

# Robust Data Clustering with Outliers via Transformed Tensor Low-Rank Representation

Tong Wu

Beijing Institute for General Artificial Intelligence  
wutong@bigai.ai

## Abstract

Recently, tensor low-rank representation (TLRR) has become a popular tool for tensor data recovery and clustering, due to its empirical success and theoretical guarantees. However, existing TLRR methods consider Gaussian or gross sparse noise, inevitably leading to performance degradation when the tensor data are contaminated by outliers or sample-specific corruptions. This paper develops an outlier-robust tensor low-rank representation (OR-TLRR) method that provides outlier detection and tensor data clustering simultaneously based on the t-SVD framework. For tensor observations with arbitrary outlier corruptions, OR-TLRR has provable performance guarantee for exactly recovering the row space of clean data and detecting outliers under mild conditions. Moreover, an extension of OR-TLRR is proposed to handle the case when parts of the data are missing. Finally, extensive experimental results on synthetic and real data demonstrate the effectiveness of the proposed algorithms. We release our code at <https://github.com/twugithub/2024-AISTATS-ORTLRR>

where  $\mathcal{L}_0$  is a low-rank tensor with the tensor columns drawn from a union of tensor subspaces (see Section 3.2), and  $\mathcal{E}_0$  is a column-sparse tensor. Both components are of arbitrary magnitudes. In particular, we do not know the locations of the nonzero columns of  $\mathcal{E}_0$ , not even how many there are. We aim to exactly recover the low-rank tensor  $\mathcal{L}_0$  from  $\mathcal{X}$  and cluster the samples in  $\mathcal{L}_0$  according to their respective tensor subspaces. This problem is important for many applications, including image/video denoising and inpainting [Liu et al., 2013b, Lu et al., 2020], data clustering [Elhamifar and Vidal, 2013, Zhou et al., 2021], background subtraction [Xia et al., 2021], and network traffic monitoring [Mardani et al., 2015].

This problem has been well studied in the matrix domain. Subspace clustering methods, for example, involve treating vectorized data samples as lying near a union of subspaces and clustering the samples based on the self-expressive model [Elhamifar and Vidal, 2013, Liu et al., 2013a]. While these matrix techniques are remarkably effective, one major shortcoming is that they can only handle 2-way (matrix) data. To deal with multi-way data such as color images, hyperspectral images and videos, one needs to reshape the tensor data into a matrix. Such a vectorization process would destroy the original spatial structure of data and lead to compromised performance [Lu et al., 2020]. This suggests that it might be useful to maintain the intrinsic structure of multi-dimensional data for efficient and reliable subsequent processing.

Recently, several tensor-based clustering algorithms that exploit the spatial aspects of tensor data using multilinear algebra tools have been proposed [Kernfeld et al., 2014, Fu et al., 2016, Zhang et al., 2018, Wu and Bajwa, 2018, Wu, 2020, Zhou et al., 2021, Yang et al., 2022, Wu, 2023]. In particular, motivated by the notion of the tensor-tensor product (t-product) [Kilmer and Martin, 2011] which generalizes the matrix multiplication for third-order tensors based on the Discrete Fourier Transform (DFT), one can assume that multi-way data lie near

## 1 INTRODUCTION

In this work, we are interested in clustering 3-way tensor data in the presence of outliers. Formally, suppose we are given a data tensor  $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$  with each lateral slice corresponding to one data sample, and we know the tensor can be decomposed as

$$\mathcal{X} = \mathcal{L}_0 + \mathcal{E}_0, \quad (1)$$

Proceedings of the 27<sup>th</sup> International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238. Copyright 2024 by the author(s).

a union of tensor subspaces [Kernfeld et al., 2014, Wu and Bajwa, 2018, Wu, 2020, Zhou et al., 2021]. As a consequence, each data sample in a union of tensor subspaces can be well represented by a t-linear combination of other points in the dataset. Interestingly, the t-product can be generalized by replacing DFT with any invertible linear transforms [Kernfeld et al., 2015]. Such a new transform based t-product is of great interest because it allows to use data-adaptive transforms for different types of tensor data. Driven by this advantage, extensive numerical examples have shown its effectiveness in many applications [Lu et al., 2019b, Song et al., 2020, Jiang et al., 2020, Lu, 2021, Kong et al., 2021, Wang et al., 2021, Qin et al., 2022, Yang et al., 2022, Jiang et al., 2023].

However, all the aforementioned tensor clustering methods simply assume the noise is sparse and uniformly distributed across  $\mathcal{E}_0$ . Thus, such methods cannot effectively handle outliers or sample-specific corruptions, which are common in real scenarios because of sensor failures, uncontrolled environments, etc. Another limitation of existing tensor subspace clustering methods is that most of them are designed for complete data. When dealing with incomplete data, we need to first recover the incomplete tensor data using any off-the-shelf tensor completion methods and then cluster the recovered data. This two-step approach may lead to sub-optimal clustering results because the downstream clustering task is independent of the optimization problem that is used for tensor recovery [Yang et al., 2022]. While the work in [Yang et al., 2022] can simultaneously recover and cluster the incomplete and noisy data, it still requires learning a dictionary from the incomplete data before clustering can be carried out.

Our goal in this paper is to perform robust low-rank tensor analysis on both complete and missing data for improved data clustering in the presence of outliers. We put forth an outlier-robust tensor low-rank representation (OR-TLRR) method which clusters the observed tensor data and detects outliers through convex optimization. Our model is more generic as it is allowed to use any invertible linear transforms that satisfy certain conditions. Interestingly, as we show in Section 4.4, the row space of  $\mathcal{L}_0$  determines the segmentation result of the ‘‘authentic’’ samples. We theoretically show that OR-TLRR guarantees the exact recovery of the row space of  $\mathcal{L}_0$  when  $\mathcal{L}_0$  and  $\mathcal{E}_0$  satisfy certain assumptions. Different from [Zhou et al., 2021, Yang et al., 2022] that use the  $\ell_1$  norm of  $\mathcal{E}$  to handle sparse noise, we focus on a different problem and thus there exist critical differences in theoretical analysis and guarantees. Lastly, we gener-

alize the OR-TLRR algorithm so that it can obtain a tensor low-rank representation of the incomplete data using only the observed entries. Experiments on synthetic and real data show the efficacy of the proposed algorithms. The main contributions of this work can be summarized as follows:

- We propose a novel OR-TLRR method for low-rank tensor analysis based on any invertible linear transforms. OR-TLRR handles outliers and sample-specific corruptions which cannot be well handled by existing tensor clustering methods.
- We provide theoretical performance guarantees for OR-TLRR: under mild conditions, OR-TLRR can exactly recover the row space of  $\mathcal{L}_0$  and detect outliers.
- We extend OR-TLRR for tensor subspace clustering with missing entries called OR-TLRR by entry wise zero fill (OR-TLRR-EWZF), in which the self-expressiveness error is restricted only to the observed entries.

## 2 RELATED WORK

Existing subspace clustering methods [Liu and Yan, 2011, Eriksson et al., 2012, Liu et al., 2012, Elhamifar and Vidal, 2013, Liu et al., 2013a, Tang et al., 2014, Yang et al., 2015, You et al., 2016, Lu et al., 2019a] that are based on the self-expressive model mainly follow a two-stage approach: (i) learning an affinity matrix by solving an optimization problem from the data using different regularizations, and (ii) applying spectral clustering (e.g., Ncut [Shi and Malik, 2000]) on the affinity matrix. The key component in subspace learning is to construct a good affinity graph. In particular, the matrix-based low-rank representation (LRR) [Liu et al., 2012, Liu et al., 2013a] algorithm imposes a low-rank constraint on the representation coefficient matrix to capture the global structure of data. Due to its physical interpretation and promising performance, many variants of LRR have been further studied. The latent LRR [Liu and Yan, 2011] extends LRR to handle the hidden effects and integrates subspace segmentation and feature extraction into a unified framework. The structure-constrained LRR [Tang et al., 2014] improves LRR for the disjoint subspace segmentation. Typical subspace clustering methods, despite their effectiveness, neglect the spatial information within the data. Due to the powerful learning ability, deep clustering methods can capture the complex nonlinear structure within the data [Xie et al., 2016, Ji et al., 2017, Ji et al., 2019,

Li et al., 2021]. However, they cannot solve the problem of clustering high-dimensional but small-scale tensor data considered in this work. In addition, they lack the theoretical guarantee for data clustering.

In order to better capture the geometry of tensor data, a high-order model based upon t-product [Kilmer and Martin, 2011, Kilmer et al., 2013] that has in particular helped advance the state-of-the-art in many applications is the union-of-tensor-subspaces model [Kernfeld et al., 2014, Wu and Bajwa, 2018, Zhou et al., 2021], which dictates that data lie near a mixture of low-dimensional tensor subspaces. One striking advantage of the t-product is its capability in capturing the “spatial-shifting” correlation that is ubiquitous in real-world data. For the purpose of inducing tensor subspace-preserving solutions, existing methods use different regularizations on the representation coefficients. For example, sparse submodule clustering (SSmC) [Kernfeld et al., 2014] imposes a group sparsity constraint on the representation tensor, while the tensor nuclear norm [Lu et al., 2020] is adopted in tensor low-rank representation (TLRR) [Zhou et al., 2021] as a convex surrogate of the tensor tubal rank. Motivated by the general t-product definition performed on unitary transforms [Kernfeld et al., 2015], a more general TLRR model that can handle incomplete data is proposed in [Yang et al., 2022]. However, both [Zhou et al., 2021] and [Yang et al., 2022] assume that noise is Gaussian or sparse and thus cannot effectively handle outlier corruptions. Indeed, robust tensor subspace clustering with outliers appears to be a challenging problem that has not been well studied both in practice and in theory. We conclude by noting that tensor nuclear norm has been utilized for multi-view clustering [Xie et al., 2018, Su et al., 2023]. Nonetheless, these works are still based on the union of subspaces model and reduce to different variants of [Elhamifar and Vidal, 2013, Liu et al., 2013a] for single-view data.

### 3 NOTATIONS AND PRELIMINARIES

In this section, we will introduce the notations and give the basic definitions used throughout the paper.

#### 3.1 Notations

For brevity, we summarize the notations in Table 1. Throughout this paper, the fields of real number and complex number are denoted as  $\mathbb{R}$  and  $\mathbb{C}$ , respectively. The definitions of the identity tensor and the conjugate transpose of a tensor will be given in Section 3.2. The frontal-slice-wise product of two third-order ten-

Table 1: Notational convention.

$\mathcal{A}$	A tensor.	$\mathbf{a}$	A vector.
$\mathbf{A}$	A matrix.	$\mu$	A scalar.
$\mathbf{I}_n$	The identity matrix.	$\mathbf{A}^H$	The conjugate transpose of $\mathbf{A}$ .
$\mathbf{A}_{i,j}$	The $(i,j)$ -th element of $\mathbf{A}$ .	$\text{tr}(\mathbf{A})$	The trace of $\mathbf{A}$ .
$\mathbf{a}_j^T$	The $j$ -th column of $\mathbf{A}$ .	$\ \mathbf{A}\ $	The largest singular value of $\mathbf{A}$ .
$\mathbf{a}_j$	The $j$ -th row of $\mathbf{A}$ .	$\ \mathbf{A}\ _*$	Sum of singular values of $\mathbf{A}$ .
$\mathcal{I}_n$	The identity tensor.	$\mathbf{A}^{(i)}$	$\mathbf{A}^{(i)} = \mathcal{A}(:, :, i)$ .
$\mathcal{A}_{i,j,k}$	The $(i,j,k)$ -th entry of $\mathcal{A}$ .	$\mathcal{A}_{(i)}$	$\mathcal{A}_{(i)} = \mathcal{A}(:, i, :)$ .
$\mathcal{A}(:, i, :)$	The $i$ -th lateral slice of $\mathcal{A}$ .	$\mathcal{A}^H$	The conjugate transpose of $\mathcal{A}$ .
$\mathcal{A}(:, :, i)$	The $i$ -th frontal slice of $\mathcal{A}$ .	$\ \mathcal{A}\ _{2,1}$	$\ \mathcal{A}\ _{2,1} = \sum_j \ \mathcal{A}(:, j, :)\ _F$ .
$\mathcal{A}(i, j, :)$	The $(i,j)$ -th tube of $\mathcal{A}$ .	$\ \mathcal{A}\ _F$	$\ \mathcal{A}\ _F = \sqrt{\sum_{i,j,k}  \mathcal{A}_{i,j,k} ^2}$ .
$\mathcal{P}_{\mathcal{U}}(\mathcal{A})$	$\mathcal{P}_{\mathcal{U}}(\mathcal{A}) = \mathcal{U} *_{\mathcal{L}} \mathcal{U}^H *_{\mathcal{L}} \mathcal{A}$	$\mathcal{P}_{\Theta}$	The projection onto $\Theta$ .
$\mathcal{P}_{\mathcal{V}}(\mathcal{A})$	$\mathcal{P}_{\mathcal{V}}(\mathcal{A}) = \mathcal{A} *_{\mathcal{L}} \mathcal{V} *_{\mathcal{L}} \mathcal{V}^H$	$\mathcal{P}_{\Theta^\perp}$	The projection onto $\Theta^\perp$ .
$\mathcal{P}_{\mathcal{V}}^{\mathcal{L}}(\mathcal{A})$	$\mathcal{P}_{\mathcal{V}}^{\mathcal{L}}(\mathcal{A}) = \mathcal{V} *_{\mathcal{L}} \mathcal{V}^H *_{\mathcal{L}} \mathcal{A}$	$\mathcal{B}(\mathcal{E})$	$\{\tilde{\mathcal{E}} : \tilde{\mathcal{E}}(:, i, :) = \frac{\mathcal{E}(:, i, :)}{\ \mathcal{E}(:, i, :)\ _F} \mid (i \in \Theta); \mathcal{P}_{\Theta^\perp} \tilde{\mathcal{E}} = \mathbf{0}\}$ .

sors  $\mathcal{A} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$  and  $\mathcal{B} \in \mathbb{C}^{n_2 \times n_4 \times n_3}$ , denoted by  $\mathcal{A} \odot \mathcal{B}$ , is a tensor  $\mathcal{C} \in \mathbb{C}^{n_1 \times n_4 \times n_3}$  such that  $\mathcal{C}^{(i)} = \mathbf{A}^{(i)} \mathbf{B}^{(i)}$ ,  $i = 1, \dots, n_3$  [Kernfeld et al., 2015].

#### 3.2 Preliminaries

The third-order t-product under an invertible linear transform  $L$  establishes the groundwork for the development of our algorithm and its definition was first given in [Kernfeld et al., 2015]. In this paper, the transformation matrix  $L$  defining the transform  $L$  is restricted to be orthogonal, i.e.,  $L \in \mathbb{C}^{n_3 \times n_3}$  satisfying

$$LL^H = L^H L = \tau \mathbf{I}_{n_3}, \quad (2)$$

where  $\tau > 0$  is a constant. We define the associated linear transform  $L(\cdot) : \mathbb{R}^{n_1 \times n_2 \times n_3} \rightarrow \mathbb{C}^{n_1 \times n_2 \times n_3}$  which gives  $\bar{\mathcal{A}}$  by performing a linear transform on any  $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$  along the 3rd dimension with inverse mapping  $L^{-1}(\cdot)$  as

$$\bar{\mathcal{A}} = L(\mathcal{A}) = \mathcal{A} \times_3 L \text{ and } L^{-1}(\bar{\mathcal{A}}) = \mathcal{A} \times_3 L^{-1}. \quad (3)$$

Here,  $\times_3$  denotes the mode-3 tensor-matrix product [Kolda and Bader, 2009], i.e.,  $\mathcal{X} \times_3 \mathbf{A} = \text{fold}_3(\mathbf{A} \cdot \text{unfold}_3(\mathcal{X}))$ , where  $\text{unfold}_3 : \mathbb{C}^{n_1 \times n_2 \times n_3} \rightarrow \mathbb{C}^{n_3 \times n_1 n_2}$  is the mode-3 unfolding operator and  $\text{fold}_3$  is its inverse operator [Kolda and Bader, 2009].

**Definition 1** (t-product [Kernfeld et al., 2015]). *The t-product of any  $\mathcal{A} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$  and  $\mathcal{B} \in \mathbb{C}^{n_2 \times n_4 \times n_3}$  under the invertible linear transform  $L$  in (3), is defined as  $\mathcal{C} = \mathcal{A} *_{\mathcal{L}} \mathcal{B} \in \mathbb{C}^{n_1 \times n_4 \times n_3}$  such that  $L(\mathcal{C}) = L(\mathcal{A}) \odot L(\mathcal{B})$ .*

We denote  $\bar{\bar{\mathcal{A}}} \in \mathbb{C}^{n_1 n_3 \times n_2 n_3}$  as

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

where  $\text{bdiag}$  is an operator which maps  $\bar{\mathcal{A}}$  to  $\bar{\bar{\mathcal{A}}}$ . The conjugate transpose of a tensor  $\mathcal{A} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$  is

the tensor  $\mathcal{A}^H \in \mathbb{C}^{n_2 \times n_1 \times n_3}$  satisfying  $L(\mathcal{A}^H)^{(i)} = (L(\mathcal{A})^{(i)})^H$ ,  $i = 1, 2, \dots, n_3$ . The identity tensor  $\mathcal{I}_n \in \mathbb{C}^{n \times n \times n_3}$  is a tensor such that each frontal slice of  $L(\mathcal{I}_n) = \bar{\mathcal{I}}_n$  is the identity matrix  $\mathbf{I}_n$ . Then  $\mathcal{I}_n = L^{-1}(\bar{\mathcal{I}}_n)$  gives the identity tensor under  $L$ . A tensor  $\mathcal{Q} \in \mathbb{C}^{n \times n \times n_3}$  is orthogonal if it satisfies  $\mathcal{Q} *_L \mathcal{Q}^H = \mathcal{Q}^H *_L \mathcal{Q} = \mathcal{I}_n$ . A tensor is called f-diagonal if each of its frontal slices is a diagonal matrix. The spectral norm of  $\mathcal{A}$  is defined as  $\|\mathcal{A}\| = \|\bar{\mathcal{A}}\|$ .

**Definition 2** (t-SVD [Song et al., 2020]). *Let  $L$  be any invertible linear transform in (3). For any  $\mathcal{A} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$ , it can be factorized as  $\mathcal{A} = \mathcal{U} *_L \mathcal{S} *_L \mathcal{V}^H$ , where  $\mathcal{U} \in \mathbb{C}^{n_1 \times n_1 \times n_3}$  and  $\mathcal{V} \in \mathbb{C}^{n_2 \times n_2 \times n_3}$  are orthogonal, and  $\mathcal{S} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$  is an f-diagonal tensor.*

**Definition 3** (Tensor tubal rank [Song et al., 2020]). *For any  $\mathcal{A} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$ , the tensor tubal rank  $\text{rank}_t(\mathcal{A})$  under  $L$  in (3) is defined as the number of nonzero singular tubes of  $\mathcal{S}$ , i.e.,  $\text{rank}_t(\mathcal{A}) = \#\{i : \mathcal{S}(i, :, \cdot) \neq \mathbf{0}\}$ , where  $\mathcal{S}$  is from the t-SVD of  $\mathcal{A} = \mathcal{U} *_L \mathcal{S} *_L \mathcal{V}^H$ .*

For computational simplicity, we use the skinny t-SVD throughout this paper. For any  $\mathcal{A} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$  with tubal rank  $r$ , the skinny t-SVD is given by  $\mathcal{A} = \mathcal{U} *_L \mathcal{S} *_L \mathcal{V}^H$ , where  $\mathcal{U} \in \mathbb{C}^{n_1 \times r \times n_3}$ ,  $\mathcal{S} \in \mathbb{C}^{r \times r \times n_3}$ , and  $\mathcal{V} \in \mathbb{C}^{n_2 \times r \times n_3}$ , in which  $\mathcal{U}^H *_L \mathcal{U} = \mathcal{V}^H *_L \mathcal{V} = \mathcal{I}_r$ .

**Definition 4** (Tensor nuclear norm [Song et al., 2020]). *The tensor nuclear norm of  $\mathcal{A} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$  under  $L$  in (3) is defined as  $\|\mathcal{A}\|_* = \frac{1}{\tau} \sum_{i=1}^{n_3} \|\bar{\mathcal{A}}^{(i)}\|_* = \frac{1}{\tau} \|\bar{\mathcal{A}}\|_*$ .*

**Definition 5** (Tensor subspace [Zhou et al., 2021]). *Given a third-order tensor  $\mathcal{D} = [\mathcal{D}_{(1)}, \dots, \mathcal{D}_{(p)}] \in \mathbb{R}^{n_1 \times p \times n_3}$  in which the elements  $\mathcal{D}_{(i)}$ 's are linearly independent, i.e., there is not a nonzero  $\mathcal{C} \in \mathbb{R}^{p \times 1 \times n_3}$  satisfying  $\mathcal{D} *_L \mathcal{C} = \mathbf{0}$ . Then the set  $\mathcal{S}_{n_3}^{n_1} = \{\mathcal{Y} | \mathcal{Y} = \mathcal{D} *_L \mathcal{C}, \forall \mathcal{C} \in \mathbb{R}^{p \times 1 \times n_3}\}$  is called a tensor subspace of dimension  $\dim(\mathcal{S}_{n_3}^{n_1}) = p$ . Here,  $\mathcal{D}_{(1)}, \dots, \mathcal{D}_{(p)}$  are called the spanning basis of  $\mathcal{S}_{n_3}^{n_1}$ .*

**Definition 6** (Tensor column space [Zhou and Feng, 2017]). *For an arbitrary tensor  $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$  with tubal rank  $r$ , assume the t-SVD of  $\mathcal{A}$  is  $\mathcal{A} = \mathcal{U} *_L \mathcal{S} *_L \mathcal{V}^H$ . Then its column space  $\text{Range}(\mathcal{A})$  is the t-linear space spanned by the columns of  $\mathcal{U} \in \mathbb{R}^{n_1 \times r \times n_3}$ , i.e.,  $\text{Range}(\mathcal{A}) = \{\mathcal{Y} | \mathcal{Y} = \mathcal{U} *_L \mathcal{C}, \forall \mathcal{C} \in \mathbb{R}^{r \times 1 \times n_3}\}$ .*

## 4 OUTLIER-ROBUST TLRR

In this section, we propose our OR-TLRR model for tensor data clustering in the presence of outliers, followed by the theoretical analysis on the recovery guarantee of OR-TLRR. The optimization details and proofs of all the Lemmas and Theorems are provided in the supplementary material.

### 4.1 Formulation of OR-TLRR

Assume that we are given  $n_2$  data samples of size  $n_1 \times n_3$ , stored as lateral slices in a tensor  $\mathcal{X} = \mathcal{L}_0 + \mathcal{E}_0 \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ , where the columns in  $\mathcal{L}_0$  belong to  $c$  distinct tensor subspaces and  $\mathcal{E}_0$  is a tensor corresponding to outliers. The challenge is to recover the low-rank tensor  $\mathcal{L}_0$  from  $\mathcal{X}$  and segment the samples in  $\mathcal{L}_0$  into  $c$  clusters. In many cases, the error term  $\mathcal{E}_0$  has sparse column supports compared to the data size. This implies that the tensor  $\ell_{2,1}$ -norm is appropriate to characterize  $\mathcal{E}_0$ . Let  $L$  be any invertible linear transform in (3) and it satisfies (2). Mathematically, the OR-TLRR model can be described as the following optimization program based on the tensor nuclear norm:

$$\min_{\mathcal{Z}, \mathcal{E}} \|\mathcal{Z}\|_* + \lambda \|\mathcal{E}\|_{2,1} \quad \text{s.t.} \quad \mathcal{X} = \mathcal{X} *_L \mathcal{Z} + \mathcal{E}, \quad (4)$$

where  $\mathcal{Z} \in \mathbb{R}^{n_2 \times n_2 \times n_3}$  corresponds to a coefficient tensor and  $\lambda > 0$  is a parameter. In this work, we directly use the raw tensor data  $\mathcal{X}$  as the dictionary. In this setting, the learned representation tensor  $\mathcal{Z}$  depicts the relations among samples and can be used for clustering. Suppose we have an optimal solution  $(\mathcal{Z}_*, \mathcal{E}_*)$  for (4). For clustering, we first perform outlier detection based on the magnitude of the residual  $\mathcal{E}_*$ , and we use  $\Theta$  to denote the indices of the ‘‘detected’’ outliers. Then we compute an affinity matrix  $\hat{\mathbf{Z}} \in \mathbb{R}^{|\Theta^c| \times |\Theta^c|}$  by setting  $\hat{\mathbf{Z}} = \frac{1}{2n_3} \sum_{k=1}^{n_3} (|\mathbf{Z}_{*[\Theta^c, \Theta^c]}^{(k)}| + |(\mathbf{Z}_{*[\Theta^c, \Theta^c]}^{(k)})^H|)$ , where  $\mathbf{A}_{[\Theta, \Theta]}$  is a submatrix of  $\mathbf{A}$  corresponding to rows and columns indexed by  $\Theta$ . Finally, clustering of the ‘‘detected’’ normal samples can be carried out using existing tools such as Ncut [Shi and Malik, 2000].

