}}_j * \dot{\boldsymbol{\mathfrak{e}}}_k^* = \boldsymbol{\mathcal{I}}_1$. These two properties are important in their proofs. However, they cannot hold simultaneously for any $L : \mathbb{R}^{n_1\times n_2\times n_3} \to \mathbb{R}^{n_1\times n_2\times n_3}$ (they hold for the special discrete Fourier transform). Our proofs have to handle more complicated operations, e.g., $L(L(\boldsymbol{\mathfrak{e}}_{ijk}))$, which do not appear in [16]. Also, we use less restrictive tensor incoherence conditions (14)-(15). The differences in the tensor basis and tensor incoherence conditions are fundamental, and thus they lead to several key differences in the proofs. Please refer to the supplementary material for more details.

5. Experiments

In this section, we conduct experiments to show that a) program (2) indeed recovers the low-rank and sparse parts correctly, and thus verify our result in Theorem 2; b) our TRPCA outperforms RPCA and other TRPCA methods in practice.



(a) Random Signs, DCT (b) Random Signs, ROM

(c) Coherent Signs, DCT (d) Coherent Signs, ROM

Figure 1: Correct recovery for varying rank and sparsity. Fraction of correct recoveries across 10 trials, as a function of $\text{rank}_t(\boldsymbol{\mathcal{L}}_0)$ (x-axis) and sparsity of $\boldsymbol{\mathcal{S}}_0$ (y-axis). (a)-(b) $\text{sgn}(\boldsymbol{\mathcal{S}}_0)$ is random, DCT and ROM are used as L, respectively; (c)-(d) $\boldsymbol{\mathcal{S}}_0 = \mathcal{P}_{\boldsymbol{\Omega}}\text{sgn}(\boldsymbol{\mathcal{L}}_0)$. DCT and ROM are used as L, respectively.

5.1. Exact Recovery from Varying Fractions of Error

We first verify the recovery guarantee in Theorem 2 on randomly generated tensors. Since Theorem 2 holds for any linear transform L with $\boldsymbol{L}$ satisfying (8). So we consider two cases of $\boldsymbol{L}$: (1) Discrete Cosine Transform (DCT) which is used in the original work of transforms based t-product [10]. We use the Matlab command `dct` to generate the DCT matrix $\boldsymbol{L}$. (2) Random Orthogonal Matrix (ROM) generated by the method with codes available online[1]. In both cases, (8) holds with $\ell = 1$. We simply consider the tensors of size $n \times n \times n$, with n =100, 200 and 300. We generate $\boldsymbol{\mathcal{L}}_0$ with tubal rank r by $\boldsymbol{\mathcal{L}}_0 = \boldsymbol{\mathcal{P}} *_L \boldsymbol{\mathcal{Q}}^\top$, where $\boldsymbol{\mathcal{P}}$ and $\boldsymbol{\mathcal{Q}}$ are $n \times r \times n$ tensors with entries independently sampled from $\mathcal{N}(0, 1/n)$ distribution. The support set $\boldsymbol{\Omega}$ (with size m) of $\boldsymbol{\mathcal{S}}_0$ is chosen uniformly at random. For all $(i, j, k) \in \boldsymbol{\Omega}$, let $[\boldsymbol{\mathcal{S}}_0]_{ijk} = \boldsymbol{\mathcal{M}}_{ijk}$, where $\boldsymbol{\mathcal{M}}$ is a tensor with independent Bernoulli ± 1 entries. For different size n, we set the tubal rank of $\boldsymbol{\mathcal{L}}_0$ as $0.1n$ and consider two cases of the sparsity $m = \|\boldsymbol{\mathcal{S}}_0\|_0 = 0.1n^3$ and $0.2n^3$.

Table 1 and 2 report the recovery results which use DCT and ROM as the linear transform L, respectively. It can be seen that the tubal rank estimations of $\boldsymbol{\mathcal{L}}_0$ are correct in all cases and the relative errors $\|\hat{\boldsymbol{\mathcal{L}}} - \boldsymbol{\mathcal{L}}_0\|_F / \|\boldsymbol{\mathcal{L}}_0\|_F$ are

[1] https://www.mathworks.com/matlabcentral/fileexchange/11783-randorthmat.