### 4.2 Tensor Linear Representation

Before introducing the proposed algorithm for clustering, we explain the connection between the tensor linear representation  $\mathcal{X} = \mathcal{A} *_L \mathcal{Z}$  and canonical vector linear representation. We introduce two operators:  $\text{vec}$  and  $\text{ivec}$ , where  $\text{vec}$  vectorizes each sample  $\mathcal{X}_{(j)}$  to a vector  $\mathbf{x}_j$  of dimension  $n_1 n_3$  and  $\text{ivec}$  is its inverse operation. For each sample  $\mathcal{X}_{(j)}$ , if its vectorized version  $\mathbf{x}_j$  can be linearly represented by a dictionary  $\mathbf{A} \in \mathbb{R}^{n_1 n_3 \times p}$  as

$$\mathbf{x}_j = \mathbf{A} \mathbf{z}_j, \quad \forall j = 1, \dots, n_2, \quad (5)$$

then one can always find two tensors  $\mathcal{A} \in \mathbb{R}^{n_1 \times p \times n_3}$  and  $\mathcal{Z} \in \mathbb{C}^{p \times n_2 \times n_3}$  such that tensor linear representation  $\mathcal{X} = \mathcal{A} *_L \mathcal{Z}$  holds, as stated below.

**Theorem 1.** *If (5) holds, then there exist two tensors  $\mathcal{A} \in \mathbb{R}^{n_1 \times p \times n_3}$  and  $\mathcal{Z} \in \mathbb{C}^{p \times n_2 \times n_3}$  such that*

$$\mathcal{X}_{(j)} = \text{ivec}(\mathbf{x}_j) = \mathcal{A} *_L \mathcal{Z}_{(j)}, \quad \forall j = 1, \dots, n_2, \quad (6)$$

where  $\mathcal{A}$  can be constructed from  $\mathbf{A}$  by setting  $\mathcal{A}_{(j)} = \text{ivec}(\mathbf{a}_j)$  and  $\mathcal{Z}$  can be computed by  $\mathcal{Z} = L^{-1}(\bar{\mathcal{Z}})$  in which  $\bar{\mathcal{Z}}(:, j, k) = \mathbf{z}_j$  ( $k = 1, \dots, n_3$ ). However, if there exists a tensor such that (6) holds, (5) may not hold.

Theorem 1 implies that tensor linear representation under linear transform can capture complex structures underlying the data which cannot be fully described by vector linear representation. We would like to emphasize that this result generalizes the one from [Zhou et al., 2021, Theorem 1] which only considered the DFT. In their proof, some special properties of DFT were used, which lead to several key differences between the two proofs.

### 4.3 Optimization of OR-TLRR

We can use Alternating Direction Method of Multipliers (ADMM) [Lin et al., 2011] to solve problem (4). The major computation lies in the update of  $\mathcal{Z}$ , which requires computing  $n_3$  SVDs of  $n_2 \times n_2$  matrices in the transform domain. Thus directly applying ADMM will be highly time consuming when the number of data samples  $n_2$  is large. To reduce the computational cost, we reformulate (4) as follows. Assume  $\mathbf{U}_{\mathcal{X}} *_{\mathbf{L}} \mathbf{S}_{\mathcal{X}} *_{\mathbf{L}} \mathbf{V}_{\mathcal{X}}^H$  is the skinny t-SVD of  $\mathcal{X}$  and  $r_{\mathcal{X}} = \text{rank}_t(\mathcal{X})$ . One can replace  $\mathcal{X}$  and  $\mathcal{Z}$  in (4) with  $\mathcal{D} = \mathbf{U}_{\mathcal{X}} *_{\mathbf{L}} \mathbf{S}_{\mathcal{X}} \in \mathbb{R}^{n_1 \times r_{\mathcal{X}} \times n_3}$  and  $\mathbf{V}_{\mathcal{X}} *_{\mathbf{L}} \mathcal{Z}'$ , respectively, where  $\mathcal{Z}' \in \mathbb{R}^{r_{\mathcal{X}} \times n_2 \times n_3}$  is one variable to be updated. This gives the following equivalent formulation:

$$\min_{\mathcal{Z}', \mathcal{E}} \|\mathcal{Z}'\|_* + \lambda \|\mathcal{E}\|_{2,1} \quad \text{s.t.} \quad \mathcal{X} = \mathcal{D} *_{\mathbf{L}} \mathcal{Z}' + \mathcal{E}. \quad (7)$$

After obtaining a solution  $(\mathcal{Z}'_*, \mathcal{E}_*)$  to the problem (7), the optimal solution to (4) can be recovered by  $(\mathbf{V}_{\mathcal{X}} *_{\mathbf{L}} \mathcal{Z}'_*, \mathcal{E}_*)$ . For any invertible linear transform  $L$ , the per-iteration complexity is  $\mathcal{O}(r_{\mathcal{X}} n_1 n_2 n_3 + r_{\mathcal{X}}(n_1 + n_2)n_3^2)$ . For some special transforms, e.g., DFT, the per-iteration complexity is  $\mathcal{O}(r_{\mathcal{X}} n_1 n_2 n_3 + r_{\mathcal{X}}(n_1 + n_2)n_3 \log(n_3))$ .

### 4.4 Exact Recovery Performance Guarantee

Here we provide the theoretical performance guarantee for OR-TLRR. The skinny t-SVD of  $\mathcal{X}$  and  $\mathcal{L}_0$  are denoted as  $\mathbf{U}_{\mathcal{X}} *_{\mathbf{L}} \mathbf{S}_{\mathcal{X}} *_{\mathbf{L}} \mathbf{V}_{\mathcal{X}}^H = \mathcal{X}$  and  $\mathbf{U}_0 *_{\mathbf{L}} \mathbf{S}_0 *_{\mathbf{L}} \mathbf{V}_0^H = \mathcal{L}_0$ , respectively. We use  $\Theta_0$  to denote the column support of  $\mathcal{E}_0$ . For convenience, we denote  $n_{(1)} = \max(n_1, n_2)$  and  $n_{(2)} = \min(n_1, n_2)$ . To better illustrate our intuition, we begin with the ‘‘ideal’’ case where there is no outlier in the data, i.e.,  $\mathcal{X} = \mathcal{L}_0$  and  $\mathcal{E}_0 = \mathbf{0}$ . Then the OR-TLRR problem degenerates to

$$\min_{\mathcal{Z}} \|\mathcal{Z}\|_* \quad \text{s.t.} \quad \mathcal{X} = \mathcal{X} *_{\mathbf{L}} \mathcal{Z}. \quad (8)$$

As shown in [Zhou et al., 2021, Yang et al., 2022], this problem has a unique solution given by  $\mathcal{Z}_* = \mathbf{V}_0 *_{\mathbf{L}} \mathbf{V}_0^H$ , which suggests that the solution of OR-TLRR identifies the row space of  $\mathcal{L}_0$  in this special case. When the data tensor is contaminated by outliers, we expect the row space of  $\mathcal{L}_0$  can still be exactly recovered. We first establish the following lemma, which states that the optimal solution to OR-TLRR always locates within the row space of raw tensor data  $\mathcal{X}$ .

**Lemma 1.** *For any optimal solution  $(\mathcal{Z}_*, \mathcal{E}_*)$  to the OR-TLRR problem (4), we have  $\mathcal{Z}_* \in \mathcal{P}_{\mathbf{V}_{\mathcal{X}}}^L$ .*

This lemma states that the column space of  $\mathcal{Z}_*$  is a subspace of  $\mathbf{V}_{\mathcal{X}}$ . Thus, in order for  $\mathcal{Z}_*$  to exactly recover  $\mathbf{V}_0$ , a necessary condition is that  $\mathbf{V}_0$  is a subspace of  $\mathbf{V}_{\mathcal{X}}$ , i.e.,  $\mathbf{V}_0 \in \mathcal{P}_{\mathbf{V}_{\mathcal{X}}}^L$ . Intuitively, if some outliers in  $\mathcal{E}_0$  lie exactly on the span of the tensor subspaces, then we can think of  $\mathcal{X}$  as containing more authentic samples than  $\mathcal{L}_0$  and this condition will be violated. To show how it can hold, we establish the following lemma:

**Lemma 2.** *If  $\text{Range}(\mathcal{L}_0)$  and  $\text{Range}(\mathcal{E}_0)$  are independent to each other, i.e.,  $\text{Range}(\mathcal{L}_0) \cap \text{Range}(\mathcal{E}_0) = \{\mathbf{0}\}$ , then  $\mathbf{V}_0 \in \mathcal{P}_{\mathbf{V}_{\mathcal{X}}}^L$ .*

Next, similar to TLRR [Zhou et al., 2021], exactly separating  $\mathcal{X}$  as the low-rank term  $\mathcal{L}_0$  plus the column-sparse term  $\mathcal{E}_0$  requires  $\mathcal{L}_0$  is not column-sparse and  $\mathcal{E}_0$  is not low-rank. To characterize this intuition, we need the following two mild conditions.

**Incoherence Condition on Low-Rank Term:** Let  $L$  be any invertible linear transform in (3) and it satisfies (2). For  $\mathcal{L} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ , assume that  $\text{rank}_t(\mathcal{L}) = r$  and it has the skinny t-SVD  $\mathcal{L} = \mathbf{U} *_{\mathbf{L}} \mathbf{S} *_{\mathbf{L}} \mathbf{V}^H$ . Then the tensor column-incoherence condition with parameter  $\mu$  is defined as

$$\mu \geq \frac{n_2 \tau}{r} \max_{j=1, \dots, n_2} \|\hat{\mathbf{V}}^H *_{\mathbf{L}} \mathring{\mathbf{e}}_j\|_F^2, \quad (9)$$

where  $\mathring{\mathbf{e}}_j$  is a tensor of size  $n_2 \times 1 \times n_3$  with the entries of the  $(j, 1)$ -th tube of  $L(\mathring{\mathbf{e}}_j)$  equaling 1 and the rest equaling 0. A small value of  $\mu$  implies the low-rank term  $\mathcal{L}$  is not column-sparse.

**Ambiguity Condition on Column Sparse Term:** Another identifiability issue arises if the outlier tensor is both column-sparse and low-rank. To avoid this pathological situation, we introduce an unambiguity condition on  $\mathcal{E}$ :

$$\|\mathcal{B}(\mathcal{E})\| \leq \log(n_2)/\theta \quad (10)$$

for some constant  $\theta \geq 4$ . In fact, condition (10) holds as long as the directions of the nonzero columns of  $\mathcal{E}$  scatter sufficiently randomly. Thus it guarantees

that the tensor  $\mathcal{E}$  cannot be low-rank when the column sparsity of  $\mathcal{E}$  is comparable to  $n_2$ . We have the following exact recovery guarantee for problem (4).

**Theorem 2.** *Suppose  $\text{Range}(\mathcal{L}_0) = \text{Range}(\mathcal{P}_{\Theta_0^\perp}(\mathcal{L}_0))$  and  $\mathcal{E}_0(:, j, :) \notin \text{Range}(\mathcal{L}_0)$  for  $j \in \Theta_0$ . Then any solution  $(\mathcal{V}_0 *_{\mathbf{L}} \mathcal{V}_0^H + \mathcal{H}, \mathcal{E}_0 - \mathcal{X} *_{\mathbf{L}} \mathcal{H})$  to (4) with  $\lambda = \frac{\theta}{4\sqrt{\log(n_{(1)})}\|\mathcal{X}\|}$  exactly recovers the row space of  $\mathcal{L}_0$  and the column support of  $\mathcal{E}_0$  with a probability at least  $1 - c_1 n_{(1)}^{-10}$ , where  $c_1$  is a positive constant, if the column support set  $\Theta_0$  is uniformly distributed among all sets of cardinality  $|\Theta_0|$  and*

$$\text{rank}_t(\mathcal{L}_0) \leq \frac{\rho_r n_2}{\mu n_1 n_3 \|\mathcal{X}\|^2} \text{ and } |\Theta_0| \leq \rho_s n_2, \quad (11)$$

where  $\rho_r$  and  $\rho_s$  are constants,  $\mathcal{X} *_{\mathbf{L}} (\mathcal{V}_0 *_{\mathbf{L}} \mathcal{V}_0^H + \mathcal{P}_{\Theta_0} \mathcal{P}_{\mathcal{V}_0}^L(\mathcal{H}))$  satisfies the column-incoherence condition (9) and  $\mathcal{E}_0 - \mathcal{X} *_{\mathbf{L}} \mathcal{P}_{\Theta_0} \mathcal{P}_{\mathcal{V}_0}^L(\mathcal{H})$  satisfies the unambiguity condition (10).

The incoherence and ambiguity conditions on  $\hat{\mathcal{L}} = \mathcal{X} *_{\mathbf{L}} \hat{\mathcal{Z}}$  and  $\hat{\mathcal{E}} = \mathcal{E}_0 - \mathcal{X} *_{\mathbf{L}} \mathcal{P}_{\Theta_0} \mathcal{P}_{\mathcal{V}_0}^L(\mathcal{H})$  is not surprising, where  $\hat{\mathcal{Z}} = \mathcal{V}_0 *_{\mathbf{L}} \mathcal{V}_0^H + \mathcal{P}_{\Theta_0} \mathcal{P}_{\mathcal{V}_0}^L(\mathcal{H})$ . In fact, the column space of  $\hat{\mathcal{Z}}$  is the same as the row space of  $\text{Range}(\mathcal{L}_0)$  and  $\hat{\mathcal{E}}$  has the same column support as that of  $\mathcal{E}_0$ . Also, notice that  $\mathcal{X} = \mathcal{X} *_{\mathbf{L}} \hat{\mathcal{Z}} + \hat{\mathcal{E}}$ . So we can consider  $\hat{\mathcal{L}}$  and  $\hat{\mathcal{E}}$  as the underlying low-rank and sparse terms, and we assume incoherence and ambiguity conditions on them instead of  $\mathcal{L}_0$  and  $\mathcal{E}_0$ . The above results demonstrate that with high probability OR-TLRR can exactly recover the row space of  $\mathcal{P}_{\Theta_0}(\mathcal{L}_0)$  and the support set  $\Theta_0$  of  $\mathcal{E}_0$ .

## 5 OR-TLRR WITH MISSING ENTRIES

Let us now consider the case where some entries of the data tensor  $\mathcal{X}$  are missing. Specifically, let  $\mathcal{W} \in \{0, 1\}^{n_1 \times n_2 \times n_3}$  be a binary matrix such that  $\mathcal{W}_{i,j,k} = 1$  if  $\mathcal{X}_{i,j,k}$  is observed and  $\mathcal{W}_{i,j,k} = 0$  otherwise. The locations of the observed entries for the entire data can be indexed by the set  $\Omega = \{(i, j, k) : \mathcal{W}_{i,j,k} = 1\}$ . Given the observed entries of  $\mathcal{X}$ ,  $\{\mathcal{X}_{i,j,k}\}_{(i,j,k) \in \Omega}$ , our goal is to segment the tensor columns of  $\mathcal{X}$  into their corresponding tensor subspaces in the presence of outliers. Since we do not have access to the complete data, it is impossible to directly solve (4). One approach is to fill the missing entries in  $\mathcal{X}$  with 0s, i.e., to replace  $\mathcal{X}$  by  $\mathcal{X}_{\text{miss}} = \mathcal{W} \odot \mathcal{X}$  and then solve (4). As such, the constraint in (4) will become  $\mathcal{X}_{\text{miss}} = \mathcal{X}_{\text{miss}} *_{\mathbf{L}} \mathcal{Z} + \mathcal{E}$  and we essentially enforce the  $(i, j, k)$ -th entry of the reconstructed tensor  $\mathcal{X}_{\text{miss}} *_{\mathbf{L}} \mathcal{Z}$  to be close to zero whenever  $\mathcal{W}_{i,j,k} = 0$ , while this term should be not penalized because we do not observe  $\mathcal{X}_{i,j,k}$ . Followed by [Yang et al., 2015], we propose to incorporate the

Table 2: Exact recovery on random problems of varying sizes. The Discrete Fourier Transform (DFT) is used as the invertible linear transform  $L$ .

$\rho = 0.2, r_\ell = 0.1n_1, \lambda = 4/(\sqrt{\log(n_{(1)})}\ \mathcal{X}\ )$				
$n_1$	$\text{rank}_t(\mathcal{P}_{\Theta_0^\perp}(\tilde{\mathcal{X}}))$	$\frac{\ \mathcal{P}_{\mathcal{V}_0}^L - \mathcal{P}_{\tilde{\mathcal{U}}}\ _F}{\ \mathcal{P}_{\mathcal{V}_0}^L\ _F}$	$\frac{\ \mathcal{P}_{\Theta_0^\perp}(\mathcal{L}_0) - \mathcal{P}_{\Theta_0^\perp}(\tilde{\mathcal{X}})\ _F}{\ \mathcal{P}_{\Theta_0^\perp}(\mathcal{L}_0)\ _F}$	$\text{dist}(\Theta_0, \tilde{\Theta})$
60	30	2.5664e-15	2.7575e-09	0
100	50	2.7290e-15	4.0480e-09	0
150	75	3.0159e-15	1.0702e-08	0
$\rho = 0.4, r_\ell = 0.1n_1, \lambda = 4/(\sqrt{\log(n_{(1)})}\ \mathcal{X}\ )$				
$n_1$	$\text{rank}_t(\mathcal{P}_{\Theta_0^\perp}(\tilde{\mathcal{X}}))$	$\frac{\ \mathcal{P}_{\mathcal{V}_0}^L - \mathcal{P}_{\tilde{\mathcal{U}}}\ _F}{\ \mathcal{P}_{\mathcal{V}_0}^L\ _F}$	$\frac{\ \mathcal{P}_{\Theta_0^\perp}(\mathcal{L}_0) - \mathcal{P}_{\Theta_0^\perp}(\tilde{\mathcal{X}})\ _F}{\ \mathcal{P}_{\Theta_0^\perp}(\mathcal{L}_0)\ _F}$	$\text{dist}(\Theta_0, \tilde{\Theta})$
60	30	2.5826e-15	1.5814e-27	0
100	50	2.8084e-15	2.7471e-27	0
150	75	3.0104e-15	5.5133e-27	0

constraint  $\mathcal{P}_\Omega(\mathcal{X}_{\text{miss}}) = \mathcal{P}_\Omega(\mathcal{X}_{\text{miss}} *_{\mathbf{L}} \mathcal{Z} + \mathcal{E})$  into (4) and solve the following problem:

$$\begin{aligned} \min_{\mathcal{Z}, \mathcal{E}} \quad & \|\mathcal{Z}\|_* + \lambda \|\mathcal{P}_\Omega(\mathcal{E})\|_{2,1} \\ \text{s.t.} \quad & \mathcal{P}_\Omega(\mathcal{X}_{\text{miss}}) = \mathcal{P}_\Omega(\mathcal{X}_{\text{miss}} *_{\mathbf{L}} \mathcal{Z} + \mathcal{E}), \end{aligned} \quad (12)$$

where  $\mathcal{P}_\Omega : \mathbb{R}^{n_1 \times n_2 \times n_3} \rightarrow \mathbb{R}^{n_1 \times n_2 \times n_3}$  is the orthogonal projector onto the span of tensors vanishing outside  $\Omega$  so that the  $(i, j, k)$ -th component of  $\mathcal{P}_\Omega(\mathcal{A})$  is equal to  $\mathcal{A}_{i,j,k}$  if  $(i, j, k) \in \Omega$  and zero otherwise. We further assume that  $\mathcal{U}_\mathcal{X} *_{\mathbf{L}} \mathcal{S}_\mathcal{X} *_{\mathbf{L}} \mathcal{V}_\mathcal{X}^H$  is the skinny t-SVD of  $\mathcal{X}_{\text{miss}}$  and  $r_\mathcal{X} = \text{rank}_t(\mathcal{X}_{\text{miss}})$ , (12) can then be transformed into a simpler problem as follows:

$$\begin{aligned} \min_{\mathcal{Z}', \mathcal{E}} \quad & \|\mathcal{Z}'\|_* + \lambda \|\mathcal{P}_\Omega(\mathcal{E})\|_{2,1} \\ \text{s.t.} \quad & \mathcal{P}_\Omega(\mathcal{X}_{\text{miss}}) = \mathcal{P}_\Omega(\mathcal{D} *_{\mathbf{L}} \mathcal{Z}' + \mathcal{E}), \end{aligned} \quad (13)$$

where  $\mathcal{D} = \mathcal{U}_\mathcal{X} *_{\mathbf{L}} \mathcal{S}_\mathcal{X}$ . We refer the reader to the supplementary material for deduction details. We dub this approach *Outlier-Robust Tensor LRR by Entry-Wise Zero-Fill* (OR-TLRR-EWZF). Note that we can also robustify TLRR [Zhou et al., 2021] by replacing  $\lambda \|\mathcal{P}_\Omega(\mathcal{E})\|_{2,1}$  in (12) with  $\lambda \|\mathcal{P}_\Omega(\mathcal{E})\|_1$ , and we call the resulting algorithm TLRR-EWZF in our experiments.

## 6 EXPERIMENTS

In this section, we perform extensive experiments on both synthetic and real data to demonstrate the usefulness of the proposed algorithms. All experiments are conducted using Matlab R2021b on an AMD Ryzen 9 5950X 3.40GHz CPU with 64GB RAM.

### 6.1 Synthetic Experiments

Here, we verify the recovery guarantee in Theorem 2 on randomly generated tensors. Three invertible linear transforms  $L$  are adopted: (a) Discrete Fourier Transform (DFT); (b) Discrete Cosine Transform (DCT); (c) Random Orthogonal Matrix (ROM). We consider  $c = 5$  tensor subspaces and generate the random

Table 3: Exact recovery on random problems of varying sizes. The Discrete Cosine Transform (DCT) is used as the invertible linear transform  $L$ .

$$\rho = 0.2, r_\ell = 0.1n_1, \lambda = 40/(\sqrt{\log(n_{(1)})}\|\mathcal{X}\|)$$

$n_1$	$\text{rank}_t(\mathcal{P}_{\Theta_0^\perp}(\tilde{\mathcal{X}}))$	$\frac{\ \mathcal{P}_{\mathcal{V}_0^L}^L - \mathcal{P}_{\tilde{\mathcal{U}}}^L\ _F}{\ \mathcal{P}_{\mathcal{V}_0^L}^L\ _F}$	$\frac{\ \mathcal{P}_{\Theta_0^\perp}(\mathcal{L}_0) - \mathcal{P}_{\Theta_0^\perp}(\tilde{\mathcal{X}})\ _F}{\ \mathcal{P}_{\Theta_0^\perp}(\mathcal{L}_0)\ _F}$	$\text{dist}(\Theta_0, \tilde{\Theta})$
60	30	2.4522e-15	1.1794e-05	0
100	50	2.5648e-15	5.3367e-06	0
150	75	2.9294e-15	5.1267e-06	0

$$\rho = 0.4, r_\ell = 0.1n_1, \lambda = 40/(\sqrt{\log(n_{(1)})}\|\mathcal{X}\|)$$

$n_1$	$\text{rank}_t(\mathcal{P}_{\Theta_0^\perp}(\tilde{\mathcal{X}}))$	$\frac{\ \mathcal{P}_{\mathcal{V}_0^L}^L - \mathcal{P}_{\tilde{\mathcal{U}}}^L\ _F}{\ \mathcal{P}_{\mathcal{V}_0^L}^L\ _F}$	$\frac{\ \mathcal{P}_{\Theta_0^\perp}(\mathcal{L}_0) - \mathcal{P}_{\Theta_0^\perp}(\tilde{\mathcal{X}})\ _F}{\ \mathcal{P}_{\Theta_0^\perp}(\mathcal{L}_0)\ _F}$	$\text{dist}(\Theta_0, \tilde{\Theta})$
60	30	2.5582e-15	2.8832e-13	0
100	50	2.6576e-15	8.6553e-15	0
150	75	2.9046e-15	1.1909e-15	0

Table 4: Exact recovery on random problems of varying sizes. The Random Orthogonal Matrix (ROM) is used as the invertible linear transform  $L$ .

$$\rho = 0.2, r_\ell = 0.1n_1, \lambda = 40/(\sqrt{\log(n_{(1)})}\|\mathcal{X}\|)$$

$n_1$	$\text{rank}_t(\mathcal{P}_{\Theta_0^\perp}(\tilde{\mathcal{X}}))$	$\frac{\ \mathcal{P}_{\mathcal{V}_0^L}^L - \mathcal{P}_{\tilde{\mathcal{U}}}^L\ _F}{\ \mathcal{P}_{\mathcal{V}_0^L}^L\ _F}$	$\frac{\ \mathcal{P}_{\Theta_0^\perp}(\mathcal{L}_0) - \mathcal{P}_{\Theta_0^\perp}(\tilde{\mathcal{X}})\ _F}{\ \mathcal{P}_{\Theta_0^\perp}(\mathcal{L}_0)\ _F}$	$\text{dist}(\Theta_0, \tilde{\Theta})$
60	30	3.7444e-14	1.0834e-05	0
100	50	4.2402e-14	6.0406e-06	0
150	75	4.0723e-14	5.6253e-06	0

$$\rho = 0.4, r_\ell = 0.1n_1, \lambda = 40/(\sqrt{\log(n_{(1)})}\|\mathcal{X}\|)$$

$n_1$	$\text{rank}_t(\mathcal{P}_{\Theta_0^\perp}(\tilde{\mathcal{X}}))$	$\frac{\ \mathcal{P}_{\mathcal{V}_0^L}^L - \mathcal{P}_{\tilde{\mathcal{U}}}^L\ _F}{\ \mathcal{P}_{\mathcal{V}_0^L}^L\ _F}$	$\frac{\ \mathcal{P}_{\Theta_0^\perp}(\mathcal{L}_0) - \mathcal{P}_{\Theta_0^\perp}(\tilde{\mathcal{X}})\ _F}{\ \mathcal{P}_{\Theta_0^\perp}(\mathcal{L}_0)\ _F}$	$\text{dist}(\Theta_0, \tilde{\Theta})$
60	30	3.4499e-14	1.2335e-07	0
100	50	3.9048e-14	3.4851e-15	0
150	75	3.7525e-14	1.2012e-15	0

tensor  $\mathcal{Q} = [\mathcal{Q}_1, \dots, \mathcal{Q}_5] \in \mathbb{R}^{n_1 \times n_2 \times n_3}$  with  $\mathcal{Q}_\ell = \mathcal{A}_\ell *_{\mathcal{L}} \mathcal{B}_\ell$ , where the entries of  $\mathcal{A}_\ell \in \mathbb{R}^{n_1 \times r_\ell \times n_3}$  and  $\mathcal{B}_\ell \in \mathbb{R}^{r_\ell \times s_\ell \times n_3}$  are independently sampled from  $\mathcal{N}(0, 1/n_1)$  distribution. We simply set  $s_\ell = n_1$ , hence  $n_2 = 5n_1$ . Then the lateral slices are independently sampled with probability  $\rho$  as outliers, and the entries of these selected columns are randomly sampled from i.i.d.  $\mathcal{N}(0, \zeta/(n_1 n_3))$  distribution, where  $\zeta$  is the averaged magnitude of the samples in  $\mathcal{Q}$ . In this way, the samples and outliers approximately have the same magnitude. The remaining entries of  $\mathcal{E}_0$  are 0s. We denote the column support of  $\mathcal{E}_0$  as  $\Theta_0$ . Finally, the observed tensor  $\mathcal{X}$  is generated by  $\mathcal{X} = \mathcal{L}_0 + \mathcal{E}_0$ , where  $\mathcal{L}_0$  is obtained by setting the lateral slices of  $\mathcal{Q}$  corresponding to nonzero columns of  $\mathcal{E}_0$  to be 0s.

To verify that OR-TLRR can perform well for various tensor sizes, we set  $n_3 = 100$ ,  $n_1 = [60, 100, 150]$ ,  $r_\ell = 0.1n_1$ , and  $\rho = [0.2, 0.4]$ . We run every experiment for 20 times and the results reported here correspond to the average of these 20 random trials. We set  $\lambda = \alpha/(\sqrt{\log(n_{(1)})}\|\mathcal{X}\|)$  and empirically choose  $\alpha$  from  $[1, 2, 4, 8, 10]$  for DFT and from  $[5, 10, 20, 30, 40, 50]$  for DCT and ROM. Note that  $\tilde{\mathcal{X}}$  denotes the recovered tensor,  $\tilde{\mathcal{U}}$  is the column space of  $\tilde{\mathcal{Z}}$ , and the recovered outlier support set of  $\tilde{\mathcal{E}}$  is denoted as  $\tilde{\Theta}$ . Table 2 - Table 4 respectively give the recovery results for the three different choices of the linear transforms  $L$ . The recovered tubal rank of  $\mathcal{P}_{\Theta_0^\perp}(\mathcal{X})$  is exactly equal to  $5r_\ell$  and the relative errors  $\|\mathcal{P}_{\mathcal{V}_0^L}^L - \mathcal{P}_{\tilde{\mathcal{U}}}^L\|_F / \|\mathcal{P}_{\mathcal{V}_0^L}^L\|_F$  and

Table 5: Description of datasets.

Dataset	#Class	#Total image	Size	Type
ORL	40	400	32 × 32	Face
COIL20	20	1440	32 × 32	Object
Umist	20	575	56 × 46	Face
FRDUE	152	3040	25 × 22	Face
USPS	10	9298	16 × 16	Handwritten

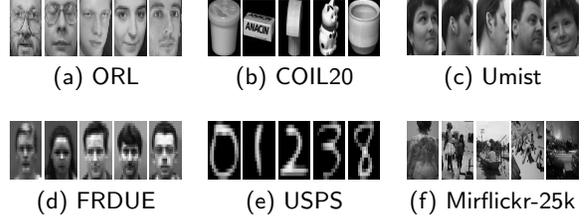


Figure 1: Sample images from the datasets.

$\|\mathcal{P}_{\Theta_0^\perp}(\mathcal{L}_0) - \mathcal{P}_{\Theta_0^\perp}(\tilde{\mathcal{X}})\|_F / \|\mathcal{P}_{\Theta_0^\perp}(\mathcal{L}_0)\|_F$  are very small (less than  $10^{-4}$ ). The Hamming distance  $\text{dist}(\Theta_0, \tilde{\Theta})$  between  $\Theta_0$  and  $\tilde{\Theta}$  is always 0. Thus, the recovery guarantee claimed in Theorem 2 has been fully verified by these numerical results. We also investigate the OR-TLRR-EWZF performance with missing data, where we set the percentage of missing entries  $\delta$  to be 10% and 20%. We implement OR-TLRR-EWZF with  $\lambda = 2/(\sqrt{\log(n_{(1)})}\|\mathcal{X}_{\text{miss}}\|)$  for DFT and  $\lambda = 30/(\sqrt{\log(n_{(1)})}\|\mathcal{X}_{\text{miss}}\|)$  for DCT and ROM. The resulting Hamming distance  $\text{dist}(\Theta_0, \tilde{\Theta})$  is again always 0, which suggests that our algorithm can detect the outliers successfully in this setting.

## 6.2 Real-World Applications

In this subsection, the proposed algorithms are based on two different linear transforms for all experiments, i.e., DFT and DCT. The corresponding methods of OR-TLRR and OR-TLRR-EWZF are called OR-TLRR-DFT/OR-TLRR-DCT and OR-TLRR-EWZF-DFT/OR-TLRR-EWZF-DCT, respectively for short. We fix  $\lambda = 1/(\sqrt{\log(n_{(1)})}\|\mathcal{X}\|)$  for OR-TLRR-DFT and  $\lambda = 1/(\sqrt{\log(n_{(1)})}\|\mathcal{X}_{\text{miss}}\|)$  for OR-TLRR-EWZF-DFT, and set  $\lambda = 100/(\sqrt{\log(n_{(1)})}\|\mathcal{X}\|)$  for OR-TLRR-DCT and  $\lambda = 100/(\sqrt{\log(n_{(1)})}\|\mathcal{X}_{\text{miss}}\|)$  for OR-TLRR-EWZF-DCT. Table 5 gives the brief introduction of five testing databases, including ORL<sup>1</sup>, COIL20<sup>2</sup>, Umist<sup>3</sup>, FRDUE<sup>4</sup>, and USPS [Hull, 1994].

### 6.2.1 Application to Outlier Detection

Here we evaluate the effectiveness of OR-TLRR on outlier detection and clustering tasks. For

<sup>1</sup><http://cam-orl.co.uk/facedatabase.html>

<sup>2</sup><https://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php>

<sup>3</sup><https://www.visioneng.org.uk/datasets/>

<sup>4</sup><https://bamdevmishra.in/codes/tensorcompletion/>

Table 6: AUC of outlier detection and clustering results (ACC, NMI and PUR) on ORL-MIRFLICKR-25k and COIL20-MIRFLICKR-25k datasets for complete data experiments.

Datasets (# Outlier)	ORL-MIR-25k (100)				COIL20-MIR-25k (100)				COIL20-MIR-25k (200)			
	AUC	ACC	NMI	PUR	AUC	ACC	NMI	PUR	AUC	ACC	NMI	PUR
Methods												
R-PCA	<b>0.9734</b>	0.4972	0.7115	0.5327	<b>0.8732</b>	0.6171	0.7301	0.6606	0.8444	0.6253	0.7215	0.6468
OR-PCA	0.9733	0.4972	0.7073	0.5328	0.8636	0.6228	0.7311	0.6663	0.8368	0.6130	0.7143	0.6510
LRR	0.7448	0.5058	0.6843	0.5360	0.8630	0.5002	0.6358	0.5956	<b>0.8836</b>	0.5252	0.6489	0.6114
TRPCA-DFT	0.9548	0.5549	0.7271	0.5806	0.7687	0.5679	0.7056	0.6441	0.7381	0.6087	0.7171	0.6581
TRPCA-DCT	0.9586	0.5522	0.7237	0.5779	0.7740	0.5344	0.6912	0.6308	0.7430	0.5740	0.7011	0.6377
OR-TPCA	0.7297	0.3851	0.5955	0.4098	0.0267	0.5675	0.6610	0.6027	0.0100	0.5750	0.6671	0.6085
TLRR-DFT	0.9550	0.5094	0.6812	0.5373	0.8219	0.5635	0.6479	0.6043	0.8049	0.5682	0.6454	0.6050
TLRR-DCT	0.9573	0.5488	0.7193	0.5753	0.7871	0.6222	0.7127	0.6647	0.7688	0.6624	0.7356	0.6947
OR-TLRR-DFT	0.9401	<b>0.5915</b>	<b>0.7525</b>	<b>0.6184</b>	0.7910	<b>0.6700</b>	<b>0.7488</b>	<b>0.6967</b>	0.7666	<b>0.6725</b>	<b>0.7471</b>	<b>0.6966</b>
OR-TLRR-DCT	0.9460	0.5691	0.7444	0.5969	0.7643	0.5552	0.6855	0.6245	0.7384	0.6087	0.7165	0.6583

Table 7: AUC of outlier detection and clustering results (ACC, NMI and PUR) on ORL-MIRFLICKR-25k dataset for missing data experiments.

Methods	$\delta = 10\%$				$\delta = 20\%$			
	AUC	ACC	NMI	PUR	AUC	ACC	NMI	PUR
TNN+TRPCA-DFT	0.9552	0.5551	0.7260	0.5806	0.9535	0.5493	0.7224	0.5753
TNN+TRPCA-DCT	<b>0.9581</b>	0.5513	0.7241	0.5774	<b>0.9583</b>	0.5529	0.7244	0.5786
TNN+OR-TPCA-DFT	0.6970	0.3772	0.5871	0.4025	0.6752	0.3735	0.5820	0.3976
TNN+TLRR-DFT	0.9537	0.5077	0.6766	0.5336	0.9570	0.4847	0.6587	0.5105
TNN+TLRR-DCT	0.9580	0.5420	0.7136	0.5690	0.9578	0.5504	0.7175	0.5773
TNN+OR-TLRR-DFT	0.9384	<b>0.5853</b>	<b>0.7475</b>	<b>0.6108</b>	0.9376	<b>0.5843</b>	<b>0.7481</b>	<b>0.6132</b>
TNN+OR-TLRR-DCT	0.9445	0.5722	0.7441	0.6009	0.9450	0.5674	0.7409	0.5949
TLRR-EWZF-DFT	0.9549	0.5088	0.6760	0.5341	0.9544	0.4892	0.6590	0.5152
TLRR-EWZF-DCT	0.9568	0.5398	0.7128	0.5687	0.9568	0.5398	0.7128	0.5674
OR-TLRR-EWZF-DFT	0.9391	0.5813	0.7449	0.6089	0.9407	0.5802	0.7417	0.6080
OR-TLRR-EWZF-DCT	0.9456	0.5608	0.7382	0.5899	0.9468	0.5631	0.7331	0.5898

the complete data experiments, we compare OR-TLRR-DFT/OR-TLRR-DCT with other state-of-the-art low-rank matrix/tensor factorization methods including R-PCA [Candès et al., 2011], OR-PCA [Zhang et al., 2015], LRR [Liu et al., 2013a], TRPCA-DFT [Lu et al., 2020], TRPCA-DCT [Lu, 2021], OR-TPCA [Zhou and Feng, 2017], TLRR-DFT [Zhou et al., 2021] and TLRR-DCT [Yang et al., 2022]. For TRPCA/OR-TPCA, let the estimated clean “inlier” data be  $\mathcal{L}$ . We first compute a matrix  $\mathbf{L}_0 = \sum_{k=1}^{n_3} \mathbf{L}^{(k)}$  and perform SVD on  $\mathbf{L}_0 = \mathbf{U}_0 \mathbf{S}_0 \mathbf{V}_0^T$ . Then we apply spectral clustering on  $\mathbf{V}_0 \mathbf{V}_0^T$  to obtain the result. For the case of data having missing entries, we compare OR-TLRR-EWZF/TLRR-EWZF with a baseline that first completes the unknown entries of  $\mathcal{X}$  using tensor nuclear norm based tensor completion model [Lu et al., 2018, Lu et al., 2019b], and then do clustering using TRPCA [Lu et al., 2020, Lu, 2021], OR-TPCA [Zhou and Feng, 2017], TLRR [Zhou et al., 2021, Yang et al., 2022], and OR-TLRR with the same linear transform that is used in the tensor completion step. The percentage of missing entries  $\delta$  is again set to be 10% and 20%. Unless otherwise stated, all regularization parameters involved in the competing methods were either tuned or selected as suggested in the respective papers.

We create two datasets by combining ORL/COIL20 with the MIRFLICKR-25k dataset (containing 25,000 images) [Huiskes and Lew, 2008], which we call ORL-MIRFLICKR-25k and COIL20-MIRFLICKR-25k, respectively. For ORL-MIRFLICKR-25k, we randomly select 100 images from MIRFLICKR-25k as outliers and the random selection is repeated 20 times. Fig. 1 (a) and Fig. 1 (f) show some examples of ORL and MIRFLICKR-25k, respectively. For COIL20-MIRFLICKR-25k, the outliers consist of 100 or 200 images selected from MIRFLICKR-25k. Theorem 2 implies the optimal solution  $\mathcal{E}$  can help detect outliers in the data. Since all the methods have the sparse component  $\mathcal{E}$ , in this paper, we use  $k$ -means to cluster all the  $\|\mathcal{E}(:, j, :)\|_F^2$ ’s into two classes (outliers vs. non-outliers) for outlier detection. The performance of outlier detection is evaluated by AUC. While investigating the segmentation performance, we first remove the outliers detected by  $k$ -means (some normal samples may be removed, and these samples are given wrong labels) and apply Ncut [Shi and Malik, 2000] on the remaining samples (possibly include undetected outliers) for clustering. We evaluate the clustering performance of the normal samples using three metrics, including ACC, NMI [Vinh et al., 2010] and PUR [Manning et al., 2010].

Table 6 summarizes the results for the complete

Table 8: Clustering results (ACC, NMI and PUR) on Umist and FRDUE datasets for complete data experiments.

Dataset	Metric	R-PCA	OR-PCA	LRR	SSC	SSC-OMP	TRPCA-DFT	TRPCA-DCT	OR-TPCA	TLRR-DFT	TLRR-DCT	OR-TLRR-DFT	OR-TLRR-DCT
Umist	ACC	0.4299	0.4155	0.3515	0.4972	0.4030	0.5774	0.5741	0.3454	0.5452	0.5007	<b>0.6117</b>	0.5125
	NMI	0.5831	0.5786	0.4718	0.6818	0.5633	0.7282	0.7331	0.3930	0.6648	0.6573	<b>0.7536</b>	0.6756
	PUR	0.4758	0.4661	0.4167	0.5897	0.5078	0.6374	0.6478	0.3871	0.6226	0.5743	<b>0.6790</b>	0.5652
FRDUE	ACC	0.6784	0.6760	0.7765	0.8200	0.4913	0.7058	0.7011	0.6807	0.8192	0.8109	<b>0.8525</b>	0.8231
	NMI	0.8624	0.8619	0.9181	0.9399	0.7378	0.8804	0.8785	0.8757	0.9365	0.9321	<b>0.9549</b>	0.9403
	PUR	0.7162	0.7131	0.8151	0.8649	0.5319	0.7435	0.7384	0.7243	0.8548	0.8403	<b>0.8832</b>	0.8578

Table 9: Clustering results (ACC, NMI and PUR) on Umist and FRDUE datasets for missing data experiments.

Methods	Umist						FRDUE					
	$\delta = 10\%$			$\delta = 20\%$			$\delta = 10\%$			$\delta = 20\%$		
	ACC	NMI	PUR									
TNN+TRPCA-DFT	0.5715	0.7261	0.6319	0.5737	0.7267	0.6333	0.7134	0.8828	0.7496	0.7274	0.8883	0.7626
TNN+TRPCA-DCT	0.5730	0.7297	0.6379	0.5743	0.7308	0.6390	0.7153	0.8825	0.7515	0.7237	0.8865	0.7590
TNN+OR-TPCA-DFT	0.3524	0.4037	0.4014	0.3495	0.3995	0.3951	0.7285	0.8941	0.7653	0.7467	0.9006	0.7817
TNN+TLRR-DFT	0.5298	0.6512	0.6024	0.5261	0.6529	0.6064	0.8389	0.9460	0.8701	0.8476	0.9504	0.8775
TNN+TLRR-DCT	0.5020	0.6557	0.5751	0.4982	0.6507	0.5719	0.8195	0.9388	0.8525	0.8274	0.9439	0.8597
TNN+OR-TLRR-DFT	<b>0.6017</b>	<b>0.7477</b>	<b>0.6707</b>	<b>0.6015</b>	<b>0.7476</b>	<b>0.6707</b>	0.8577	0.9577	0.8873	0.8682	0.9625	0.8967
TNN+OR-TLRR-DCT	0.5048	0.6749	0.5611	0.5015	0.6722	0.5573	0.8339	0.9458	0.8650	0.8377	0.9487	0.8696
TLRR-EWZF-DFT	0.5704	0.7249	0.6461	0.5512	0.7112	0.6289	0.8616	0.9568	0.8928	0.8726	0.9624	0.9012
TLRR-EWZF-DCT	0.5071	0.6731	0.5795	0.5030	0.6776	0.5713	0.8520	0.9570	0.8831	0.8546	0.9573	0.8842
OR-TLRR-EWZF-DFT	0.5705	0.7215	0.6356	0.5554	0.7034	0.6203	<b>0.8641</b>	<b>0.9626</b>	<b>0.8959</b>	<b>0.8893</b>	<b>0.9752</b>	<b>0.9160</b>
OR-TLRR-EWZF-DCT	0.5079	0.6740	0.5639	0.5145	0.6787	0.5716	0.8496	0.9566	0.8820	0.8555	0.9579	0.8858

data experiments. From these results, we can find that matrix-based algorithms seem to work better than tensor-based ones in terms of outlier detection. OR-TLRR-DFT always achieves the best clustering performance. On the widely used ACC metric, OR-TLRR-DFT respectively improves by 3.9%, 7.6% and 1.5% over the runner-up on the three testing cases (left-right). Such improvement can be attributed to the reason that (1) it takes advantage of the multi-dimensional structure of tensor data; (2) it uses the  $\ell_{2,1}$  norm which can better depict outliers than the  $\ell_1$  norm. From Table 7 and Table 10 (in the supplementary material), we observe that TNN+OR-TLRR-DFT outperforms all other methods in terms of clustering on the ORL-MIRFLICKR-25k dataset, and it has very marginal improvement over the second best, OR-TLRR-EWZF-DFT. As for the COIL20-MIRFLICKR-25k dataset, OR-TLRR-EWZF-DCT consistently achieves the best clustering performance in ACC, NMI and PUR. This may be explained because the tensor completion step fails as the data tensor is not low-rank.

## 6.2.2 Application to Image Clustering

Here, we conduct experiments on Umist and FRDUE datasets for face clustering. Due to shadows and facial expressions on face images displayed in Fig. 1 (c) and Fig. 1 (d), it can be argued that the  $\ell_{2,1}$  norm can characterize the noise better than the  $\ell_1$  norm. Table 8 presents the clustering results for all the methods. As can be seen from this table, OR-TLRR-DFT shows noticeable improvement in all the metrics compared to

other algorithms. The accuracy outperforms the second best by 5.9% and 3.6% on Umist and FRDUE datasets, respectively. Similar to previous results, we observe from Table 9 that TNN+OR-TLRR-DFT performs the best on Umist dataset and OR-TLRR-EWZF-DFT outperforms other methods on FRDUE dataset. Understanding the root cause behind this phenomenon is the subject of future research. We also examine the robustness of our methods for dealing with sample-specific corruptions, and we include these results in the supplementary material.

## 7 CONCLUSION

In this work, we proposed an outlier-robust tensor low-rank representation method for tensor data clustering in the presence of outliers. We proved that OR-TLRR can recover the row space of the authentic data and identify the outliers with high probability. We further extended the proposed method to the missing data scenario. Extensive numerical results on synthetic and real data demonstrated the effectiveness of our algorithms. In this work, we directly use the raw tensor data as the dictionary. While the performance and the corresponding theoretical guarantee of other possible options for the dictionary, e.g., apply TRPCA on the raw data and use the estimation of the clean data as the dictionary, will be investigated in future work. Our future work also includes learning the optimal linear transform for different types of data.