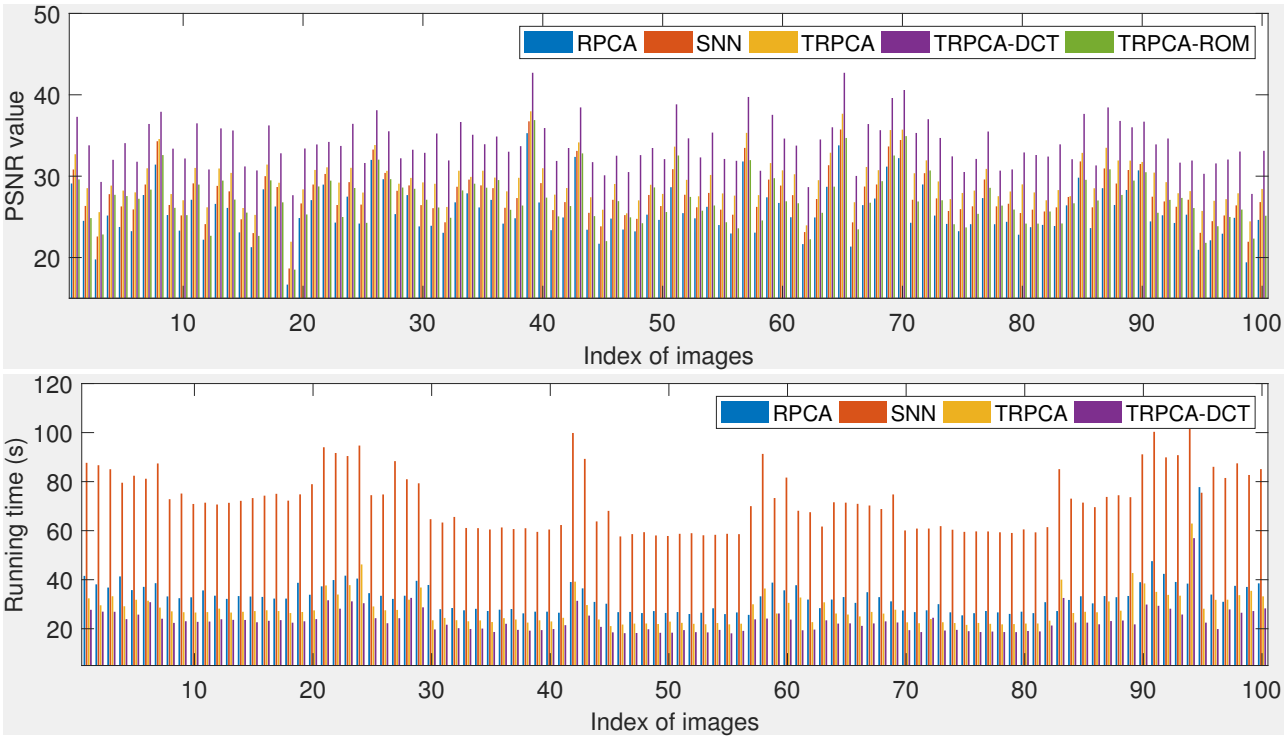


Figure 2: Comparison of the PSNR values (up) and running time (bottom) of RPCA, SNN, TRPCA, our TRPCA-DCT and TRPCA-ROM for image denoising on 50 images. **Best viewed in $\times 2$ sized color pdf file.**

very small (all around 10^{-5}). The sparsity estimation of $\boldsymbol{\mathcal{S}}_0$ is not as exact as the rank estimation. But the relative errors $\|\hat{\boldsymbol{\mathcal{S}}} - \boldsymbol{\mathcal{S}}_0\|_F / \|\boldsymbol{\mathcal{S}}_0\|_F$ are all very small (less than 10^{-8}). These results well verify our theory in Theorem 2.