## References

- [Cai et al., 2010] Cai, J.-F., Candès, E. J., and Shen, Z. (2010). A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982.
- [Candès et al., 2011] Candès, E. J., Li, X., Ma, Y., and Wright, J. (2011). Robust principal component analysis? *Journal of the ACM*, 58(3):1–37.
- [Donoho, 1995] Donoho, D. L. (1995). De-noising by soft-thresholding. *IEEE Transactions on Information Theory*, 41(3):613–627.
- [Elhamifar and Vidal, 2013] Elhamifar, E. and Vidal, R. (2013). Sparse subspace clustering: Algorithm, theory, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2765–2781.
- [Eriksson et al., 2012] Eriksson, B., Balzano, L., and Nowak, R. (2012). High-rank matrix completion. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 373–381.
- [Fu et al., 2016] Fu, Y., Gao, J., Tien, D., Lin, Z., and Hong, X. (2016). Tensor LRR and sparse coding-based subspace clustering. *IEEE Transactions on Neural Networks and Learning Systems*, 27(10):2120–2133.
- [Huiskes and Lew, 2008] Huiskes, M. J. and Lew, M. S. (2008). The MIR flickr retrieval evaluation. In *Proceedings of the ACM International Conference on Multimedia Information Retrieval*, pages 39–43.
- [Hull, 1994] Hull, J. J. (1994). A database for handwritten text recognition research. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5(16):550–554.
- [Ji et al., 2017] Ji, P., Zhang, T., Li, H., Salzmann, M., and Reid, I. (2017). Deep subspace clustering networks. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, volume 30, pages 24–33.
- [Ji et al., 2019] Ji, X., Vedaldi, A., and Henriques, J. (2019). Invariant information clustering for unsupervised image classification and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision*, pages 9865–9874.
- [Jiang et al., 2020] Jiang, T.-X., Ng, M. K., Zhao, X.-L., and Huang, T.-Z. (2020). Framelet representation of tensor nuclear norm for third-order tensor completion. *IEEE Transactions on Image Processing*, 29:7233–7244.
- [Jiang et al., 2023] Jiang, T.-X., Zhao, X.-L., Zhang, H., and Ng, M. K. (2023). Dictionary learning with low-rank coding coefficients for tensor completion. *IEEE Transactions on Neural Networks and Learning Systems*, 34(2):932–946.
- [Kernfeld et al., 2014] Kernfeld, E., Aeron, S., and Kilmer, M. (2014). Clustering multi-way data: A novel algebraic approach. *arXiv preprint*.
- [Kernfeld et al., 2015] Kernfeld, E., Kilmer, M., and Aeron, S. (2015). Tensor-tensor products with invertible linear transforms. *Linear Algebra and its Applications*, 485:545–570.
- [Kilmer et al., 2013] Kilmer, M. E., Braman, K., Hao, N., and Hoover, R. C. (2013). Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging. *SIAM Journal on Matrix Analysis and Applications*, 34(1):148–172.
- [Kilmer and Martin, 2011] Kilmer, M. E. and Martin, C. D. (2011). Factorization strategies for third-order tensors. *Linear Algebra and its Applications*, 435(3):641–658.
- [Kolda and Bader, 2009] Kolda, T. G. and Bader, B. W. (2009). Tensor decompositions and applications. *SIAM Review*, 51(3):455–500.
- [Kong et al., 2021] Kong, H., Lu, C., and Lin, Z. (2021). Tensor Q-rank: new data dependent definition of tensor rank. *Machine Learning*, 110(7):1867–1900.
- [Li et al., 2021] Li, Y., Hu, P., Liu, Z., Peng, D., Zhou, J. T., and Peng, X. (2021). Contrastive clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 8547–8555.
- [Lin et al., 2011] Lin, Z., Liu, R., and Su, Z. (2011). Linearized alternating direction method with adaptive penalty for low-rank representation. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 612–620.
- [Liu et al., 2013a] Liu, G., Lin, Z., Yan, S., Sun, J., Yu, Y., and Ma, Y. (2013a). Robust recovery of subspace structures by low-rank representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):171–184.
- [Liu et al., 2012] Liu, G., Xu, H., and Yan, S. (2012). Exact subspace segmentation and outlier detection by low-rank representation. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 703–711.

- [Liu and Yan, 2011] Liu, G. and Yan, S. (2011). Latent low-rank representation for subspace segmentation and feature extraction. In *Proceedings of the IEEE Conference on Computer Vision*, pages 1615–1622.
- [Liu et al., 2013b] Liu, J., Musialski, P., Wonka, P., and Ye, J. (2013b). Tensor completion for estimating missing values in visual data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):208–220.
- [Lu, 2021] Lu, C. (2021). Transforms based tensor robust PCA: Corrupted low-rank tensors recovery via convex optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision*, pages 1145–1152.
- [Lu et al., 2020] Lu, C., Feng, J., Chen, Y., Liu, W., Lin, Z., and Yan, S. (2020). Tensor robust principal component analysis with a new tensor nuclear norm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(4):925–938.
- [Lu et al., 2019a] Lu, C., Feng, J., Lin, Z., Mei, T., and Yan, S. (2019a). Subspace clustering by block diagonal representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2):487–501.
- [Lu et al., 2018] Lu, C., Feng, J., Lin, Z., and Yan, S. (2018). Exact low tubal rank tensor recovery from Gaussian measurements. In *Proceedings of the International Joint Conferences on Artificial Intelligence*, pages 2504–2510.
- [Lu et al., 2019b] Lu, C., Peng, X., and Wei, Y. (2019b). Low-rank tensor completion with a new tensor nuclear norm induced by invertible linear transforms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5996–6004.
- [Manning et al., 2010] Manning, C., Raghavan, P., and Schütze, H. (2010). *Introduction to information retrieval*. Cambridge University Press.
- [Mardani et al., 2015] Mardani, M., Mateos, G., and Giannakis, G. B. (2015). Subspace learning and imputation for streaming big data matrices and tensors. *IEEE Transactions on Signal Processing*, 63(10):2663–2677.
- [Qin et al., 2022] Qin, W., Wang, H., Zhang, F., Wang, J., Luo, X., and Huang, T. (2022). Low-rank high-order tensor completion with applications in visual data. *IEEE Transactions on Image Processing*, 31:2433–2448.
- [Shi and Malik, 2000] Shi, J. and Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905.
- [Song et al., 2020] Song, G., Ng, M. K., and Zhang, X. (2020). Robust tensor completion using transformed tensor singular value decomposition. *Numerical Linear Algebra with Applications*, 27(3):e2299.
- [Su et al., 2023] Su, Y., Hong, Z., Wu, X., and Lu, C. (2023). Invertible linear transforms based adaptive multi-view subspace clustering. *Signal Processing*, 209:109014.
- [Tang et al., 2014] Tang, K., Liu, R., Su, Z., and Zhang, J. (2014). Structure-constrained low-rank representation. *IEEE Transactions on Neural Networks and Learning Systems*, 25(12):2167–2179.
- [Tropp, 2012] Tropp, J. A. (2012). User-friendly tail bounds for sums of random matrices. *Foundations of Computational Mathematics*, 12:389–434.
- [Vinh et al., 2010] Vinh, N. X., Epps, J., and Bailey, J. (2010). Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *Journal of Machine Learning Research*, 11:2837–2854.
- [Wang et al., 2021] Wang, J.-L., Huang, T.-Z., Zhao, X.-L., Jiang, T.-X., and Ng, M. K. (2021). Multi-dimensional visual data completion via low-rank tensor representation under coupled transform. *IEEE Transactions on Image Processing*, 30:3581–3596.
- [Wu, 2020] Wu, T. (2020). Graph regularized low-rank representation for submodule clustering. *Pattern Recognition*, 100:107145.
- [Wu, 2023] Wu, T. (2023). Online tensor low-rank representation for streaming data clustering. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(2):602–617.
- [Wu and Bajwa, 2018] Wu, T. and Bajwa, W. U. (2018). A low tensor-rank representation approach for clustering of imaging data. *IEEE Signal Processing Letters*, 25(8):1196–1200.
- [Xia et al., 2021] Xia, G., Chen, B., Sun, H., and Liu, Q. (2021). Nonconvex low-rank kernel sparse subspace learning for keyframe extraction and motion segmentation. *IEEE Transactions on Neural Networks and Learning Systems*, 32(4):1612–1626.
- [Xie et al., 2016] Xie, J., Girshick, R., and Farhadi, A. (2016). Unsupervised deep embedding for clustering analysis. In *Proceedings of the International Conference on Machine Learning*, pages 478–487.

- [Xie et al., 2018] Xie, Y., Tao, D., Zhang, W., Liu, Y., Zhang, L., and Qu, Y. (2018). On unifying multi-view self-representations for clustering by tensor multi-rank minimization. *International Journal of Computer Vision*, 126(11):1157–1179.
- [Yang et al., 2015] Yang, C., Robinson, D., and Vidal, R. (2015). Sparse subspace clustering with missing entries. In *Proceedings of the International Conference on Machine Learning*, pages 2463–2472.
- [Yang et al., 2022] Yang, J.-H., Chen, C., Dai, H.-N., Ding, M., Wu, Z.-B., and Zheng, Z. (2022). Robust corrupted data recovery and clustering via generalized transformed tensor low-rank representation. *IEEE Transactions on Neural Networks and Learning Systems*.
- [You et al., 2016] You, C., Robinson, D. P., and Vidal, R. (2016). Scalable sparse subspace clustering by orthogonal matching pursuit. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3918–3927.
- [Zhang et al., 2015] Zhang, H., Lin, Z., Zhang, C., and Chang, E. (2015). Exact recoverability of robust PCA via outlier pursuit with tight recovery bounds. In *Proceedings of the AAAI Conference on Artificial Intelligence*.
- [Zhang et al., 2018] Zhang, J., Li, X., Jing, P., Liu, J., and Su, Y. (2018). Low-rank regularized heterogeneous tensor decomposition for subspace clustering. *IEEE Signal Processing Letters*, 25(3):333–337.
- [Zhou and Feng, 2017] Zhou, P. and Feng, J. (2017). Outlier-robust tensor PCA. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2263–2271.
- [Zhou et al., 2021] Zhou, P., Lu, C., Feng, J., Lin, Z., and Yan, S. (2021). Tensor low-rank representation for data recovery and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(5):1718–1732.

---

## Supplementary Material: Robust Data Clustering with Outliers via Transformed Tensor Low-Rank Representation

---

### A EXTENSION OF PRELIMINARIES

#### A.1 Notations

The operator norm of an operator on tensor is  $\|\mathcal{L}\| = \sup_{\|\mathcal{A}\|_F=1} \|\mathcal{L}(\mathcal{A})\|_F$ . The inner product between two matrices  $\mathbf{A}$  and  $\mathbf{B}$  in  $\mathbb{C}^{n_1 \times n_2}$  is defined as  $\langle \mathbf{A}, \mathbf{B} \rangle = \text{tr}(\mathbf{A}^H \mathbf{B})$ , whereas the inner product between two tensors  $\mathcal{A}$  and  $\mathcal{B}$  in  $\mathbb{C}^{n_1 \times n_2 \times n_3}$  is defined as  $\langle \mathcal{A}, \mathcal{B} \rangle = \sum_{i=1}^{n_3} \langle \mathbf{A}^{(i)}, \mathbf{B}^{(i)} \rangle$ . The dual norm of tensor  $\ell_{2,1}$  norm is the tensor  $\ell_{2,\infty}$  norm, which is defined as  $\|\mathcal{A}\|_{2,\infty} = \max_j \|\mathcal{A}(:, j, :)\|_F$ . The infinity norm of  $\mathcal{A}$  is  $\|\mathcal{A}\|_\infty = \max_{i,j,k} |\mathcal{A}_{i,j,k}|$ .

**Definition 7** (Tensor pseudo-inverse [Zhou et al., 2021]). *For an arbitrary tensor  $\mathcal{A} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$ , its pseudo-inverse under  $L$  in (3) is defined as a tensor  $\mathcal{A}^\dagger \in \mathbb{C}^{n_2 \times n_1 \times n_3}$  which satisfies (i)  $\mathcal{A} *_L \mathcal{A}^\dagger *_L \mathcal{A} = \mathcal{A}$ , (ii)  $\mathcal{A}^\dagger *_L \mathcal{A} *_L \mathcal{A}^\dagger = \mathcal{A}^\dagger$ , (iii)  $(\mathcal{A} *_L \mathcal{A}^\dagger)^H = \mathcal{A} *_L \mathcal{A}^\dagger$ , and (iv)  $(\mathcal{A}^\dagger *_L \mathcal{A})^H = \mathcal{A}^\dagger *_L \mathcal{A}$ .*

**Definition 8** (Standard tensor basis [Lu et al., 2019b]). *The tensor **column basis** with respect to the transform  $L$ , denoted as  $\hat{\mathbf{e}}_i$ , is a tensor of size  $n_1 \times 1 \times n_3$  with the entries of the  $(i, 1)$ -th tube of  $L(\hat{\mathbf{e}}_i)$  equaling 1 and the rest equaling 0. Similarly, the **row basis**  $\hat{\mathbf{e}}_j^H$  is of size  $1 \times n_2 \times n_3$  with the entries of the  $(1, j)$ -th tube of  $L(\hat{\mathbf{e}}_j^H)$  equaling to 1 and the rest equaling to 0. The **tube basis**  $\hat{\mathbf{e}}_k$  is a tensor of size  $1 \times 1 \times n_3$  with the  $(1, 1, k)$ -th entry of  $L(\hat{\mathbf{e}}_k)$  equaling 1 and the rest equaling 0.*

Denote  $\bar{\mathbf{e}}_{ijk}$  as a unit tensor with only the  $(i, j, k)$ -th entry equaling 1 and others equaling 0. Based on Definition 8,  $\bar{\mathbf{e}}_{ijk}$  can be expressed as  $\bar{\mathbf{e}}_{ijk} = L(\hat{\mathbf{e}}_i *_L \hat{\mathbf{e}}_k *_L \hat{\mathbf{e}}_j^H)$ . Then for any tensor  $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ , we have  $\mathcal{A}_{i,j,k} = \langle \mathcal{A}, \bar{\mathbf{e}}_{ijk} \rangle$  and  $\mathcal{A} = \sum_{i,j,k} \langle \mathcal{A}, \bar{\mathbf{e}}_{ijk} \rangle \bar{\mathbf{e}}_{ijk}$ .

Next, we define some commonly used operators in this document. Assume that  $\mathbf{U}_{\mathcal{X}} *_L \mathbf{S}_{\mathcal{X}} *_L \mathbf{V}_{\mathcal{X}}^H$ ,  $\mathbf{U}_0 *_L \mathbf{S}_0 *_L \mathbf{V}_0^H$ , and  $\mathbf{U} *_L \mathbf{S} *_L \mathbf{V}^H$  are the skinny t-SVDs of  $\mathcal{X}$ ,  $\mathcal{L}_0$  and  $\mathcal{Z}$ , respectively. The projection onto the column space  $\mathbf{U}$  is given by  $\mathcal{P}_{\mathbf{U}}(\mathcal{A}) = \mathbf{U} *_L \mathbf{U}^H *_L \mathcal{A}$ , and similarly for the row space  $\mathcal{P}_{\mathbf{V}}(\mathcal{A}) = \mathcal{A} *_L \mathbf{V} *_L \mathbf{V}^H$ . Sometimes, we need to apply  $\mathcal{P}_{\mathbf{V}}$  on the left side of a tensor. This special operator is denoted by  $\mathcal{P}_{\mathbf{V}}^L(\mathcal{A}) = \mathbf{V} *_L \mathbf{V}^H *_L \mathcal{A}$ . The tensor  $\mathcal{P}_{\Theta}(\mathcal{A})$  is obtained from  $\mathcal{A}$  by setting tensor column  $\mathcal{A}(:, j, :)$  to zero for all  $j \notin \Theta$ . The projection onto the space spanned by  $\mathbf{U}$  and  $\mathbf{V}$  is given by  $\mathcal{P}_{\mathcal{T}}(\mathcal{A}) = \mathcal{P}_{\mathbf{U}}(\mathcal{A}) + \mathcal{P}_{\mathbf{V}}(\mathcal{A}) - \mathcal{P}_{\mathbf{U}}\mathcal{P}_{\mathbf{V}}(\mathcal{A})$ , where  $\mathcal{P}_{\mathbf{U}}\mathcal{P}_{\mathbf{V}}(\mathcal{A}) = \mathbf{U} *_L \mathbf{U}^H *_L \mathcal{A} *_L \mathbf{V} *_L \mathbf{V}^H$ . The complementary operators,  $\mathcal{P}_{\mathbf{U}^\perp}$ ,  $\mathcal{P}_{\mathbf{V}^\perp}$ ,  $\mathcal{P}_{\mathbf{V}^\perp}^L$  and  $\mathcal{P}_{\mathcal{T}^\perp}$  are defined as usual. Finally, we use  $\Theta^c$  to denote the complement of  $\Theta$  and  $\mathcal{P}_{\Theta^\perp}$  is the projection onto  $\Theta^c$ .

#### A.2 Preliminaries

We consider the invertible linear transform  $L : \mathbb{R}^{n_1 \times n_2 \times n_3} \rightarrow \mathbb{C}^{n_1 \times n_2 \times n_3}$  defined as follows:

$$\bar{\mathcal{A}} = L(\mathcal{A}) = \mathcal{A} \times_3 \mathbf{L}, \quad (14)$$

where  $\times_3$  denotes the mode-3 tensor-matrix product, and  $\mathbf{L} \in \mathbb{C}^{n_3 \times n_3}$  is an arbitrary invertible matrix. In this document, we always use  $\mathbf{L}$  satisfying

$$\mathbf{L}\mathbf{L}^H = \mathbf{L}^H\mathbf{L} = \tau \mathbf{I}_{n_3}, \quad (15)$$

where  $\tau > 0$ . Using (15), we have the following properties:

$$\|\mathcal{A}\|_F = \frac{1}{\sqrt{\tau}} \|\bar{\mathcal{A}}\|_F = \frac{1}{\sqrt{\tau}} \|\bar{\bar{\mathcal{A}}}\|_F \quad \text{and} \quad \langle \mathcal{A}, \mathcal{B} \rangle = \frac{1}{\tau} \langle \bar{\mathcal{A}}, \bar{\mathcal{B}} \rangle = \frac{1}{\tau} \langle \bar{\bar{\mathcal{A}}}, \bar{\bar{\mathcal{B}}} \rangle.$$

## B OPTIMIZATION DETAILS OF OR-TLRR

In this section, we elaborate the details of our optimization strategy to solve problem (7) in the manuscript. To solve (7) in an efficient manner, we resort to “variable splitting” of  $\mathcal{Z}'$ , which transforms (7) into the following:

$$\min_{\mathcal{Z}', \mathcal{J}, \mathcal{E}} \|\mathcal{Z}'\|_* + \lambda \|\mathcal{E}\|_{2,1} \quad \text{s.t.} \quad \mathcal{Z}' = \mathcal{J}, \mathcal{X} = \mathcal{D} *_L \mathcal{J} + \mathcal{E}. \quad (16)$$

The augmented Lagrangian function of (16) is

$$\begin{aligned} \mathcal{L}_1(\mathcal{Z}', \mathcal{J}, \mathcal{E}, \mathcal{Y}_1, \mathcal{Y}_2, \beta) &= \|\mathcal{Z}'\|_* + \lambda \|\mathcal{E}\|_{2,1} + \langle \mathcal{Y}_1, \mathcal{Z}' - \mathcal{J} \rangle + \langle \mathcal{Y}_2, \mathcal{X} - \mathcal{D} *_L \mathcal{J} - \mathcal{E} \rangle \\ &+ \frac{\beta}{2} (\|\mathcal{Z}' - \mathcal{J}\|_F^2 + \|\mathcal{X} - \mathcal{D} *_L \mathcal{J} - \mathcal{E}\|_F^2), \end{aligned} \quad (17)$$

where the tensors  $\mathcal{Y}_1$  and  $\mathcal{Y}_2$  comprise Lagrange multipliers and  $\beta > 0$  is a penalty parameter. The optimization of (17) can be done iteratively by minimizing  $\mathcal{L}_1$  with respect to  $\mathcal{Z}'$ ,  $\mathcal{J}$  and  $\mathcal{E}$  over one tensor at a time while keeping the others fixed.

**Updating  $\mathcal{Z}'$ :** When other variables are fixed, the problem of updating  $\mathcal{Z}'$  can be written as

$$\mathcal{Z}'^{(t+1)} = \arg \min_{\mathcal{Z}'} \|\mathcal{Z}'\|_* + \frac{\beta^{(t)}}{2} \|\mathcal{B}_1^{(t+1)} - \mathcal{Z}'\|_F^2,$$

where  $\mathcal{B}_1^{(t+1)} = \mathcal{J}^{(t)} - \frac{\mathcal{Y}_1^{(t)}}{\beta^{(t)}}$ . We can optimize its equivalent problem:

$$\bar{\mathcal{Z}}'^{(t+1)} = \arg \min_{\bar{\mathcal{Z}}'} \frac{1}{n_3} \left( \|\bar{\mathcal{Z}}'\|_* + \frac{\beta^{(t)}}{2} \|\bar{\mathcal{B}}_1^{(t+1)} - \bar{\mathcal{Z}}'\|_F^2 \right),$$

where  $\bar{\mathcal{B}}_1^{(t+1)} = \text{bdiag}(\bar{\mathcal{B}}_1^{(t+1)})$  and  $\bar{\mathcal{B}}_1^{(t+1)} = L(\mathcal{B}_1^{(t+1)})$ . Since  $\bar{\mathcal{Z}}'$  is a block diagonal matrix, we only need to update all the diagonal block matrices  $\bar{\mathcal{Z}}'^{(k)}$  by

$$\bar{\mathcal{Z}}'^{(k)(t+1)} = \Upsilon_{\frac{1}{\beta^{(t)}}}(\bar{\mathcal{B}}_1^{(k)(t+1)}), \quad (18)$$

where  $\Upsilon_{\zeta}(\cdot)$  denotes singular value thresholding (SVT) operator [Cai et al., 2010]. Finally, we can compute  $\mathcal{Z}'^{(t+1)} = L^{-1}(\bar{\mathcal{Z}}'^{(t+1)})$ .

**Updating  $\mathcal{E}$ :** We can update  $\mathcal{E}$  by solving

$$\mathcal{E}^{(t+1)} = \arg \min_{\mathcal{E}} \lambda \|\mathcal{E}\|_{2,1} + \frac{\beta^{(t)}}{2} \|\mathcal{B}_2^{(t+1)} - \mathcal{E}\|_F^2,$$

where  $\mathcal{B}_2^{(t+1)} = \mathcal{X} - \mathcal{D} *_L \mathcal{J}^{(t)} + \frac{\mathcal{Y}_2^{(t)}}{\beta^{(t)}}$ . Then the closed-form solution for  $\mathcal{E}$  is given by

$$\mathcal{E}_{(j)}^{(t+1)} = \begin{cases} \frac{\|\mathcal{B}_{2(j)}^{(t+1)}\|_F - \frac{\lambda}{\beta^{(t)}}}{\|\mathcal{B}_{2(j)}^{(t+1)}\|_F} \mathcal{B}_{2(j)}^{(t+1)}, & \text{if } \|\mathcal{B}_{2(j)}^{(t+1)}\|_F > \frac{\lambda}{\beta^{(t)}}, \\ \mathbf{0}, & \text{otherwise,} \end{cases} \quad (19)$$

where  $\mathcal{B}_{2(j)}^{(t+1)} = \mathcal{B}_2^{(t+1)}(:, j, :)$ .

**Updating  $\mathcal{J}$ :** Keeping other tensors in (17) fixed, the subproblem of updating  $\mathcal{J}$  at the  $(t+1)$ -th iteration has the form

$$\begin{aligned} \mathcal{J}^{(t+1)} &= \arg \min_{\mathcal{J}} \|\mathcal{P}_1^{(t+1)} - \mathcal{J}\|_F^2 + \|\mathcal{P}_2^{(t+1)} - \mathcal{D} *_L \mathcal{J}\|_F^2 \\ &= (\mathcal{D}^H *_L \mathcal{D} + \mathcal{I}_{r_{\mathcal{X}}})^{-1} *_L \left( \mathcal{P}_1^{(t+1)} + \mathcal{D}^H *_L \mathcal{P}_2^{(t+1)} \right), \end{aligned} \quad (20)$$

where  $\mathcal{P}_1^{(t+1)} = \mathcal{Z}'^{(t+1)} + \frac{\mathbf{y}_1^{(t)}}{\beta^{(t)}}$  and  $\mathcal{P}_2^{(t+1)} = \mathcal{X} - \mathcal{E}^{(t+1)} + \frac{\mathbf{y}_2^{(t)}}{\beta^{(t)}}$ . We can transform (20) into the transform domain and it can be shown that the  $k$ -th frontal slice of  $\bar{\mathcal{J}}^{(t+1)}$ , denoted by  $\bar{\mathbf{J}}^{(k)(t+1)}$ , has the following closed-form solution:

$$\bar{\mathbf{J}}^{(k)(t+1)} = \bar{\mathbf{Q}}^{(k)} \left( \bar{\mathbf{P}}_1^{(k)(t+1)} + (\bar{\mathbf{D}}^{(k)})^H \bar{\mathbf{P}}_2^{(k)(t+1)} \right), \quad (21)$$

where  $\bar{\mathbf{Q}} = (\mathcal{D}^H *_L \mathcal{D} + \mathcal{I}_{r_{\mathcal{X}}})^{-1}$ . The update for  $\mathcal{J}^{(t+1)}$  can be expressed as  $\mathcal{J}^{(t+1)} = L^{-1}(\bar{\mathcal{J}}^{(t+1)})$ .

After obtaining a solution  $(\mathcal{Z}'_*, \mathcal{E}_*)$  to the problem (7), the optimal solution to (4) in the manuscript can be recovered by  $(\mathcal{V}_{\mathcal{X}} *_L \mathcal{Z}'_*, \mathcal{E}_*)$ . The ADMM framework for solving (4) in the manuscript can be summarized in Algorithm 1. Since problem (16) is a convex problem, whose convergence behavior can be guaranteed by leveraging the results in [Lin et al., 2011].

**Complexity Analysis.** At each iteration, when updating  $\mathcal{J}^{(t+1)}$ , the computational cost for the matrix product and inverse transform is  $\mathcal{O}(r_{\mathcal{X}} n_1 n_2 n_3 + r_{\mathcal{X}} (n_1 + n_2) n_3^2)$ . The major cost of updating  $\mathcal{Z}'^{(t+1)}$  includes  $n_3$  SVDs on  $r_{\mathcal{X}} \times n_2$  matrices at the cost of  $r_{\mathcal{X}}^2 n_2 n_3$  and computing the inverse transform at the cost of  $\mathcal{O}(r_{\mathcal{X}} n_2 n_3^2)$ . As for updating  $\mathcal{E}^{(t+1)}$ , the step of computing tensor product  $\mathcal{D} *_L \mathcal{J}^{(t+1)}$  takes  $\mathcal{O}(r_{\mathcal{X}} n_1 n_2 n_3 + r_{\mathcal{X}} (n_1 + n_2) n_3^2)$ . Thus the total cost of Algorithm 1 is  $\mathcal{O}(r_{\mathcal{X}} n_1 n_2 n_3 + r_{\mathcal{X}} (n_1 + n_2) n_3^2)$  per iteration. We conclude this section by noting that for some special invertible linear transforms  $L$ , e.g., DFT, since the application of DFT on an  $n_3$ -dimensional vector requires  $\mathcal{O}(n_3 \log(n_3))$  operations, the per-iteration complexity is  $\mathcal{O}(r_{\mathcal{X}} n_1 n_2 n_3 + r_{\mathcal{X}} (n_1 + n_2) n_3 \log(n_3))$ .

---

**Algorithm 1** Outlier-Robust Tensor LRR (OR-TLRR)

---

**Input:** Data tensor  $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ , linear transform  $L$ , and parameter  $\lambda$ .

**Initialize:**  $\mathcal{D} = \mathcal{U}_{\mathcal{X}} *_L \mathcal{S}_{\mathcal{X}}$  with the skinny t-SVD  $\mathcal{U}_{\mathcal{X}} *_L \mathcal{S}_{\mathcal{X}} *_L \mathcal{V}_{\mathcal{X}}^H$  of  $\mathcal{X}$ ,  $\mathcal{Z}^{(0)} = \mathcal{J}^{(0)} = \mathbf{y}_1^{(0)} = \mathbf{0} \in \mathbb{R}^{r_{\mathcal{X}} \times n_2 \times n_3}$ ,  $\mathcal{E}^{(0)} = \mathbf{y}_2^{(0)} = \mathbf{0} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ ,  $\gamma = 1.1$ ,  $\beta^{(0)} = 10^{-5}$ ,  $\beta_{\max} = 10^8$ ,  $\epsilon = 10^{-5}$ , and  $t = 0$ .

- 1: **while** not converged **do**
- 2: Fix other variables and update  $\mathcal{Z}'$  via (18).
- 3: Fix other variables and update  $\mathcal{E}$  via (19).
- 4: Fix other variables and update  $\mathcal{J}$  via (21).
- 5: Update Lagrange multipliers:
 
$$\mathbf{y}_1^{(t+1)} = \mathbf{y}_1^{(t)} + \beta^{(t)} (\mathcal{Z}'^{(t+1)} - \mathcal{J}^{(t+1)}),$$

$$\mathbf{y}_2^{(t+1)} = \mathbf{y}_2^{(t)} + \beta^{(t)} (\mathcal{X} - \mathcal{D} *_L \mathcal{J}^{(t+1)} - \mathcal{E}^{(t+1)}).$$
- 6:  $\beta^{(t+1)} = \min(\beta_{\max}, \gamma \beta^{(t)})$ .
- 7: Check the convergence conditions:

$$\max \left\{ \begin{array}{l} \|\mathcal{Z}'^{(t+1)} - \mathcal{Z}'^{(t)}\|_{\infty}, \|\mathcal{J}^{(t+1)} - \mathcal{J}^{(t)}\|_{\infty} \\ \|\mathcal{E}^{(t+1)} - \mathcal{E}^{(t)}\|_{\infty}, \|\mathcal{Z}'^{(t+1)} - \mathcal{J}^{(t+1)}\|_{\infty} \\ \|\mathcal{X} - \mathcal{D} *_L \mathcal{J}^{(t+1)} - \mathcal{E}^{(t+1)}\|_{\infty} \end{array} \right\} \leq \epsilon.$$

- 8:  $t = t + 1$ .
- 9: **end while**

**Output:**  $\mathcal{Z}_* = \mathcal{V}_{\mathcal{X}} *_L \mathcal{Z}'^{(t+1)}$  and  $\mathcal{E}_* = \mathcal{E}^{(t+1)}$ .

---

## C OPTIMIZATION DETAILS OF OR-TLRR WITH MISSING ENTRIES

In this section, we present our algorithm to solve problem (13) in the manuscript. By introducing two auxiliary variables  $\mathcal{J}$  and  $\mathcal{H}$ , we make the objective function of (13) separable and reformulate (13) as

$$\begin{aligned} \min_{\mathcal{Z}', \mathcal{J}, \mathcal{H}, \mathcal{E}} \quad & \|\mathcal{Z}'\|_* + \lambda \|\mathcal{P}_{\Omega}(\mathcal{E})\|_{2,1} \\ \text{s.t.} \quad & \mathcal{Z}' = \mathcal{J}, \mathcal{H} = \mathcal{D} *_L \mathcal{J} + \mathcal{E}, \mathcal{P}_{\Omega}(\mathcal{X}_{\text{miss}}) = \mathcal{P}_{\Omega}(\mathcal{H}). \end{aligned} \quad (22)$$

The corresponding augmented Lagrangian function of (22) is

$$\begin{aligned} \mathcal{L}_2(\mathcal{Z}', \mathcal{J}, \mathcal{H}, \mathcal{E}, \mathcal{Y}_1, \mathcal{Y}_2, \beta) &= \|\mathcal{Z}'\|_* + \lambda \|\mathcal{P}_\Omega(\mathcal{E})\|_{2,1} + \Psi_\Omega(\mathcal{X}_{\text{miss}} - \mathcal{H}) + \langle \mathcal{Y}_1, \mathcal{Z}' - \mathcal{J} \rangle + \langle \mathcal{Y}_2, \mathcal{H} - \mathcal{D} *_{\mathcal{L}} \mathcal{J} - \mathcal{E} \rangle \\ &\quad + \frac{\beta}{2} (\|\mathcal{Z}' - \mathcal{J}\|_F^2 + \|\mathcal{H} - \mathcal{D} *_{\mathcal{L}} \mathcal{J} - \mathcal{E}\|_F^2), \end{aligned} \quad (23)$$

where  $\Psi_\Omega(\cdot)$  denotes the indicator function

$$\Psi_\Omega(\mathcal{A} - \mathcal{B}) = \begin{cases} 0, & \text{if } \mathcal{A}_\Omega = \mathcal{B}_\Omega, \\ \infty, & \text{otherwise.} \end{cases}$$

The difference between the optimization of (23) and that of (17) mainly lies in the step of updating  $\mathcal{E}$  and one additional step for updating  $\mathcal{H}$ . Concretely, the  $\mathcal{H}$ -subproblem is

$$\mathcal{H}^{(t+1)} = \arg \min_{\mathcal{H}} \|\mathcal{H} - \mathcal{D} *_{\mathcal{L}} \mathcal{J}^{(t)} - \mathcal{E}^{(t)} + \frac{\mathcal{Y}_2^{(t)}}{\beta^{(t)}}\|_F^2 + \Psi_\Omega(\mathcal{X}_{\text{miss}} - \mathcal{H}). \quad (24)$$

Thus, we have  $\mathcal{H}^{(t+1)} = \mathcal{P}_{\Omega^c}(\mathcal{D} *_{\mathcal{L}} \mathcal{J}^{(t)} + \mathcal{E}^{(t)} - \frac{\mathcal{Y}_2^{(t)}}{\beta^{(t)}}) + \mathcal{P}_\Omega(\mathcal{X}_{\text{miss}})$ , where  $\Omega^c$  is the complement of  $\Omega$ . On the other hand, the variable  $\mathcal{E}$  can be updated by solving

$$\mathcal{E}^{(t+1)} = \arg \min_{\mathcal{E}} \lambda \|\mathcal{P}_\Omega(\mathcal{E})\|_{2,1} + \frac{\beta^{(t)}}{2} \|\mathcal{B}^{(t+1)} - \mathcal{E}\|_F^2, \quad (25)$$

where  $\mathcal{B}^{(t+1)} = \mathcal{H}^{(t+1)} - \mathcal{D} *_{\mathcal{L}} \mathcal{J}^{(t)} + \frac{\mathcal{Y}_2^{(t)}}{\beta^{(t)}}$ . Then the closed-form solution of  $\mathcal{E}$  is given by

$$\mathcal{E}_{(j)}^{(t+1)} = \begin{cases} \frac{\|\mathcal{W}_{(j)} \odot \mathcal{B}_{(j)}^{(t+1)}\|_F - \frac{\lambda}{\beta^{(t)}}}{\|\mathcal{W}_{(j)} \odot \mathcal{B}_{(j)}^{(t+1)}\|_F} \mathcal{W}_{(j)} \odot \mathcal{B}_{(j)}^{(t+1)} + (1 - \mathcal{W}_{(j)}) \odot \mathcal{B}_{(j)}^{(t+1)}, & \text{if } \|\mathcal{W}_{(j)} \odot \mathcal{B}_{(j)}^{(t+1)}\|_F > \frac{\lambda}{\beta^{(t)}}, \\ (1 - \mathcal{W}_{(j)}) \odot \mathcal{B}_{(j)}^{(t+1)}, & \text{otherwise,} \end{cases}$$

where  $\mathcal{B}_{(j)}^{(t+1)} = \mathcal{B}^{(t+1)}(:, j, :)$ . The implementation of the ADMM algorithm is outlined in Algorithm 2. We dub this approach *Outlier-Robust Tensor LRR by Entry-Wise Zero-Fill* (OR-TLRR-EWZF). We conclude this section by noting that we can also robustify TLRR [Zhou et al., 2021] by replacing  $\lambda \|\mathcal{P}_\Omega(\mathcal{E})\|_{2,1}$  in (12) with  $\lambda \|\mathcal{P}_\Omega(\mathcal{E})\|_1$  as a means of performing TLRR with missing data. Effectively, the overall optimization process also relies on Algorithm 2, with the difference being that the closed-form solution of  $\mathcal{E}$  can now be computed by

$$\mathcal{E}_{(j)}^{(t+1)} = \Pi_{\frac{\lambda}{\beta^{(t)}}} \left( \mathcal{W}_{(j)} \odot \mathcal{B}_{(j)}^{(t+1)} \right) + (1 - \mathcal{W}_{(j)}) \odot \mathcal{B}_{(j)}^{(t+1)},$$

where  $\Pi_{\frac{\lambda}{\beta^{(t)}}}(\cdot)$  is the soft-thresholding operator [Donoho, 1995]. We call the resulting algorithm TLRR-EWZF in our experiments.

## D PROOF OF THEOREM 1

*Proof.* Let  $\mathbf{x}_j = \text{vec}(\mathcal{X}_{(j)})$  and  $\mathbf{X}_j = \text{squeeze}(\mathcal{X}_{(j)}) \in \mathbb{R}^{n_1 \times n_3}$ . Then, we can obtain

$$\mathbf{x}_j = \begin{bmatrix} \mathbf{X}_j \mathbf{e}_1 \\ \vdots \\ \mathbf{X}_j \mathbf{e}_{n_3} \end{bmatrix},$$

where  $\mathbf{e}_q \in \mathbb{R}^{n_3}$  is the  $q$ -th standard basis of  $\mathbb{R}^{n_3}$  whose  $q$ -th entry is 1 and the rest is 0. Next, we define a tensor  $\mathcal{A} \in \mathbb{R}^{n_1 \times p \times n_3}$  with  $\mathcal{A}_{(j)} = \text{ivector}(\mathbf{a}_j)$ , and let  $\mathbf{A}_j = \text{squeeze}(\mathcal{A}_{(j)}) \in \mathbb{R}^{n_1 \times n_3}$ ,  $j = 1, \dots, p$ . Then we can write  $\mathbf{x}_j = \mathbf{A}_j \mathbf{z}_j$  as its equivalent form:

$$\begin{bmatrix} \mathbf{X}_j \mathbf{e}_1 \\ \vdots \\ \mathbf{X}_j \mathbf{e}_{n_3} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 \mathbf{e}_1 & \dots & \mathbf{A}_p \mathbf{e}_1 \\ \vdots & \ddots & \vdots \\ \mathbf{A}_1 \mathbf{e}_{n_3} & \dots & \mathbf{A}_p \mathbf{e}_{n_3} \end{bmatrix} \mathbf{z}_j.$$

**Algorithm 2** Outlier-Robust Tensor LRR by Entry-Wise Zero-Fill (OR-TLRR-EWZF)

---

**Input:** Partially observed data tensor  $\mathcal{P}_\Omega(\mathcal{X}) \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ , linear transform  $L$ , and parameter  $\lambda$ .

**Initialize:**  $\mathcal{D} = \mathcal{U}_\mathcal{X} *_{L} \mathcal{S}_\mathcal{X}$  with the skinny t-SVD  $\mathcal{U}_\mathcal{X} *_{L} \mathcal{S}_\mathcal{X} *_{L} \mathcal{V}_\mathcal{X}^H$  of  $\mathcal{X}_{\text{miss}}$ ,  $\mathcal{Z}^{(0)} = \mathcal{J}^{(0)} = \mathcal{Y}_1^{(0)} = \mathbf{0} \in \mathbb{R}^{r \times n_2 \times n_3}$ ,  $\mathcal{E}^{(0)} = \mathcal{H}^{(0)} = \mathcal{Y}_2^{(0)} = \mathbf{0} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ ,  $\gamma = 1.1$ ,  $\beta^{(0)} = 10^{-5}$ ,  $\beta_{\max} = 10^8$ ,  $\epsilon = 10^{-5}$ , and  $t = 0$ .

- 1: **while** not converged **do**
- 2: Fix other variables and update  $\mathcal{H}$  by solving (24).
- 3: Fix other variables and update  $\mathcal{Z}'$  by solving:

$$\mathcal{Z}'^{(t+1)} = \arg \min_{\mathcal{Z}'} \|\mathcal{Z}'\|_* + \frac{\mu^{(t)}}{2} \|\mathcal{Z}' - \mathcal{J}^{(t)} + \frac{\mathcal{Y}_1^{(t)}}{\beta^{(t)}}\|_F^2.$$

- 4: Fix other variables and update  $\mathcal{E}$  by solving (25).
- 5: Fix other variables and update  $\mathcal{J}$  by solving:

$$\mathcal{J}^{(t+1)} = \arg \min_{\mathcal{J}} \|\mathcal{Z}'^{(t+1)} - \mathcal{J} + \frac{\mathcal{Y}_1^{(t)}}{\beta^{(t)}}\|_F^2 + \|\mathcal{H}^{(t+1)} - \mathcal{D} *_{L} \mathcal{J} - \mathcal{E}^{(t+1)} + \frac{\mathcal{Y}_2^{(t)}}{\beta^{(t)}}\|_F^2.$$

- 6: Update Lagrange multipliers:
  - $\mathcal{Y}_1^{(t+1)} = \mathcal{Y}_1^{(t)} + \beta^{(t)}(\mathcal{Z}'^{(t+1)} - \mathcal{J}^{(t+1)})$ ,
  - $\mathcal{Y}_2^{(t+1)} = \mathcal{Y}_2^{(t)} + \beta^{(t)}(\mathcal{H}^{(t+1)} - \mathcal{D} *_{L} \mathcal{J}^{(t+1)} - \mathcal{E}^{(t+1)})$ .
- 7:  $\beta^{(t+1)} = \min(\beta_{\max}, \gamma\beta^{(t)})$ .
- 8: Check the convergence conditions:

$$\max \left\{ \begin{array}{l} \|\mathcal{Z}'^{(t+1)} - \mathcal{Z}'^{(t)}\|_\infty, \|\mathcal{J}^{(t+1)} - \mathcal{J}^{(t)}\|_\infty \\ \|\mathcal{E}^{(t+1)} - \mathcal{E}^{(t)}\|_\infty, \|\mathcal{H}^{(t+1)} - \mathcal{H}^{(t)}\|_\infty \\ \|\mathcal{Z}'^{(t+1)} - \mathcal{J}^{(t+1)}\|_\infty \\ \|\mathcal{H}^{(t+1)} - \mathcal{D} *_{L} \mathcal{J}^{(t+1)} - \mathcal{E}^{(t+1)}\|_\infty \end{array} \right\} \leq \epsilon.$$

- 9:  $t = t + 1$ .
- 10: **end while**

**Output:**  $\mathcal{Z}_* = \mathcal{V}_\mathcal{X} *_{L} \mathcal{Z}'^{(t+1)}$  and  $\mathcal{E}_* = \mathcal{E}^{(t+1)}$ .

---

For the  $s$ -th row of  $L$  ( $s = 1, \dots, n_3$ ), we have

$$\begin{aligned} \mathbf{X}_j \mathbf{l}_{s,T}^T &= \sum_{q=1}^{n_3} \mathbf{L}_{s,q} \mathbf{X}_j \mathbf{e}_q \\ &= \sum_{q=1}^{n_3} \mathbf{L}_{s,q} [\mathbf{A}_1 \mathbf{e}_q, \dots, \mathbf{A}_p \mathbf{e}_q] \mathbf{z}_j \\ &= [\mathbf{A}_1, \dots, \mathbf{A}_p] \sum_{q=1}^{n_3} \mathbf{L}_{s,q} \begin{bmatrix} \mathbf{e}_q & & \\ & \ddots & \\ & & \mathbf{e}_q \end{bmatrix} \mathbf{z}_j \\ &= [\mathbf{A}_1, \dots, \mathbf{A}_p] \begin{bmatrix} \mathbf{l}_{s,T}^T & & \\ & \ddots & \\ & & \mathbf{l}_{s,T}^T \end{bmatrix} \mathbf{z}_j \\ &= [\mathbf{A}_1 \mathbf{l}_{s,T}^T, \dots, \mathbf{A}_p \mathbf{l}_{s,T}^T] \mathbf{z}_j. \end{aligned} \tag{26}$$

We further define

$$\mathbf{A}_{l_{s,T}} = [\mathbf{A}_1 \mathbf{l}_{s,T}^T, \dots, \mathbf{A}_p \mathbf{l}_{s,T}^T] \in \mathbb{C}^{n_1 \times n_3}$$

and a tensor  $\bar{\mathcal{Z}} \in \mathbb{R}^{p \times n_2 \times n_3}$  with  $\bar{\mathcal{Z}}(:, j, k) = \mathbf{z}_j \in \mathbb{R}^p$ ,  $k = 1, \dots, n_3$ . The tensor  $\mathcal{Z}$  can be computed by setting  $\mathcal{Z} = L^{-1}(\bar{\mathcal{Z}})$ . Note that the block diagonal matrix  $\bar{\mathbf{A}}$  can now be written as

$$\bar{\mathbf{A}} = \begin{bmatrix} \mathbf{A}_{l_{1,T}} & & \\ & \ddots & \\ & & \mathbf{A}_{l_{n_3,T}} \end{bmatrix}.$$

Let  $\bar{\mathbf{X}}_j = \text{bdiag}(L(\mathcal{X}_{(j)}))$  and  $\bar{\mathbf{Z}}_j = \text{bdiag}(L(\mathcal{Z}_{(j)}))$ , we can establish that

$$\begin{aligned} \bar{\mathbf{X}}_j &= \begin{bmatrix} \mathbf{X}_j l_{1,T}^T & & \\ & \ddots & \\ & & \mathbf{X}_j l_{n_3,T}^T \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{A}_{l_{1,T}} & & \\ & \ddots & \\ & & \mathbf{A}_{l_{n_3,T}} \end{bmatrix} \begin{bmatrix} \mathbf{z}_j & & \\ & \ddots & \\ & & \mathbf{z}_j \end{bmatrix} \\ &= \bar{\mathbf{A}} \bar{\mathbf{Z}}_j. \end{aligned}$$

So for any  $\mathbf{z}_j$ , we have

$$\mathcal{X}_{(j)} = \mathcal{A} *_L \mathcal{Z}_{(j)}, \quad \forall j = 1, \dots, n_2. \quad (27)$$

Conversely, for any tensor linear representation (27),  $\mathcal{Z}_{(j)}$  may not satisfy  $\bar{\mathcal{Z}}(:, j, k) = \mathbf{z}_j$ , i.e., there may not exist  $\mathbf{z}_j$  such that  $\mathbf{x}_j = \mathbf{A}\mathbf{z}_j$ . Thus, if the linear representation relationship in vector space holds, then there exists a feasible solution such that the tensor linear representation also holds. However, there is no guarantee that the reverse is also true.  $\square$

## E PROOFS OF LEMMA 1 AND LEMMA 2

Before proving Lemma 1, we first give the following lemma which is used in the proof.