5.2. Phase Transition in Tubal Rank and Sparsity

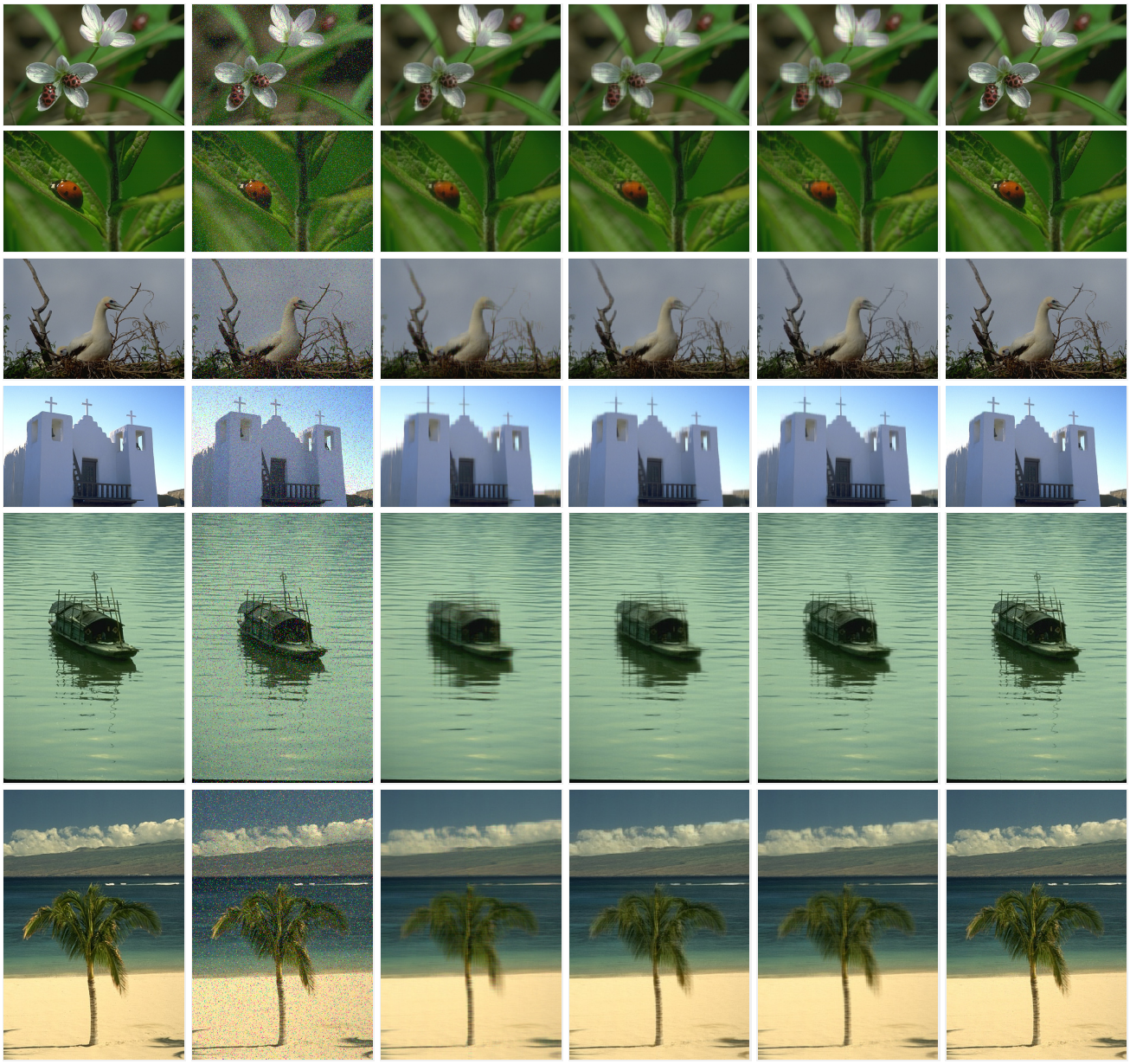
Theorem 2 shows that the recovery is correct when the tubal rank of $\boldsymbol{\mathcal{L}}_0$ is relatively low and $\boldsymbol{\mathcal{S}}_0$ is relatively sparse. Now we examine the recovery phenomenon with varying tubal rank of $\boldsymbol{\mathcal{L}}_0$ from varying sparsity of $\boldsymbol{\mathcal{S}}_0$. We consider tensors of size $\mathbb{R}^{n\times n\times n_3}$, where $n = 100$ and $n_3 = 50$. We generate $\boldsymbol{\mathcal{L}}_0$ in the same way as the previous section. We consider two distributions of the support of $\boldsymbol{\mathcal{S}}_0$. The first case is the Bernoulli model for the support of $\boldsymbol{\mathcal{S}}_0$, with random signs: each entry of $\boldsymbol{\mathcal{S}}_0$ takes on value 0 with probability $1-\rho$, and values ± 1 each with probability $\rho/2$. The second case chooses the support $\boldsymbol{\Omega}$ in accordance with the Bernoulli model, but this time sets $\boldsymbol{\mathcal{S}}_0 = \mathcal{P}_{\boldsymbol{\Omega}}\text{sgn}(\boldsymbol{\mathcal{L}}_0)$. We set $\frac{r}{n} = [0.01 : 0.01 : 0.5]$ and $\rho_s = [0.01 : 0.01 : 0.5]$. For each $(\frac{r}{n}, \rho_s)$-pair, we simulate 10 test instances and declare a trial to be successful if $\hat{\boldsymbol{\mathcal{L}}}$ satisfies $\|\hat{\boldsymbol{\mathcal{L}}} - \boldsymbol{\mathcal{L}}_0\|_F / \|\boldsymbol{\mathcal{L}}_0\|_F \leq 10^{-3}$. Figure 1 shows the fraction of correct recovery for each pair $(\frac{r}{n}, \rho_s)$ for two settings of $\boldsymbol{\mathcal{S}}_0$. The white region indicates the exact recovery while the black one indicates the failure. The experiment shows that the recovery is correct when the tubal rank of $\boldsymbol{\mathcal{L}}_0$ is relatively low and the errors $\boldsymbol{\mathcal{S}}_0$ is relatively sparse. More importantly, the results show that the used linear transforms are not important, as long as property (8) holds. The recovery performances are very similar even different linear transforms are used. This experiment further well verify our main result in Theorem 2.