**Lemma 3.** *Let  $\mathcal{A} \in \mathbb{R}^{n_1 \times n_4 \times n_3}$  be a nonzero dictionary and  $\mathcal{X} = \mathcal{A} *_L \mathcal{Z}$  has feasible solutions, i.e.,  $\mathcal{X} \in \text{Range}(\mathcal{A})$ . Then the following problem*

$$\min_{\mathcal{Z}} \|\mathcal{Z}\|_* \quad \text{s.t.} \quad \mathcal{X} = \mathcal{A} *_L \mathcal{Z}$$

has a unique minimizer defined by

$$\mathcal{Z}_* = \mathcal{A}^\dagger *_L \mathcal{X}.$$

*Proof.* For the problem

$$\min_{\mathcal{Z}} \|\mathcal{Z}\|_* \quad \text{s.t.} \quad \mathcal{X} = \mathcal{A} *_L \mathcal{Z} \quad (28)$$

is equivalent to

$$\min_{\bar{\mathcal{Z}}} \|\bar{\mathcal{Z}}\|_* \quad \text{s.t.} \quad \bar{\mathbf{X}} = \bar{\mathbf{A}} \bar{\mathcal{Z}}. \quad (29)$$

Since  $\bar{\mathbf{X}}$ ,  $\bar{\mathbf{A}}$  and  $\bar{\mathcal{Z}}$  are three block-diagonal matrices, (29) can be divided into  $n_3$  simple problems.

$$\min_{\bar{\mathcal{Z}}^{(k)}} \|\bar{\mathcal{Z}}^{(k)}\|_* \quad \text{s.t.} \quad \bar{\mathbf{X}}^{(k)} = \bar{\mathbf{A}}^{(k)} \bar{\mathcal{Z}}^{(k)}, \quad k = 1, \dots, n_3. \quad (30)$$

Since  $\mathcal{X} \in \text{Range}(\mathcal{A})$ , we have  $\bar{\mathbf{X}}^{(k)} \in \text{Span}(\bar{\mathbf{A}}^{(k)})$ , where  $\text{Span}(\mathbf{B})$  denotes the linear space spanned by the columns of a matrix  $\mathbf{B}$  [Liu et al., 2013a]. Then, by [Liu et al., 2013a, Theorem 4.1], we have that  $\bar{\mathcal{Z}}^{(k)} = (\bar{\mathbf{A}}^{(k)})^\dagger \bar{\mathbf{X}}^{(k)}$  is the unique optimal solution to (30). Hence, we can obtain the unique solution  $\bar{\mathcal{Z}} = (\bar{\mathbf{A}})^\dagger \bar{\mathbf{X}}$  to problem (29). Furthermore, the unique solution to (28) is  $\mathcal{Z} = \mathcal{A}^\dagger *_L \mathcal{X}$ . The proof is completed.  $\square$

### E.1 Proof of Lemma 1

*Proof.* Note that OR-TLRR problem (4) always has feasible solutions, e.g.,  $(\mathcal{Z} = \mathbf{0}, \mathcal{E} = \mathcal{X})$ . Thus, an optimal solution, denoted as  $(\mathcal{Z}_*, \mathcal{E}_*)$ , exists. By Lemma 3, we have

$$\begin{aligned} \mathcal{Z}_* &= \arg \min_{\mathcal{Z}} \|\mathcal{Z}\|_* \quad \text{s.t.} \quad \mathcal{X} - \mathcal{E}_* = \mathcal{X} *_L \mathcal{Z} \\ &= \mathcal{X}^\dagger *_L (\mathcal{X} - \mathcal{E}_*), \end{aligned}$$

which simply leads to  $\mathcal{Z}_* \in \mathcal{P}_{\mathcal{V}_\mathcal{X}}^L$ .  $\square$

### E.2 Proof of Lemma 2

*Proof.* Suppose the skinny t-SVDs of  $\mathcal{X}$ ,  $\mathcal{L}_0$  and  $\mathcal{E}_0$  are  $\mathcal{U}_\mathcal{X} *_L \mathcal{S}_\mathcal{X} *_L \mathcal{V}_\mathcal{X}^H$ ,  $\mathcal{U}_0 *_L \mathcal{S}_0 *_L \mathcal{V}_0^H$  and  $\mathcal{U}_\mathcal{E} *_L \mathcal{S}_\mathcal{E} *_L \mathcal{V}_\mathcal{E}^H$ , respectively. Let  $\mathcal{U}_0^\perp$  and  $\mathcal{U}_\mathcal{E}^\perp$  be the orthogonal complements of  $\mathcal{U}_0$  and  $\mathcal{U}_\mathcal{E}$ , respectively. Since  $\text{Range}(\mathcal{L}_0)$  and  $\text{Range}(\mathcal{E}_0)$  are independent,  $\text{Span}(\bar{\mathbf{U}}_0^{(k)})$  and  $\text{Span}(\bar{\mathbf{U}}_\mathcal{E}^{(k)})$  are independent,  $\forall k = 1, \dots, n_3$ , which further suggests that  $[\bar{\mathbf{U}}_0^{(k)\perp}, \bar{\mathbf{U}}_\mathcal{E}^{(k)\perp}]$  spans the whole space. Thus the following system has feasible solutions  $\bar{\mathbf{Y}}_0^{(k)}$  and  $\bar{\mathbf{Y}}_\mathcal{E}^{(k)}$ :

$$\bar{\mathbf{U}}_0^{(k)\perp} (\bar{\mathbf{U}}_0^{(k)})^H \bar{\mathbf{Y}}_0^{(k)} + \bar{\mathbf{U}}_\mathcal{E}^{(k)\perp} (\bar{\mathbf{U}}_\mathcal{E}^{(k)})^H \bar{\mathbf{Y}}_\mathcal{E}^{(k)} = \mathbf{I}_{n_1}.$$

Denote  $\bar{\mathbf{Y}}^{(k)} = \mathbf{I}_{n_1} - \bar{\mathbf{U}}_0^{(k)\perp} (\bar{\mathbf{U}}_0^{(k)})^H \bar{\mathbf{Y}}_0^{(k)}$ , then it can be verified that

$$(\bar{\mathbf{L}}_0^{(k)})^H \bar{\mathbf{Y}}^{(k)} = (\bar{\mathbf{L}}_0^{(k)})^H \quad \text{and} \quad (\bar{\mathbf{E}}_0^{(k)})^H \bar{\mathbf{Y}}^{(k)} = \mathbf{0}.$$

We have that  $(\bar{\mathbf{X}}^{(k)})^H \bar{\mathbf{Y}}^{(k)} = (\bar{\mathbf{L}}_0^{(k)})^H$ ,  $\forall k = 1, \dots, n_3$ . Hence the system  $\mathcal{X}^H *_L \mathcal{Y} = \mathcal{L}_0^H$  has a feasible solution  $\mathcal{Y}$ , which simply leads to  $\mathcal{V}_\mathcal{X} *_L \mathcal{S}_\mathcal{X}^\dagger *_L \mathcal{U}_\mathcal{X}^H *_L \mathcal{Y} *_L \mathcal{U}_0 *_L \mathcal{S}_0 = \mathcal{V}_0$  and thus  $\mathcal{V}_0 \in \mathcal{P}_{\mathcal{V}_\mathcal{X}}^L$ .  $\square$

## F PROOFS OF THEOREM 2

Now we prove Theorem 2 in manuscript. Section F.1 identifies the necessary and sufficient conditions (called equivalent conditions) for any pair  $(\mathcal{Z}, \mathcal{E})$  to produce the exact recovery results. Section F.2 gives the dual conditions under which OR-TLRR succeeds. Then we construct a dual certificate in Section F.3 such that the dual conditions hold. Section F.4 gives the proofs of some lemmas which are used in Section F.3. For convenience, we consider the general Bernoulli sampling. Specifically, for  $\Theta_0 = \{j | \delta_j = 1\}$ ,  $\delta_j$  are i.i.d. Bernoulli variables and take value 1 with probability  $\rho$  and 0 with probability  $1 - \rho$ . Thus, we denote the Bernoulli sampling by  $\Theta_0 \sim \text{Ber}(\rho)$ .

### F.1 Equivalent Conditions

**Theorem 3.** *Let the pair  $(\mathcal{Z}', \mathcal{E}')$  satisfy  $\mathcal{X} = \mathcal{X} *_L \mathcal{Z}' + \mathcal{E}'$ . Denote the skinny t-SVD of  $\mathcal{Z}'$  as  $\mathcal{U}' *_L \mathcal{S}' *_L \mathcal{V}'^H$ , and the column support of  $\mathcal{E}'$  as  $\Theta'$ . If  $\mathcal{P}_{\mathcal{V}_0}^L(\mathcal{Z}') = \mathcal{Z}'$  and  $\mathcal{P}_{\Theta_0}(\mathcal{E}') = \mathcal{E}'$ , then  $\mathcal{U}' *_L \mathcal{U}'^H = \mathcal{V}_0 *_L \mathcal{V}_0^H$  and  $\Theta' = \Theta_0$ .*

*Proof.* Since  $\mathcal{P}_{\mathcal{V}_0}^L(\mathcal{Z}') = \mathcal{Z}'$  implies that  $\mathcal{U}'$  is a subspace of  $\mathcal{V}_0$ , to prove  $\mathcal{U}' *_L \mathcal{U}'^H = \mathcal{V}_0 *_L \mathcal{V}_0^H$ , it is sufficient to prove  $\text{rank}_t(\mathcal{Z}') \geq \text{rank}_t(\mathcal{L}_0)$ . Note that  $\mathcal{P}_{\Theta_0^\perp}(\mathcal{X}) = \mathcal{L}_0$ . We have

$$\mathcal{L}_0 = \mathcal{P}_{\Theta_0^\perp}(\mathcal{X}) = \mathcal{P}_{\Theta_0^\perp}(\mathcal{X} *_L \mathcal{Z}' + \mathcal{E}') = \mathcal{P}_{\Theta_0^\perp}(\mathcal{X} *_L \mathcal{Z}') = \mathcal{X} *_L \mathcal{P}_{\Theta_0^\perp}(\mathcal{Z}').$$

Therefore, we can establish that

$$\text{rank}_t(\mathcal{L}_0) = \text{rank}_t(\mathcal{X} *_L \mathcal{P}_{\Theta_0^\perp}(\mathcal{Z}')) \leq \text{rank}_t(\mathcal{P}_{\Theta_0^\perp}(\mathcal{Z}')) \leq \text{rank}_t(\mathcal{Z}').$$

Next, since  $\mathcal{P}_{\Theta_0}(\mathcal{E}') = \mathcal{E}'$  implies that  $\Theta' \subseteq \Theta_0$ , if we can prove  $\Theta_0 \cap (\Theta')^c = \emptyset$ , then we can have  $\Theta' = \Theta_0$ . In order to do so, we first have

$$\begin{aligned} \mathcal{P}_{\Theta_0}(\mathcal{L}_0) = \mathbf{0} &\Rightarrow \mathcal{U}_0 *_L \mathcal{S}_0 *_L \mathcal{P}_{\Theta_0}(\mathcal{V}_0^H) = \mathbf{0} \\ &\Rightarrow \mathcal{P}_{\Theta_0}(\mathcal{V}_0^H) = \mathbf{0} \\ &\Rightarrow \mathcal{V}_0 *_L \mathcal{P}_{\Theta_0}(\mathcal{V}_0^H) = \mathbf{0}. \end{aligned}$$

Also,  $\mathbf{V}_0 \in \mathcal{P}_{\mathbf{V}_X}^L$  implies that  $\mathbf{V}_0 *_{\mathbf{L}} \mathbf{V}_0^H = \mathbf{V}_0 *_{\mathbf{L}} \mathbf{V}_0^H *_{\mathbf{L}} \mathbf{V}_X *_{\mathbf{L}} \mathbf{V}_X^H$ , which then leads to  $\mathbf{V}_0 *_{\mathbf{L}} \mathbf{V}_0^H *_{\mathbf{L}} \mathbf{V}_X *_{\mathbf{L}} \mathbf{V}_X^H = \mathbf{P}_{\Theta_0}(\mathbf{V}_X^H) = \mathbf{V}_0 *_{\mathbf{L}} \mathbf{P}_{\Theta_0}(\mathbf{V}_0^H) = \mathbf{0}$ . Because  $\Theta_0 \cap (\Theta')^c \subseteq \Theta_0$ , we can obtain  $\mathbf{V}_0 *_{\mathbf{L}} \mathbf{V}_0^H *_{\mathbf{L}} \mathbf{V}_X *_{\mathbf{L}} \mathbf{P}_{\Theta_0 \cap (\Theta')^c}(\mathbf{V}_X^H) = \mathbf{0}$ . On the other hand, note that  $\Theta_0 \cap (\Theta')^c \subseteq (\Theta')^c$ , we have the following:

$$\begin{aligned}
 \mathbf{X} &= \mathbf{X} *_{\mathbf{L}} \mathbf{Z}' + \mathcal{E}' \\
 \Rightarrow \mathcal{P}_{\Theta_0 \cap (\Theta')^c}(\mathbf{X}) &= \mathbf{X} *_{\mathbf{L}} \mathcal{P}_{\Theta_0 \cap (\Theta')^c}(\mathbf{Z}') \\
 \Rightarrow \mathbf{U}_X *_{\mathbf{L}} \mathcal{S}_X *_{\mathbf{L}} \mathcal{P}_{\Theta_0 \cap (\Theta')^c}(\mathbf{V}_X^H) & \\
 &= \mathbf{U}_X *_{\mathbf{L}} \mathcal{S}_X *_{\mathbf{L}} \mathbf{V}_X^H *_{\mathbf{L}} \mathcal{P}_{\Theta_0 \cap (\Theta')^c}(\mathbf{Z}') \\
 \Rightarrow \mathcal{P}_{\Theta_0 \cap (\Theta')^c}(\mathbf{V}_X^H) &= \mathbf{V}_X^H *_{\mathbf{L}} \mathcal{P}_{\Theta_0 \cap (\Theta')^c}(\mathbf{Z}') \\
 \Rightarrow \mathbf{V}_0 *_{\mathbf{L}} \mathbf{V}_0^H *_{\mathbf{L}} \mathbf{V}_X *_{\mathbf{L}} \mathcal{P}_{\Theta_0 \cap (\Theta')^c}(\mathbf{V}_X^H) & \\
 &= \mathbf{V}_0 *_{\mathbf{L}} \mathbf{V}_0^H *_{\mathbf{L}} \mathbf{V}_X *_{\mathbf{L}} \mathbf{V}_X^H *_{\mathbf{L}} \mathcal{P}_{\Theta_0 \cap (\Theta')^c}(\mathbf{Z}') \\
 \Rightarrow \mathbf{V}_0 *_{\mathbf{L}} \mathbf{V}_0^H *_{\mathbf{L}} \mathbf{V}_X *_{\mathbf{L}} \mathbf{V}_X^H *_{\mathbf{L}} \mathcal{P}_{\Theta_0 \cap (\Theta')^c}(\mathbf{Z}') &= \mathbf{0} \\
 \stackrel{\textcircled{1}}{\Rightarrow} \mathbf{V}_0 *_{\mathbf{L}} \mathbf{V}_0^H *_{\mathbf{L}} \mathcal{P}_{\Theta_0 \cap (\Theta')^c}(\mathbf{Z}') &= \mathbf{0} \\
 \stackrel{\textcircled{2}}{\Rightarrow} \mathcal{P}_{\Theta_0 \cap (\Theta')^c}(\mathbf{Z}') &= \mathbf{0},
 \end{aligned}$$

where  $\textcircled{1}$  holds because  $\mathbf{Z}' \in \mathcal{P}_{\mathbf{V}_X}^L$  and  $\textcircled{2}$  holds because  $\mathbf{Z}' = \mathbf{V}_0 *_{\mathbf{L}} \mathbf{V}_0^H *_{\mathbf{L}} \mathbf{Z}'$ . It follows from  $\mathbf{X} = \mathcal{L}_0 + \mathcal{E}_0$  that  $\mathcal{P}_{\Theta_0 \cap (\Theta')^c}(\mathcal{E}_0) = \mathcal{P}_{\Theta_0 \cap (\Theta')^c}(\mathbf{X} - \mathcal{L}_0) = \mathcal{P}_{\Theta_0 \cap (\Theta')^c}(\mathbf{X}) = \mathbf{X} *_{\mathbf{L}} \mathcal{P}_{\Theta_0 \cap (\Theta')^c}(\mathbf{Z}') = \mathbf{0}$ . Hence,  $\Theta_0 \cap (\Theta')^c = \emptyset$ .  $\square$

This theorem implies that the exact recovery is equivalent to two constraints:  $\mathcal{P}_{\mathbf{V}_0}^L(\mathbf{Z}) = \mathbf{Z}$  and  $\mathcal{P}_{\Theta_0}(\mathcal{E}) = \mathcal{E}$ . As will be seen in the following, this can largely facilitate the proof of Theorem 2.

## F.2 Dual Conditions

We now prove the dual conditions of the OR-TLRR problem (4).

**Lemma 4** (Dual conditions for OR-TLRR). *Assume that  $\text{Range}(\mathcal{L}_0) = \text{Range}(\mathcal{P}_{\Theta_0}(\mathcal{L}_0))$ ,  $\mathcal{E}_0(:, j, :) \notin \text{Range}(\mathcal{L}_0)$  for  $j \in \Theta_0$ , and  $(\mathbf{Z}_*, \mathcal{E}_*) = (\mathbf{V}_0 *_{\mathbf{L}} \mathbf{V}_0^H + \mathcal{H}, \mathcal{E}_0 - \mathbf{X} *_{\mathbf{L}} \mathcal{H})$  is an arbitrary solution to (4). Let  $\hat{\mathbf{Z}} = \mathbf{V}_0 *_{\mathbf{L}} \mathbf{V}_0^H + \mathcal{P}_{\Theta_0} \mathcal{P}_{\mathbf{V}_0}^L(\mathcal{H})$  and  $\hat{\mathcal{E}} = \mathcal{E}_0 - \mathbf{X} *_{\mathbf{L}} \mathcal{P}_{\Theta_0} \mathcal{P}_{\mathbf{V}_0}^L(\mathcal{H})$ . Suppose that  $\|\mathcal{P}_{\Theta} \mathcal{P}_{\mathbf{V}}\| < 1$ ,  $\lambda > 2\sqrt{\frac{\mu r}{n_2 \tau}}$ , and  $\hat{\mathcal{L}} = \mathbf{X} *_{\mathbf{L}} \hat{\mathbf{Z}}$  obeys the tensor column-incoherence condition. Then the column space of  $\mathbf{Z}_*$  is the same as the row space of  $\mathcal{L}_0$  and  $\mathcal{E}_*$  has the same column indices as those of  $\mathcal{E}_0$ , provided that there exists a pair  $(\mathcal{W}, \mathcal{F})$  obeying*

$$\mathcal{W} = \lambda(\mathbf{X}^* *_{\mathbf{L}} \mathcal{B}(\mathcal{E}) + \mathcal{F}),$$

with  $\mathcal{P}_{\mathbf{V}}(\mathcal{W}) = \mathbf{0}$ ,  $\|\mathcal{W}\| \leq 1/2$ ,  $\mathcal{P}_{\Theta}(\mathcal{F}) = \mathbf{0}$  and  $\|\mathcal{F}\|_{2, \infty} \leq 1/2$ .

*Proof.* The subgradients of tensor nuclear norm and tensor  $\ell_{2,1}$  norm can be written as

$$\begin{aligned}
 \partial_{\hat{\mathbf{Z}}} \|\hat{\mathbf{Z}}\|_* &= \{\hat{\mathbf{U}} *_{\mathbf{L}} \hat{\mathbf{V}}^H + \hat{\mathbf{Q}} | \mathcal{P}_{\hat{\mathbf{T}}}(\hat{\mathbf{Q}}) = \mathbf{0}, \|\hat{\mathbf{Q}}\| \leq 1\}, \\
 \partial_{\hat{\mathcal{E}}} \|\hat{\mathcal{E}}\|_{2,1} &= \{\mathcal{B}(\hat{\mathcal{E}}) + \hat{\mathcal{C}} | \mathcal{P}_{\Theta}(\hat{\mathcal{C}}) = \mathbf{0}, \|\hat{\mathcal{C}}\|_{2, \infty} \leq 1\}.
 \end{aligned}$$

We denote the skinny t-SVDs of  $\hat{\mathbf{Z}}$  and  $\hat{\mathcal{L}}$  as  $\hat{\mathbf{U}} *_{\mathbf{L}} \hat{\mathbf{S}} *_{\mathbf{L}} \hat{\mathbf{V}}^H$  and  $\mathbf{U}_{\hat{\mathcal{L}}} *_{\mathbf{L}} \mathcal{S}_{\hat{\mathcal{L}}} *_{\mathbf{L}} \mathbf{V}_{\hat{\mathcal{L}}}^H$ , respectively. Since  $\hat{\mathcal{L}} = \mathbf{X} *_{\mathbf{L}} \hat{\mathbf{Z}}$ , we have

$$\mathbf{V}_{\hat{\mathcal{L}}}^H = \mathcal{S}_{\hat{\mathcal{L}}}^\dagger *_{\mathbf{L}} \mathbf{U}_{\hat{\mathcal{L}}}^H *_{\mathbf{L}} \mathbf{X} *_{\mathbf{L}} \hat{\mathbf{U}} *_{\mathbf{L}} \hat{\mathbf{S}} *_{\mathbf{L}} \hat{\mathbf{V}}^H,$$

which implies that the horizontal slices, i.e., the tensor rows of  $\hat{\mathbf{V}}^H$  span the tensor rows of  $\mathbf{V}_{\hat{\mathcal{L}}}^H$ . Notice that  $\mathbf{V}_{\hat{\mathcal{L}}}^H$  is row orthonormal, there exists a row orthonormal tensor  $\mathcal{R}$  such that  $\hat{\mathbf{V}}^H = \mathcal{R} *_{\mathbf{L}} \mathbf{V}_{\hat{\mathcal{L}}}^H$ . Then we have

$$\begin{aligned}
 \|\hat{\mathbf{U}} * \hat{\mathbf{V}}^H\|_{2, \infty} &= \max_j \|\hat{\mathbf{U}} *_{\mathbf{L}} \hat{\mathbf{V}}^H *_{\mathbf{L}} \hat{\mathbf{e}}_j\|_F = \max_j \|\hat{\mathbf{V}}^H *_{\mathbf{L}} \hat{\mathbf{e}}_j\|_F \\
 &= \max_j \|\mathcal{R} *_{\mathbf{L}} \mathbf{V}_{\hat{\mathcal{L}}}^H *_{\mathbf{L}} \hat{\mathbf{e}}_j\|_F \leq \max_j \|\mathbf{V}_{\hat{\mathcal{L}}}^H *_{\mathbf{L}} \hat{\mathbf{e}}_j\|_F \leq \sqrt{\frac{\mu r}{n_2 \tau}}.
 \end{aligned}$$

Let  $\mathcal{H}_1 = \mathcal{P}_{\Theta_0} \mathcal{P}_{\mathcal{V}_0^L}(\mathcal{H})$  and  $\mathcal{H}_2 = \mathcal{P}_{\Theta_0^\perp} \mathcal{P}_{\mathcal{V}_0^L}(\mathcal{H}) + \mathcal{P}_{\Theta_0^\perp} \mathcal{P}_{\mathcal{V}_0^\perp}(\mathcal{H}) + \mathcal{P}_{\Theta_0} \mathcal{P}_{\mathcal{V}_0^\perp}(\mathcal{H})$ . Note that  $(\hat{\mathcal{Z}}, \hat{\mathcal{E}})$  satisfies  $\mathcal{X} = \mathcal{X} *_L \hat{\mathcal{Z}} + \hat{\mathcal{E}}$ ,  $\mathcal{P}_{\mathcal{V}_0^L}(\hat{\mathcal{Z}}) = \hat{\mathcal{Z}}$  and  $\mathcal{P}_{\Theta_0}(\hat{\mathcal{E}}) = \hat{\mathcal{E}}$ . Then by Theorem 3, we have  $\mathcal{P}_{\hat{\mathcal{U}}} = \mathcal{P}_{\mathcal{V}_0^L}$  and  $\hat{\Theta} = \Theta_0$ . We can obtain that

$$\begin{aligned}
 & \|\mathcal{Z}_*\|_* + \lambda \|\mathcal{E}_*\|_{2,1} - \|\hat{\mathcal{Z}}\|_* - \lambda \|\hat{\mathcal{E}}\|_{2,1} \\
 & \geq \langle \hat{\mathcal{U}} *_L \hat{\mathcal{V}}^H + \hat{\mathcal{Q}}, \mathcal{Z}_* - \hat{\mathcal{Z}} \rangle + \lambda \langle \mathcal{B}(\hat{\mathcal{E}}) + \hat{\mathcal{C}}, \mathcal{E}_* - \hat{\mathcal{E}} \rangle \\
 & = \langle \hat{\mathcal{U}} *_L \hat{\mathcal{V}}^H + \hat{\mathcal{Q}}, \mathcal{H}_2 \rangle - \lambda \langle \mathcal{B}(\hat{\mathcal{E}}) + \hat{\mathcal{C}}, \mathcal{X}^H *_L \mathcal{H}_2 \rangle \\
 & = \langle \hat{\mathcal{U}} *_L \hat{\mathcal{V}}^H, \mathcal{P}_{\Theta_0^\perp}(\mathcal{H}) \rangle + \langle \hat{\mathcal{Q}}, \mathcal{P}_{\mathcal{V}_0^L}(\mathcal{H}) \rangle - \lambda \langle \mathcal{B}(\hat{\mathcal{E}}), \mathcal{X} *_L \mathcal{P}_{\mathcal{V}_0^\perp}(\mathcal{H}) \rangle - \lambda \langle \hat{\mathcal{C}}, \mathcal{X} *_L \mathcal{P}_{\Theta_0^\perp}(\mathcal{H}) \rangle \\
 & = \langle \hat{\mathcal{U}} *_L \hat{\mathcal{V}}^H, \mathcal{P}_{\Theta_0^\perp}(\mathcal{H}) \rangle + \langle \hat{\mathcal{Q}}, \mathcal{P}_{\mathcal{V}_0^L}(\mathcal{H}) \rangle - \lambda \langle \mathcal{X}^H *_L \mathcal{B}(\hat{\mathcal{E}}), \mathcal{P}_{\mathcal{V}_0^\perp}(\mathcal{H}) \rangle - \lambda \langle \mathcal{X}^H *_L \hat{\mathcal{C}}, \mathcal{P}_{\Theta_0^\perp}(\mathcal{H}) \rangle.
 \end{aligned}$$

By the duality between the tensor nuclear norm and the tensor spectral norm, there exists a tensor  $\mathcal{Q}_0$  such that  $\langle \mathcal{Q}_0, \mathcal{P}_{\hat{\mathcal{V}}^\perp} \mathcal{P}_{\mathcal{V}_0^\perp}^L(\mathcal{H}) \rangle = \|\mathcal{P}_{\hat{\mathcal{V}}^\perp} \mathcal{P}_{\mathcal{V}_0^\perp}^L(\mathcal{H})\|_*$  and  $\|\mathcal{Q}_0\| = 1$ . We set  $\hat{\mathcal{Q}} = \mathcal{P}_{\mathcal{V}_0^\perp}^L \mathcal{P}_{\hat{\mathcal{V}}^\perp}(\mathcal{Q}_0) \in \mathcal{P}_{\mathcal{T}^\perp}$ . Similarly, thanks to the duality between the  $\ell_{2,1}$  norm and  $\ell_{2,\infty}$  norm, we can pick  $\hat{\mathcal{C}} \in \mathcal{P}_{\Theta_0^\perp}$  such that  $\langle \mathcal{X}^H *_L \hat{\mathcal{C}}, \mathcal{P}_{\Theta_0^\perp}(\mathcal{H}) \rangle = -\|\mathcal{P}_{\Theta_0^\perp}(\mathcal{H})\|_{2,1}$ . On the other hand, we can establish that

$$\begin{aligned}
 & |\langle \mathcal{X}^H *_L \mathcal{B}(\hat{\mathcal{E}}), \mathcal{P}_{\mathcal{V}_0^\perp}^L(\mathcal{H}) \rangle| \\
 & = |\langle \mathcal{W} - \lambda \mathcal{F}, \mathcal{P}_{\mathcal{V}_0^\perp}^L(\mathcal{H}) \rangle| \\
 & \leq |\langle \mathcal{W}, \mathcal{P}_{\mathcal{V}_0^\perp}^L(\mathcal{H}) \rangle| + \lambda |\langle \mathcal{F}, \mathcal{P}_{\mathcal{V}_0^\perp}^L(\mathcal{H}) \rangle| \\
 & \leq \frac{1}{2} \|\mathcal{P}_{\hat{\mathcal{V}}^\perp} \mathcal{P}_{\mathcal{V}_0^\perp}^L(\mathcal{H})\|_* + \frac{\lambda}{2} \|\mathcal{P}_{\Theta_0^\perp} \mathcal{P}_{\mathcal{V}_0^\perp}^L(\mathcal{H})\|_{2,1},
 \end{aligned}$$

where the last inequality holds because  $\mathcal{W} \in \mathcal{P}_{\hat{\mathcal{V}}^\perp}$  and  $\mathcal{F} \in \mathcal{P}_{\Theta_0^\perp}$ . Hence, we have

$$\begin{aligned}
 & \|\mathcal{Z}_*\|_* + \lambda \|\mathcal{E}_*\|_{2,1} - \|\hat{\mathcal{Z}}\|_* - \lambda \|\hat{\mathcal{E}}\|_{2,1} \\
 & \geq \|\mathcal{P}_{\hat{\mathcal{V}}^\perp} \mathcal{P}_{\mathcal{V}_0^\perp}^L(\mathcal{H})\|_* - \sqrt{\frac{\mu r}{n_2 \tau}} \|\mathcal{P}_{\Theta_0^\perp}(\mathcal{H})\|_{2,1} + \lambda \|\mathcal{P}_{\Theta_0^\perp}(\mathcal{H})\|_{2,1} - \frac{1}{2} \|\mathcal{P}_{\hat{\mathcal{V}}^\perp} \mathcal{P}_{\mathcal{V}_0^\perp}^L(\mathcal{H})\|_* - \frac{\lambda}{2} \|\mathcal{P}_{\Theta_0^\perp} \mathcal{P}_{\mathcal{V}_0^\perp}^L(\mathcal{H})\|_{2,1} \\
 & = \frac{1}{2} \|\mathcal{P}_{\hat{\mathcal{V}}^\perp} \mathcal{P}_{\mathcal{V}_0^\perp}^L(\mathcal{H})\|_* - \sqrt{\frac{\mu r}{n_2 \tau}} \|\mathcal{P}_{\Theta_0^\perp}(\mathcal{H})\|_{2,1} + \lambda \|\mathcal{P}_{\Theta_0^\perp}(\mathcal{H})\|_{2,1} - \frac{\lambda}{2} \|\mathcal{P}_{\Theta_0^\perp} \mathcal{P}_{\mathcal{V}_0^\perp}^L(\mathcal{H})\|_{2,1} \\
 & \geq \frac{1}{2} \|\mathcal{P}_{\hat{\mathcal{V}}^\perp} \mathcal{P}_{\mathcal{V}_0^\perp}^L(\mathcal{H})\|_* + \left( \frac{\lambda}{2} - \sqrt{\frac{\mu r}{n_2 \tau}} \right) \|\mathcal{P}_{\Theta_0^\perp}(\mathcal{H})\|_{2,1}.
 \end{aligned}$$

Note that  $(\mathcal{Z}_*, \mathcal{E}_*) = (\mathcal{V}_0 *_L \mathcal{V}_0^H + \mathcal{H}, \mathcal{E}_0 - \mathcal{X} *_L \mathcal{H})$  is an arbitrary optimal solution, the above inequality shows  $\|\mathcal{P}_{\hat{\mathcal{V}}^\perp} \mathcal{P}_{\mathcal{V}_0^\perp}^L(\mathcal{H})\|_* = \|\mathcal{P}_{\Theta_0^\perp} \mathcal{P}_{\mathcal{V}_0^\perp}^L(\mathcal{H})\|_{2,1} = \|\mathcal{P}_{\Theta_0^\perp}(\mathcal{H})\|_{2,1} = 0$ , i.e.,  $\mathcal{P}_{\mathcal{V}_0^\perp}^L(\mathcal{H}) \in \mathcal{P}_{\hat{\mathcal{V}}} \cap \mathcal{P}_{\Theta_0}$  and  $\mathcal{H} \in \mathcal{P}_{\Theta_0}$ . By the assumption that  $\|\mathcal{P}_{\Theta_0} \mathcal{P}_{\hat{\mathcal{V}}}\| < 1$ , we have  $\mathcal{P}_{\mathcal{V}_0^\perp}^L(\mathcal{H}) = \mathbf{0}$ , and so  $\mathcal{H} \in \mathcal{P}_{\mathcal{V}_0^L}$ . Thus, the solution  $(\mathcal{Z}_*, \mathcal{E}_*)$  also satisfies  $\mathcal{P}_{\mathcal{V}_0^L}(\mathcal{Z}_*) = \mathcal{Z}_*$  and  $\mathcal{P}_{\Theta_0}(\mathcal{E}_*) = \mathcal{E}_*$ . It can be concluded that the solution  $(\mathcal{Z}_*, \mathcal{E}_*)$  also exactly recovers the row space of  $\mathcal{L}_0$  and the column support of  $\mathcal{E}_0$ .  $\square$

Lemma 4 implies that if we can find a dual certificate  $\mathcal{W}$  obeying

$$\begin{aligned}
 (a) \quad & \mathcal{W} \in \mathcal{P}_{\hat{\mathcal{V}}^\perp}, \\
 (b) \quad & \mathcal{P}_{\Theta_0}(\mathcal{W}) = \lambda \mathcal{X}^H *_L \mathcal{B}(\hat{\mathcal{E}}), \\
 (c) \quad & \|\mathcal{W}\| \leq 1/2, \\
 (d) \quad & \|\mathcal{P}_{\Theta_0^\perp}(\mathcal{W})\|_{2,\infty} \leq \lambda/2,
 \end{aligned} \tag{31}$$

then we can exactly recover the row space of  $\mathcal{L}_0$  and the column support of outlier tensor  $\mathcal{E}_0$ .

### F.3 Dual Certification via the Least Squares

Before we construct the dual certificate  $\mathcal{W}$ , we first give some key lemmas, which will be proved in Section F.4.

**Lemma 5.** Assume  $\hat{\Theta} \sim \text{Ber}(\rho)$ . Then with probability at least  $1 - n_{(1)}^{-10}$ ,

$$\|\mathcal{P}_{\hat{\mathbf{V}}} - \frac{1}{\rho}\mathcal{P}_{\hat{\mathbf{V}}}\mathcal{P}_{\hat{\Theta}}\mathcal{P}_{\hat{\mathbf{V}}}\| \leq \epsilon,$$

provided that  $\rho \geq c_2\mu rn_3 \log(n_{(1)})/(\epsilon^2 n_2)$  for some numerical constant  $c_2$ .

**Corollary 1.** Assume  $\hat{\Theta} \sim \text{Ber}(\rho)$ . Then with probability at least  $1 - n_{(1)}^{-10}$ ,

$$\|\mathcal{P}_{\hat{\Theta}}\mathcal{P}_{\hat{\mathbf{V}}}\|^2 \leq (1 - \rho)\epsilon + \rho < 1,$$

provided that  $1 - \rho \geq c_2\mu rn_3 \log(n_{(1)})/(\epsilon^2 n_2)$  for some numerical constant  $c_2$ .

Now we construct the dual certificate  $\mathcal{W}$  and verify its validity.

**Lemma 6.** Suppose that  $\hat{\Theta} \sim \text{Ber}(\rho)$  and the assumptions in Theorem 2 are satisfied. Then with high probability,

$$\mathcal{W} = \lambda \mathcal{P}_{\hat{\mathbf{V}}^\perp} \sum_{k=0}^{+\infty} (\mathcal{P}_{\hat{\Theta}}\mathcal{P}_{\hat{\mathbf{V}}}\mathcal{P}_{\hat{\Theta}})^k (\mathcal{X}^H *_L \mathcal{B}(\hat{\mathcal{E}}))$$

obeys the dual conditions (31).

*Proof.* Note that by Corollary 1, we have  $\|\mathcal{P}_{\hat{\Theta}}\mathcal{P}_{\hat{\mathbf{V}}}\mathcal{P}_{\hat{\Theta}}\| = \|\mathcal{P}_{\hat{\Theta}}\mathcal{P}_{\hat{\mathbf{V}}}\|^2 < 1$ . Thus,  $\mathcal{W}$  is well defined. Now we verify the four conditions in (31) in turn.

**Proof of (a):** It is easy to verify that  $\mathcal{W} \in \mathcal{P}_{\hat{\mathbf{V}}^\perp}$ .

**Proof of (b):** By the construction of  $\mathcal{W}$ , we have

$$\begin{aligned} \mathcal{P}_{\hat{\Theta}}(\mathcal{W}) &= \lambda \mathcal{P}_{\hat{\Theta}}\mathcal{P}_{\hat{\mathbf{V}}^\perp} \sum_{k=0}^{+\infty} (\mathcal{P}_{\hat{\Theta}}\mathcal{P}_{\hat{\mathbf{V}}}\mathcal{P}_{\hat{\Theta}})^k (\mathcal{X}^H *_L \mathcal{B}(\hat{\mathcal{E}})) \\ &= \lambda \mathcal{P}_{\hat{\Theta}}(\mathcal{I}_{n_2} - \mathcal{P}_{\hat{\mathbf{V}}}) \sum_{k=0}^{+\infty} (\mathcal{P}_{\hat{\Theta}}\mathcal{P}_{\hat{\mathbf{V}}}\mathcal{P}_{\hat{\Theta}})^k (\mathcal{X}^H *_L \mathcal{B}(\hat{\mathcal{E}})) \\ &= \lambda \mathcal{P}_{\hat{\Theta}}(\mathcal{I}_{n_2} - \mathcal{P}_{\hat{\Theta}}\mathcal{P}_{\hat{\mathbf{V}}}\mathcal{P}_{\hat{\Theta}}) \sum_{k=0}^{+\infty} (\mathcal{P}_{\hat{\Theta}}\mathcal{P}_{\hat{\mathbf{V}}}\mathcal{P}_{\hat{\Theta}})^k (\mathcal{X}^H *_L \mathcal{B}(\hat{\mathcal{E}})) \\ &= \lambda \mathcal{P}_{\hat{\Theta}} \left( \sum_{k=0}^{+\infty} (\mathcal{P}_{\hat{\Theta}}\mathcal{P}_{\hat{\mathbf{V}}}\mathcal{P}_{\hat{\Theta}})^k - \sum_{k=1}^{+\infty} (\mathcal{P}_{\hat{\Theta}}\mathcal{P}_{\hat{\mathbf{V}}}\mathcal{P}_{\hat{\Theta}})^k \right) (\mathcal{X}^H *_L \mathcal{B}(\hat{\mathcal{E}})) \\ &= \lambda \mathcal{P}_{\hat{\Theta}}(\mathcal{X}^H *_L \mathcal{B}(\hat{\mathcal{E}})) \\ &= \lambda \mathcal{X}^H *_L \mathcal{B}(\hat{\mathcal{E}}). \end{aligned}$$

Thus,  $\mathcal{W}$  obeys the condition (31)(b).

**Proof of (c):** Let  $\mathcal{G} = \sum_{k=0}^{+\infty} (\mathcal{P}_{\hat{\Theta}}\mathcal{P}_{\hat{\mathbf{V}}}\mathcal{P}_{\hat{\Theta}})^k$ . Since  $\|\mathcal{B}(\hat{\mathcal{E}})\| \leq \sqrt{\log(n_2)}/\theta$ , we have

$$\|\mathcal{W}\| \leq \lambda \|\mathcal{P}_{\hat{\mathbf{V}}^\perp}\| \|\mathcal{G}\| \|\mathcal{X}\| \|\mathcal{B}(\hat{\mathcal{E}})\| \leq \frac{\lambda}{1 - \sigma^2} \|\mathcal{X}\| \frac{\sqrt{\log(n_2)}}{\theta},$$

where  $\sigma = \sqrt{\rho + \epsilon(1 - \rho)}$ . If  $\lambda \leq \frac{\theta(1 - \sigma^2)}{2\|\mathcal{X}\|\sqrt{\log(n_2)}}$ , then  $\|\mathcal{W}\| \leq 1/2$ . Note that in Lemma 4, we require that  $\lambda > 2\sqrt{\frac{\mu r}{n_2 \tau}}$ . Thus,  $\lambda \in \left( 2\sqrt{\frac{\mu r}{n_2 \tau}}, \frac{\theta(1 - \sigma^2)}{2\|\mathcal{X}\|\sqrt{\log(n_2)}} \right]$ .

**Proof of (d):** We can rewrite  $\mathcal{P}_{\hat{\Theta}^\perp}(\mathcal{W})$  as

$$\begin{aligned} \mathcal{P}_{\hat{\Theta}^\perp}(\mathcal{W}) &= \lambda \mathcal{P}_{\hat{\Theta}^\perp}\mathcal{P}_{\hat{\mathbf{V}}^\perp}\mathcal{G}(\mathcal{X}^H *_L \mathcal{B}(\hat{\mathcal{E}})) \\ &= \lambda \mathcal{P}_{\hat{\Theta}^\perp}(\mathcal{I}_{n_2} - \mathcal{P}_{\hat{\mathbf{V}}})\mathcal{G}(\mathcal{X}^H *_L \mathcal{B}(\hat{\mathcal{E}})) \\ &= -\lambda \mathcal{P}_{\hat{\Theta}^\perp}\mathcal{P}_{\hat{\mathbf{V}}}\mathcal{G}(\mathcal{X}^H *_L \mathcal{B}(\hat{\mathcal{E}})). \end{aligned}$$

Note that  $\bar{\mathbf{e}}_{ijk}$  is the unit tensor with the  $(i, j, k)$ -th entry equaling to 1 and the rest equaling to 0. Thus, the  $(i, j)$ -th tube of  $\bar{\mathbf{e}}_{ijk}$  is the only nonzero tube and it equals  $L(\hat{\mathbf{e}}_k)$ , which further suggests that the  $(i, j)$ -th tube of  $L(\bar{\mathbf{e}}_{ijk})$  equals  $L(L(\hat{\mathbf{e}}_k))$ . We have

$$\begin{aligned}
 & \max_j \sum_{i,k} \|\mathcal{P}_{\hat{\mathbf{V}}}(\bar{\mathbf{e}}_{ijk})\|_F^2 \\
 &= \max_j \sum_{i,k} \|L(\hat{\mathbf{e}}_i *_{\mathcal{L}} \hat{\mathbf{e}}_k *_{\mathcal{L}} \hat{\mathbf{e}}_j^H) *_{\mathcal{L}} \hat{\mathbf{V}} *_{\mathcal{L}} \hat{\mathbf{V}}^H\|_F^2 \\
 &= \max_j \sum_{i,k} \|L(\hat{\mathbf{e}}_i *_{\mathcal{L}} \hat{\mathbf{e}}_k *_{\mathcal{L}} \hat{\mathbf{e}}_j^H) *_{\mathcal{L}} \hat{\mathbf{V}}\|_F^2 \\
 &= \max_j \sum_{i,k} \frac{1}{\tau} \|L(\hat{\mathbf{e}}_i) \odot L(L(\hat{\mathbf{e}}_k)) \odot L(\hat{\mathbf{e}}_j^H) \odot L(\hat{\mathbf{V}})\|_F^2 \\
 &= \max_j \sum_{i,k} \frac{1}{\tau} \|L(L(\hat{\mathbf{e}}_k)) \odot L(\hat{\mathbf{e}}_j^H) \odot L(\hat{\mathbf{V}})\|_F^2.
 \end{aligned}$$

Define  $\mathcal{Q} = \hat{\mathbf{e}}_j^H *_{\mathcal{L}} \hat{\mathbf{V}}$ . Then  $\|L(L(\hat{\mathbf{e}}_k)) \odot \mathcal{Q}\|_F^2 = \sum_q |\mathbf{L}_{q,k}|^2 \|\bar{\mathcal{Q}}(:, :, q)\|_F^2 \leq \sum_q (\max_q |\mathbf{L}_{q,k}|^2) \|\bar{\mathcal{Q}}(:, :, q)\|_F^2 \leq \tau \sum_q \|\bar{\mathcal{Q}}(:, :, q)\|_F^2 = \tau \|\bar{\mathcal{Q}}\|_F^2 = \tau^2 \|\mathcal{Q}\|_F^2$ . Hence,

$$\max_j \sum_{i,k} \|\mathcal{P}_{\hat{\mathbf{V}}}(\bar{\mathbf{e}}_{ijk})\|_F^2 \leq \tau \max_j \sum_{i,k} \|\hat{\mathbf{e}}_j^H *_{\mathcal{L}} \hat{\mathbf{V}}\|_F^2 = \frac{\mu r n_1 n_3}{n_2}.$$

Then, we can obtain

$$\begin{aligned}
 & \|\mathcal{P}_{\hat{\mathbf{V}}}\mathcal{G}(\mathcal{X}^H *_{\mathcal{L}} \mathcal{B}(\hat{\mathcal{E}}))\|_{2,\infty}^2 \\
 &= \max_b \sum_{i,k} \left\langle \mathcal{P}_{\hat{\mathbf{V}}}\mathcal{G}(\mathcal{X}^H *_{\mathcal{L}} \mathcal{B}(\hat{\mathcal{E}})), \bar{\mathbf{e}}_{ibk} \right\rangle^2 \\
 &= \max_b \sum_{i,k} \left\langle \mathcal{X}^H *_{\mathcal{L}} \mathcal{B}(\hat{\mathcal{E}}), \mathcal{G}\mathcal{P}_{\hat{\mathbf{V}}}(\bar{\mathbf{e}}_{ibk}) \right\rangle^2 \\
 &= \max_b \sum_{i,k} \sum_j \left\langle \mathcal{X}^H *_{\mathcal{L}} \mathcal{B}(\hat{\mathcal{E}}) *_{\mathcal{L}} \hat{\mathbf{e}}_j, \mathcal{G}\mathcal{P}_{\hat{\mathbf{V}}}(\bar{\mathbf{e}}_{ibk}) *_{\mathcal{L}} \hat{\mathbf{e}}_j \right\rangle^2 \\
 &\leq \max_b \sum_{i,j,k} \|\mathcal{X}^H *_{\mathcal{L}} \mathcal{B}(\hat{\mathcal{E}}) *_{\mathcal{L}} \hat{\mathbf{e}}_j\|_F^2 \|\mathcal{G}\mathcal{P}_{\hat{\mathbf{V}}}(\bar{\mathbf{e}}_{ibk}) *_{\mathcal{L}} \hat{\mathbf{e}}_j\|_F^2 \\
 &\leq \max_b \sum_{i,j,k} \|\mathcal{X}^H *_{\mathcal{L}} \mathcal{B}(\hat{\mathcal{E}})\|_{2,\infty}^2 \|\mathcal{G}\mathcal{P}_{\hat{\mathbf{V}}}(\bar{\mathbf{e}}_{ibk}) *_{\mathcal{L}} \hat{\mathbf{e}}_j\|_F^2 \\
 &= \max_b \sum_{i,k} \|\mathcal{X}^H *_{\mathcal{L}} \mathcal{B}(\hat{\mathcal{E}})\|_{2,\infty}^2 \|\mathcal{G}\mathcal{P}_{\hat{\mathbf{V}}}(\bar{\mathbf{e}}_{ibk})\|_F^2 \\
 &\leq \max_b \sum_{i,k} \|\mathcal{X}\|^2 \|\mathcal{G}\|^2 \|\mathcal{P}_{\hat{\mathbf{V}}}(\bar{\mathbf{e}}_{ibk})\|_F^2 \\
 &\leq \frac{\mu r n_1 n_3}{n_2} \left(\frac{1}{1-\sigma^2}\right)^2 \|\mathcal{X}\|^2 \\
 &\leq \frac{1}{4},
 \end{aligned}$$

where the last inequality holds because we require  $r \leq n_2(1-\sigma^2)^2/(4\mu n_1 n_3 \|\mathcal{X}\|^2)$ . In the meanwhile, when proving Corollary 1, we require  $r \leq (1-\rho)\epsilon^2 n_2/(c_2 \mu n_3 \log(n_1))$ . Thus, we can further obtain

$$\begin{aligned}
 \|\mathcal{P}_{\hat{\Theta}^\perp}(\mathcal{W})\|_{2,\infty} &= \lambda \|\mathcal{P}_{\hat{\Theta}^\perp} \mathcal{P}_{\hat{\mathbf{V}}}\mathcal{G}(\mathcal{X}^H *_{\mathcal{L}} \mathcal{B}(\hat{\mathcal{E}}))\|_{2,\infty} \\
 &\leq \lambda \|\mathcal{P}_{\hat{\mathbf{V}}}\mathcal{G}(\mathcal{X}^H *_{\mathcal{L}} \mathcal{B}(\hat{\mathcal{E}}))\|_{2,\infty} \\
 &= \frac{\lambda}{2}.
 \end{aligned}$$

So  $\mathcal{W}$  obeys the condition (31)(d).

**Checking the ranges of  $\lambda$  and  $r$ :** When we prove Corollary 1 and bound  $\|\mathcal{P}_{\hat{\Theta}^\perp}(\mathcal{W})\|_{2,\infty}$ , we require

$$r \leq \min\left(\frac{n_2(1-\sigma^2)^2}{4\mu n_1 n_3 \|\mathcal{X}\|^2}, \frac{(1-\rho)\epsilon^2 n_2}{c_2 \mu n_3 \log(n_{(1)})}\right),$$

Thus we have  $r \leq \frac{\rho_r n_2}{\mu n_1 n_3 \|\mathcal{X}\|^2}$ , where  $\rho_r$  is a constant. On the other hand, we require

$$\lambda \in \left(2\sqrt{\frac{\mu r}{n_2 \tau}}, \frac{\theta(1-\sigma^2)}{2\|\mathcal{X}\|\sqrt{\log(n_2)}}\right].$$

Let  $\rho \leq 1/2 - \epsilon$ , we then have  $2(1-\sigma^2) \geq 1$ . Accordingly, we can obtain

$$\lambda \in \left(\min\left(\frac{1}{\sqrt{\tau n_1 n_3} \|\mathcal{X}\|}, \frac{2\epsilon}{\sqrt{c_2 \tau n_3 \log(n_{(1)})}}\right), \frac{\theta}{4\sqrt{\log(n_2)} \|\mathcal{X}\|}\right].$$

So we can set  $\lambda = \frac{\theta}{4\sqrt{\log(n_{(1)})} \|\mathcal{X}\|}$ . The proof is completed.  $\square$

#### F.4 Proofs of Some Lemmas

We first introduce Lemma 7 which will be used in the proof of Lemma 5.