5.3. Applications to Image Recovery

The marix and tensor completion have been applied for image recovery [16]. The main motivation is that the color image can be well approximated by low-rank matrices or tensors. In this experiment, we consider to apply our TRPCA model using DCT for image recovery, and compare our method with state-of-the-art methods.

For any color image of size $n_1 \times n_2$, we can formate it as a tensor $n_1 \times n_2 \times n_3$, where $n_3 = 3$. The frontal slices correspond to the three channels of the color images[2]. We randomly select 100 color images from the Berkeley segmentation dataset [19] for this test. We randomly set 10% of pixels to random values in [0, 255], and their positions are unknown. We compare our TRPCA model with RPCA [3], SNN [8] and TRPCA [16], which also own the theo-

[2]There have different ways of tensor constructions from color images. We observe that this way of tensor construction achieves the best performance in practice.



(a) (b) (c) (d) (e) (f)

Figure 3: Performance comparison for image recovery on some sample images. (a) Original image; (b) observed image; (c)-(f) recovered images by RPCA, SNN, TRPCA, and our TRPCA-DCT, respectively.

retical recovery guarantee. For RPCA, we apply it on each channel separably and combine the results. For SNN, we set $\boldsymbol{\lambda} = [15, 15, 1.5]$. TRPCA [16] uses the discrete Fourier transform. In this experiment, we use DCT and ROM as the transforms in our new TRPCA method, denoted as TRPCA-DCT and TRPCA-ROM, respectively. The Peak Signal-to-Noise Ratio (PSNR) value [16] is used to evaluate the recovery performance. The higher PSNR value indicates better recovery performance.

Figure 2 shows the comparison of the PSNR values and running time of all the compared methods. Some examples are given in Figure 3. It can be seen that in most cases, our new TRPCA-DCT achieves the best performance. This implies that the discrete Fourier transform used in [16] may not be the best for this task, though the reason is not very clear now. However, if the random orthogonal matrix (ROM) is used as the linear transform, the results are generally bad. This is reasonable and it implies that the choice of

the linear transform L is crucial in practice, though the best linear transform is currently not clear. Figure 2 (b) shows that our method is as efficient as RPCA and TRPCA.

6. Conclusions

Based on the t-product under invertible linear transforms, we defined a new tensor nuclear norm and proposed a new TRPCA model given the transforms satisfying certain conditions. We theoretically prove that the convex TRPCA model can recover both the low-rank and sparse components exactly.

This work provides a new direction for the pursuit of low-rank tensor recovery. The tensor tubal rank and tensor nuclear norm definitions both depend on the given tensor and the used linear transforms. The best transforms will be task dependent. So, looking for the optimized transforms is a very interesting future work. It is always important to apply such a new technique for some other applications, e.g., subspace clustering for motion segmentation and outlier detection [27].

References

[1] Evrim Acar, Tamara G. Kolda, Daniel M. Dunlavy, and Morten Mørup. Scalable tensor factorizations with missing data. In *ICDM*, pages 701–712, 2009. 1

[2] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011. 4

[3] Emmanuel J Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *J. ACM*, 58(3), 2011. 1, 6

[4] Yudong Chen. Incoherence-optimal matrix completion. *IEEE Trans. Information Theory*, 61(5):2909–2923, May 2015. 2

[5] Sanne Engelen, Stina Frosch, and Bo M Jorgensen. A fully robust parafac method for analyzing fluorescence data. *J. Chemometrics*, 23(3):124–131, 2009. 1

[6] Thomas Franz, Antje Schultz, Sergej Sizov, and Steffen Staab. Triplerank: Ranking semantic web data by tensor decomposition. In *Int'l Semantic Web Conf.*, pages 213–228, 2009. 1

[7] Christopher J Hillar and Lek-Heng Lim. Most tensor problems are NP-hard. *J. ACM*, 60(6):45, 2013. 1

[8] Bo Huang, Cun Mu, Donald Goldfarb, and John Wright. Provable models for robust low-rank tensor completion. *Pacific J. Optimization*, 11(2):339–364, 2015. 1, 6

[9] Hui Ji, Sibin Huang, Zuowei Shen, and Yuhong Xu. Robust video restoration by joint sparse and low rank matrix approximation. *SIAM J. Imaging Sciences*, 4(4):1122–1142, 2011. 1

[10] Eric Kernfeld, Misha Kilmer, and Shuchin Aeron. Tensor–tensor products with invertible linear transforms. *Linear Algebra and its Applications*, 485:545–570, 2015. 2, 3, 5

[11] Misha E Kilmer and Carla D Martin. Factorization strategies for third-order tensors. *Linear Algebra and its Applications*, 435(3):641–658, 2011. 2

[12] Tamara G Kolda and Brett W Bader. Tensor decompositions and applications. *SIAM Rev.*, 51(3):455–500, 2009. 1, 2

[13] Nadia Kreimer and Mauricio D Sacchi. Nuclear norm minimization and tensor completion in exploration seismology. In *ICASSP*, pages 4275–4279, 2013. 1

[14] Ji Liu, Przemyslaw Musialski, Peter Wonka, and Jieping Ye. Tensor completion for estimating missing values in visual data. *TPAMI*, 35(1):208–220, 2013.

[15] Canyi Lu, Jiashi Feng, Yudong Chen, Wei Liu, Zhouchen Lin, and Shuicheng Yan. Tensor robust principal component analysis: Exact recovery of corrupted low-rank tensors via convex optimization. In *CVPR*. IEEE, 2016. 2, 3

[16] Canyi Lu, Jiashi Feng, Yudong Chen, Wei Liu, Zhouchen Lin, and Shuicheng Yan. Tensor robust principal component analysis with a new tensor nuclear norm. *TPAMI*, 2019. 2, 3, 4, 5, 6, 7

[17] Canyi Lu, Jiashi Feng, Zhouchen Lin, and Shuicheng Yan. Exact low tubal rank tensor recovery from Gaussian measurements. In *IJCAI*, 2018. 2

[18] Canyi Lu, Xi Peng, and Yunchao Wei. Low-rank tensor completion with a new tensor nuclear norm induced by invertible linear transforms. In *CVPR*, pages 5996–6004, 2019. 2

[19] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, volume 2, pages 416–423. IEEE, 2001. 6

[20] Cun Mu, Bo Huang, John Wright, and Donald Goldfarb. Square deal: Lower bounds and improved relaxations for tensor recovery. In *ICML*, pages 73–81, 2014. 1

[21] Yigang Peng, Arvind Ganesh, John Wright, Wenli Xu, and Yi Ma. RASL: Robust alignment by sparse and low-rank decomposition for linearly correlated images. *TPAMI*, 34(11):2233–2246, 2012. 1

[22] Oguz Semerci, Ning Hao, Misha E Kilmer, and Eric L Miller. Tensor-based formulation and nuclear norm regularization for multienergy computed tomography. *TIP*, 23(4):1678–1693, 2014. 2

[23] Nicholas D Sidiropoulos, Rasmus Bro, and Georgios B Giannakis. Parallel factor analysis in sensor array processing. *TSP*, 48(8):2377–2388, 2000. 1

[24] Marco Signoretto, R Van De Plas, B De Moor, and Johan A K Suykens. Tensor versus matrix completion: A comparison with application to spectral data. *IEEE Signal Processing Letters*, 18(7):403–406, 2011. 1

[25] Ryota Tomioka, Taiji Suzuki, Kohei Hayashi, and Hisashi Kashima. Statistical performance of convex tensor decomposition. In *NIPS*, pages 972–980, 2011. 1

[26] M Alex O Vasilescu and Demetri Terzopoulos. Multilinear analysis of image ensembles: Tensorfaces. In *ECCV*, pages 447–460. Springer, 2002. 1

[27] Pan Zhou and Jiashi Feng. Outlier-robust tensor PCA. In *CVPR*, pages 3938–3946, 2017. 8