**Lemma 7.** [Tropp, 2012] Consider a finite sequence  $\{\mathbf{X}_i \in \mathbb{R}^{d_1 \times d_2}\}$  of independent, random matrices that satisfy the assumption that  $\mathbb{E}[\mathbf{X}_i] = \mathbf{0}$  and  $\|\mathbf{X}_i\| \leq \nu$ . Let  $\omega = \max(\|\sum_i \mathbb{E}[\mathbf{X}_i \mathbf{X}_i^H]\|, \|\sum_i \mathbb{E}[\mathbf{X}_i^H \mathbf{X}_i]\|)$ . Then, for any  $t \geq 0$ , we have

$$\begin{aligned} \mathbb{P}\left[\left\|\sum_i \mathbf{X}_i\right\| \geq t\right] &\leq (d_1 + d_2) \exp\left(-\frac{t^2}{2\omega + \frac{2}{3}\nu t}\right) \\ &\leq (d_1 + d_2) \exp\left(-\frac{3t^2}{8\omega}\right), \quad \text{for } t \leq \omega/\nu. \end{aligned}$$

##### F.4.1 Proof of Lemma 5

*Proof.* Define a set  $\phi = \{(\mathcal{Z}_1, \mathcal{Z}_2) \mid \|\mathcal{Z}_1\|_F \leq 1, \mathcal{Z}_2 = \pm \mathcal{Z}_1\}$ . Since  $(\rho^{-1}\mathcal{P}_{\hat{\mathbf{v}}}\mathcal{P}_{\hat{\Theta}}\mathcal{P}_{\hat{\mathbf{v}}} - \mathcal{P}_{\hat{\mathbf{v}}})$  is a self-adjoint operator, we can use the variational form of the operator norm to compute its operator norm as follows:

$$\begin{aligned} &\left\|\frac{1}{\rho}\mathcal{P}_{\hat{\mathbf{v}}}\mathcal{P}_{\hat{\Theta}}\mathcal{P}_{\hat{\mathbf{v}}} - \mathcal{P}_{\hat{\mathbf{v}}}\right\| \\ &= \sup_{\phi} \sum_{i,j,k} \left(\frac{\delta_j}{\rho} - 1\right) \langle \mathcal{P}_{\hat{\mathbf{v}}}(\mathcal{Z}_1), \bar{\mathbf{e}}_{ijk} \rangle \langle \mathcal{P}_{\hat{\mathbf{v}}}(\mathcal{Z}_2), \bar{\mathbf{e}}_{ijk} \rangle \\ &= \sup_{\phi} \sum_{i,j,k} \frac{\delta_j - \rho}{\rho\tau^2} \langle \text{bdiag}(\overline{\mathcal{P}_{\hat{\mathbf{v}}}(\bar{\mathbf{e}}_{ijk})}), \bar{\mathcal{Z}}_1 \rangle \langle \text{bdiag}(\overline{\mathcal{P}_{\hat{\mathbf{v}}}(\bar{\mathbf{e}}_{ijk})}), \bar{\mathcal{Z}}_2 \rangle \\ &= \left\| \sum_{i,j,k} \frac{\delta_j - \rho}{\rho\tau} \text{bdiag}(\overline{\mathcal{P}_{\hat{\mathbf{v}}}(\bar{\mathbf{e}}_{ijk})}) \cdot \text{bdiag}(\overline{\mathcal{P}_{\hat{\mathbf{v}}}(\bar{\mathbf{e}}_{ijk})}) \right\| \\ &= \left\| \sum_j \frac{\delta_j - \rho}{\rho\tau} \sum_{i,k} \text{bdiag}(\overline{\mathcal{P}_{\hat{\mathbf{v}}}(\bar{\mathbf{e}}_{ijk})}) \cdot \text{bdiag}(\overline{\mathcal{P}_{\hat{\mathbf{v}}}(\bar{\mathbf{e}}_{ijk})}) \right\|, \end{aligned}$$

where  $\delta_j$  obeys i.i.d. Bernoulli distribution  $\text{Ber}(\rho)$ . Define a set  $\psi = \{(\bar{\mathbf{D}}_1, \bar{\mathbf{D}}_2) \mid \|\bar{\mathbf{D}}_1\|_F \leq 1, \bar{\mathbf{D}}_2 = \pm \bar{\mathbf{D}}_1\}$ . Note that  $\text{bdiag}(\overline{\mathcal{P}_{\hat{\mathbf{v}}}(\bar{\mathbf{e}}_{ijk})})$  is a diagonal matrix and thus  $\bar{\mathbf{D}}_1$  and  $\bar{\mathbf{D}}_2$  are also diagonal matrices and there exist two tensors  $\mathcal{D}_1$  and  $\mathcal{D}_2$  such that  $\bar{\mathbf{D}}_1 = \text{bdiag}(L(\mathcal{D}_1))$  and  $\bar{\mathbf{D}}_2 = \text{bdiag}(L(\mathcal{D}_2))$ . For brevity, we further define

$$\tilde{\mathbf{H}}_j = \frac{\delta_j - \rho}{\rho\tau} \sum_{i,k} \text{bdiag}(\overline{\mathcal{P}_{\hat{\mathbf{v}}}(\bar{\mathbf{e}}_{ijk})}) \cdot \text{bdiag}(\overline{\mathcal{P}_{\hat{\mathbf{v}}}(\bar{\mathbf{e}}_{ijk})}),$$

and have

$$\|\frac{1}{\rho}\mathcal{P}_{\hat{\mathbf{V}}}\mathcal{P}_{\hat{\Theta}}\mathcal{P}_{\hat{\mathbf{V}}} - \mathcal{P}_{\hat{\mathbf{V}}}\| = \|\sum_j \tilde{\mathbf{H}}_j\|.$$

By the above definitions,  $\tilde{\mathbf{H}}_j$  is self-adjoint and  $\mathbb{E}[\tilde{\mathbf{H}}_j] = \mathbf{0}$ . To prove the result by the non-commutative Bernstein inequality, we need to bound  $\|\tilde{\mathbf{H}}_j\|$  and  $\|\sum_j \mathbb{E}[\tilde{\mathbf{H}}_j\tilde{\mathbf{H}}_j^H]\|$ . Firstly, we give an useful inequality:

$$\begin{aligned} & \|\mathcal{P}_{\hat{\mathbf{V}}}(\mathcal{A})\|_{2,\infty}^2 \\ &= \max_j \sum_{i,k} \langle \mathcal{A}, \mathcal{P}_{\hat{\mathbf{V}}}(\bar{\mathbf{e}}_{ijk}) \rangle^2 \\ &= \max_j \sum_{i,k} \langle \mathcal{A}, L(\hat{\mathbf{e}}_i *_{\mathbf{L}} \hat{\mathbf{e}}_k *_{\mathbf{L}} \hat{\mathbf{e}}_j^H) *_{\mathbf{L}} \hat{\mathbf{V}} *_{\mathbf{L}} \hat{\mathbf{V}}^H \rangle^2 \\ &= \max_j \sum_{i,k} \frac{1}{\tau^2} \langle \bar{\mathcal{A}}, L(\hat{\mathbf{e}}_i) \odot L(L(\hat{\mathbf{e}}_k)) \odot L(\hat{\mathbf{e}}_j^H) \odot L(\hat{\mathbf{V}}) \odot L(\hat{\mathbf{V}}^H) \rangle^2 \\ &= \max_j \sum_{i,k} \frac{1}{\tau^2} \langle L(L(\hat{\mathbf{e}}_k^H)) \odot L(\hat{\mathbf{e}}_i^H) \odot \bar{\mathcal{A}}, L(\hat{\mathbf{e}}_j^H) \odot L(\hat{\mathbf{V}}) \odot L(\hat{\mathbf{V}}^H) \rangle^2 \\ &\leq \max_j \sum_{i,k} \frac{1}{\tau^2} \|L(L(\hat{\mathbf{e}}_k^H)) \odot L(\hat{\mathbf{e}}_i^H) \odot \bar{\mathcal{A}}\|_F^2 \|L(\hat{\mathbf{e}}_j^H) \odot L(\hat{\mathbf{V}}) \odot L(\hat{\mathbf{V}}^H)\|_F^2 \\ &= \max_j \sum_{i,k} \frac{1}{\tau^2} \|L(L(\hat{\mathbf{e}}_k^H)) \odot \bar{\mathcal{A}}(i, :, :)\|_F^2 \|L(\hat{\mathbf{e}}_j^H) \odot L(\hat{\mathbf{V}})\|_F^2 \\ &= \max_j \sum_{i,k} \frac{1}{\tau} \|L(L(\hat{\mathbf{e}}_k^H)) \odot \bar{\mathcal{A}}(i, :, :)\|_F^2 \|\hat{\mathbf{e}}_j^H *_{\mathbf{L}} \hat{\mathbf{V}}\|_F^2 \\ &\leq \max_j \sum_{i,k} \|\bar{\mathcal{A}}(i, :, :)\|_F^2 \|\hat{\mathbf{e}}_j^H *_{\mathbf{L}} \hat{\mathbf{V}}\|_F^2 \\ &= \max_j n_3 \|\bar{\mathcal{A}}\|_F^2 \|\hat{\mathbf{e}}_j^H *_{\mathbf{L}} \hat{\mathbf{V}}\|_F^2 \\ &= n_3 \tau \|\mathcal{A}\|_F^2 \max_j \|\hat{\mathbf{e}}_j^H *_{\mathbf{L}} \hat{\mathbf{V}}\|_F^2 \\ &\leq \frac{\mu r n_3}{n_2} \|\mathcal{A}\|_F^2. \end{aligned}$$

Then we can bound  $\|\tilde{\mathbf{H}}_j\|$  in the following:

$$\begin{aligned} & \|\tilde{\mathbf{H}}_j\| \\ &= \sup_{\psi} \frac{\delta_j - \rho}{\rho \tau} \sum_{i,k} \langle \text{bdiag}(\overline{\mathcal{P}_{\hat{\mathbf{V}}}(\bar{\mathbf{e}}_{ijk})}), \bar{\mathbf{D}}_1 \rangle \langle \text{bdiag}(\overline{\mathcal{P}_{\hat{\mathbf{V}}}(\bar{\mathbf{e}}_{ijk})}), \bar{\mathbf{D}}_2 \rangle \\ &\leq \sup_{\psi} \frac{\tau}{\rho} |\delta_j - \rho| \sum_{i,k} |\langle \mathbf{D}_1, \mathcal{P}_{\hat{\mathbf{V}}}(\bar{\mathbf{e}}_{ijk}) \rangle| |\langle \mathbf{D}_2, \mathcal{P}_{\hat{\mathbf{V}}}(\bar{\mathbf{e}}_{ijk}) \rangle| \\ &= \sup_{\psi} \frac{\tau}{\rho} |\delta_j - \rho| \sum_{i,k} |\langle \mathcal{P}_{\hat{\mathbf{V}}}(\mathbf{D}_1), \bar{\mathbf{e}}_{ijk} \rangle|^2 \\ &\leq \sup_{\psi} \frac{\tau}{\rho} \sum_{i,k} \langle \mathcal{P}_{\hat{\mathbf{V}}}(\mathbf{D}_1), \bar{\mathbf{e}}_{ijk} \rangle^2 \\ &\leq \sup_{\psi} \frac{\tau}{\rho} \|\mathcal{P}_{\hat{\mathbf{V}}}(\mathbf{D}_1)\|_{2,\infty}^2 \\ &\leq \sup_{\psi} \frac{\tau}{\rho} \frac{\mu r n_3}{n_2} \|\mathbf{D}_1\|_F^2 \\ &\leq \frac{\mu r n_3}{\rho n_2} := \nu. \end{aligned}$$

Note that  $\mathbb{E}[(\rho^{-1}\delta_j - 1)^2] = \rho^{-1}(1 - \rho) \leq \rho^{-1}$ . We further obtain

$$\begin{aligned}
 & \left\| \sum_j \mathbb{E}[\tilde{\mathbf{H}}_j \tilde{\mathbf{H}}_j^H] \right\| \\
 &= \sup_{\psi} \sum_j \mathbb{E}[(\rho^{-1}\delta_j - 1)^2] \tau^2 \left( \sum_{i,k} \langle \mathcal{D}_1, \mathcal{P}_{\hat{\mathbf{v}}}(\bar{\mathbf{e}}_{ijk}) \rangle \langle \mathcal{D}_2, \mathcal{P}_{\hat{\mathbf{v}}}(\bar{\mathbf{e}}_{ijk}) \rangle \right)^2 \\
 &\leq \frac{\tau^2}{\rho} \sup_{\psi} \sum_j \left( \sum_{i,k} |\langle \mathcal{D}_1, \mathcal{P}_{\hat{\mathbf{v}}}(\bar{\mathbf{e}}_{ijk}) \rangle| |\langle \mathcal{D}_2, \mathcal{P}_{\hat{\mathbf{v}}}(\bar{\mathbf{e}}_{ijk}) \rangle| \right)^2 \\
 &= \frac{\tau^2}{\rho} \sup_{\psi} \sum_j \left( \sum_{i,k} \langle \mathcal{D}_1, \mathcal{P}_{\hat{\mathbf{v}}}(\bar{\mathbf{e}}_{ijk}) \rangle^2 \right)^2 \\
 &= \frac{\tau^2}{\rho} \sup_{\psi} \sum_j \|\mathcal{P}_{\hat{\mathbf{v}}}(\mathcal{D}_1)\|_{2,\infty}^2 \left( \sum_{i,k} \langle \mathcal{D}_1, \mathcal{P}_{\hat{\mathbf{v}}}(\bar{\mathbf{e}}_{ijk}) \rangle^2 \right) \\
 &= \frac{\tau^2}{\rho} \frac{\mu r n_3}{\tau n_2} \sup_{\psi} \sum_j \left( \sum_{i,k} \langle \mathcal{D}_1, \mathcal{P}_{\hat{\mathbf{v}}}(\bar{\mathbf{e}}_{ijk}) \rangle^2 \right) \\
 &= \frac{\tau \mu r n_3}{\rho n_2} \sup_{\psi} \sum_j \left( \sum_{i,k} \langle \mathcal{D}_1, \mathcal{P}_{\hat{\mathbf{v}}}(\bar{\mathbf{e}}_{ijk}) \rangle^2 \right) \\
 &\leq \frac{\tau \mu r n_3}{\rho n_2} \sup_{\psi} \|\mathcal{D}_1\|_F^2 \\
 &\leq \frac{\mu r n_3}{\rho n_2} := \omega.
 \end{aligned}$$

Since  $\epsilon$  is small,  $\omega/\nu = 1 > \epsilon$ . By Lemma 7, we have

$$\begin{aligned}
 & \mathbb{P}\left(\|\mathcal{P}_{\hat{\mathbf{v}}} - \frac{1}{\rho} \mathcal{P}_{\hat{\mathbf{v}}} \mathcal{P}_{\hat{\Theta}} \mathcal{P}_{\hat{\mathbf{v}}}\| > \epsilon\right) \\
 &= \mathbb{P}\left(\left\| \sum_j \mathbb{E}[\tilde{\mathbf{H}}_j] \right\| > \epsilon\right) \\
 &\leq (n_1 + n_2) n_3 \exp\left(-\frac{3\epsilon^2}{8\omega}\right) \\
 &\leq (n_1 + n_2) n_3 \exp\left(-\frac{3\epsilon^2 \rho n_2}{8\mu r n_3}\right).
 \end{aligned}$$

Let  $\rho \geq c_2 \mu r n_3 \log(n_{(1)}) / (\epsilon^2 n_2)$ . Then the following inequality holds.

$$\begin{aligned}
 & \mathbb{P}\left(\|\mathcal{P}_{\hat{\mathbf{v}}} - \frac{1}{\rho} \mathcal{P}_{\hat{\mathbf{v}}} \mathcal{P}_{\hat{\Theta}} \mathcal{P}_{\hat{\mathbf{v}}}\| \leq \epsilon\right) \\
 &= 1 - \left(\|\mathcal{P}_{\hat{\mathbf{v}}} - \frac{1}{\rho} \mathcal{P}_{\hat{\mathbf{v}}} \mathcal{P}_{\hat{\Theta}} \mathcal{P}_{\hat{\mathbf{v}}}\| > \epsilon\right) \\
 &\geq 1 - (n_1 + n_2) n_3 \exp\left(-\frac{3\epsilon^2 \rho n_2}{8\mu r n_3}\right) \\
 &\geq 1 - 2n_3 (n_{(1)})^{-\frac{3c_2}{8} + 1}.
 \end{aligned}$$

By choosing an appropriate  $c_2$ , we have  $\mathbb{P}(\|\mathcal{P}_{\hat{\mathbf{v}}} - \frac{1}{\rho} \mathcal{P}_{\hat{\mathbf{v}}} \mathcal{P}_{\hat{\Theta}} \mathcal{P}_{\hat{\mathbf{v}}}\| \leq \epsilon) \geq 1 - n_{(1)}^{-10}$ . The proof is completed.  $\square$

#### F.4.2 Proof of Corollary 1

*Proof.* From Lemma 5 we have

$$\left\| \mathcal{P}_{\hat{\mathbf{v}}} - \frac{1}{1-\rho} \mathcal{P}_{\hat{\mathbf{v}}} \mathcal{P}_{\hat{\Theta}} \mathcal{P}_{\hat{\mathbf{v}}} \right\| \leq \epsilon,$$

provided that  $1 - \rho \geq c_2 \mu r n_3 \log(n_{(1)}) / (\epsilon^2 n_2)$ . Note that  $\mathcal{I} = \mathcal{P}_{\hat{\Theta}} + \mathcal{P}_{\hat{\Theta}^\perp}$ , we have

$$\|\mathcal{P}_{\hat{\mathcal{V}}} - \frac{1}{1-\rho} \mathcal{P}_{\hat{\mathcal{V}}} \mathcal{P}_{\hat{\Theta}^\perp} \mathcal{P}_{\hat{\mathcal{V}}}\| = \frac{1}{1-\rho} \|\rho \mathcal{P}_{\hat{\mathcal{V}}} - \mathcal{P}_{\hat{\mathcal{V}}} \mathcal{P}_{\hat{\Theta}} \mathcal{P}_{\hat{\mathcal{V}}}\|.$$

Then, by the triangular inequality

$$\|\mathcal{P}_{\hat{\mathcal{V}}} \mathcal{P}_{\hat{\Theta}} \mathcal{P}_{\hat{\mathcal{V}}}\| \leq \|\mathcal{P}_{\hat{\mathcal{V}}} \mathcal{P}_{\hat{\Theta}} \mathcal{P}_{\hat{\mathcal{V}}} - \rho \mathcal{P}_{\hat{\mathcal{V}}}\| + \|\rho \mathcal{P}_{\hat{\mathcal{V}}}\| \leq (1 - \rho)\epsilon + \rho.$$

The proof is completed by using  $\|\mathcal{P}_{\hat{\Theta}} \mathcal{P}_{\hat{\mathcal{V}}}\|^2 = \|\mathcal{P}_{\hat{\mathcal{V}}} \mathcal{P}_{\hat{\Theta}} \mathcal{P}_{\hat{\mathcal{V}}}\|$ .  $\square$

## G ADDITIONAL EXPERIMENTS

### G.1 Application to Outlier Detection

Here, we provide results for the missing data experiments on the COIL20-MIRFLICKR-25k dataset in Table 10.

Table 10: AUC of outlier detection and clustering results (ACC, NMI and PUR) on COIL20-MIRFLICKR-25k dataset for missing data experiments.

# Outlier	100												200											
	$\delta = 10\%$				$\delta = 20\%$				$\delta = 10\%$				$\delta = 20\%$											
	AUC	ACC	NMI	PUR																				
TNN+TRPCA-DFT	0.7686	0.5661	0.7055	0.6422	0.7676	0.5621	0.7021	0.6407	0.7393	0.5942	0.7092	0.6499	0.7374	0.5905	0.7087	0.6467								
TNN+TRPCA-DCT	0.7739	0.5354	0.6907	0.6322	0.7726	0.5428	0.6923	0.6343	0.7442	0.5740	0.7028	0.6395	0.7422	0.5732	0.6976	0.6339								
TNN+OR-TPCA-DFT	0.0218	0.5613	0.6566	0.5980	0.0158	0.5708	0.6621	0.6049	0.0070	0.5780	0.6675	0.6089	0.0050	0.5734	0.6646	0.6048								
TNN+TLRR-DFT	0.8170	0.5554	0.6417	0.5997	<b>0.8158</b>	0.5616	0.6505	0.6080	0.8032	0.5600	0.6396	0.5986	<b>0.8009</b>	0.5675	0.6488	0.6069								
TNN+TLRR-DCT	0.7871	0.6268	0.7188	0.6711	0.7876	0.6251	0.7177	0.6695	0.7670	0.6692	0.7415	0.7036	0.7694	0.6680	0.7432	0.7036								
TNN+OR-TLRR-DFT	0.7888	0.6667	0.7499	0.6935	0.7841	0.6653	0.7474	0.6923	0.7653	0.6657	0.7409	0.6868	0.7604	0.6636	0.7408	0.6867								
TNN+OR-TLRR-DCT	0.7657	0.5757	0.6970	0.6376	0.7617	0.5798	0.6980	0.6405	0.7361	0.6038	0.7119	0.6546	0.7351	0.6116	0.7156	0.6608								
TLRR-EWZF-DFT	0.8095	0.5438	0.6255	0.5849	0.7933	0.5160	0.6132	0.5621	0.7923	0.5682	0.6380	0.6034	0.7752	0.5497	0.6182	0.5757								
TLRR-EWZF-DCT	0.7610	0.6446	0.7512	0.6781	0.7269	0.6618	0.7518	0.6913	0.7065	0.6391	0.7482	0.6712	0.6648	0.6742	0.7505	0.6931								
OR-TLRR-EWZF-DFT	0.7919	0.6430	0.7346	0.6719	0.7905	0.6051	0.7153	0.6434	0.7658	0.6685	0.7426	0.6885	0.7630	0.6489	0.7299	0.6753								
OR-TLRR-EWZF-DCT	<b>0.8382</b>	<b>0.6750</b>	<b>0.7586</b>	<b>0.7118</b>	0.7955	<b>0.6763</b>	<b>0.7548</b>	<b>0.6981</b>	<b>0.8225</b>	<b>0.6803</b>	<b>0.7598</b>	<b>0.7042</b>	0.7496	<b>0.6887</b>	<b>0.7603</b>	<b>0.7038</b>								

### G.2 Application to Image Clustering

Table 11: Clustering results (ACC, NMI and PUR) on the noisy Umist dataset.

Methods	$\sigma^2 = 0.1$			$\sigma^2 = 0.2$			$\sigma^2 = 0.3$			$\sigma^2 = 0.4$		
	ACC	NMI	PUR									
R-PCA	0.4252	0.5825	0.4739	0.4230	0.5820	0.4712	0.4185	0.5794	0.4679	0.4215	0.5835	0.4724
OR-PCA	0.3955	0.5762	0.4544	0.3985	0.5776	0.4558	0.3950	0.5703	0.4504	0.3975	0.5740	0.4554
LRR	0.3400	0.4358	0.3952	0.3255	0.4044	0.3748	0.3178	0.3924	0.3643	0.3170	0.3898	0.3634
SSC	0.4848	0.6662	0.5762	0.4810	0.6618	0.5727	0.4846	0.6631	0.5767	0.4762	0.6595	0.5706
SSC-OMP	0.4144	0.5670	0.5093	0.4062	0.5573	0.5006	0.4203	0.5668	0.5105	0.4094	0.5574	0.5061
TRPCA-DFT	0.5701	0.7186	0.6324	0.5683	0.7129	0.6305	0.5581	0.7037	0.6219	0.5624	0.7052	0.6258
TRPCA-DCT	0.5747	0.7218	0.6389	0.5692	0.7167	0.6358	0.5649	0.7104	0.6324	0.5673	0.7090	0.6336
OR-TPCA	0.3240	0.3642	0.3648	0.3170	0.3556	0.3588	0.3170	0.3508	0.3570	0.3195	0.3530	0.3565
TLRR-DFT	0.4437	0.5426	0.5143	0.4216	0.5100	0.4810	0.4198	0.5057	0.4778	0.4177	0.5020	0.4729
TLRR-DCT	0.4956	0.6425	0.5657	0.4834	0.6300	0.5573	0.4712	0.6087	0.5458	0.4720	0.6045	0.5437
OR-TLRR-DFT	<b>0.5992</b>	<b>0.7429</b>	<b>0.6667</b>	<b>0.5955</b>	<b>0.7408</b>	<b>0.6628</b>	<b>0.5820</b>	<b>0.7320</b>	<b>0.6525</b>	<b>0.5902</b>	<b>0.7317</b>	<b>0.6570</b>
OR-TLRR-DCT	0.4930	0.6618	0.5493	0.4907	0.6583	0.5452	0.4909	0.6581	0.5448	0.4984	0.6595	0.5546

In order to study the robustness of our methods for dealing with sample-specific corruptions, we add white Gaussian noise to 20% of the images in the Umist dataset, and we use  $\Theta$  to denote the indices of the selected images. We use  $\mathcal{X}^{\text{noise}}$  to denote the noisy samples. For each sample  $\mathcal{X}^{\text{noise}}(:, j, :)$ ,  $j \in \Theta$ ,  $\mathcal{X}^{\text{noise}}(:, j, :)$  can be expressed as  $\mathcal{X}^{\text{noise}}(:, j, :) = \text{ivc}(\text{vec}(\mathcal{X}(:, j, :)) + \varphi_j)$ , where  $\varphi_j \in \mathbb{R}^{n_1 n_3}$  is additive white Gaussian noise with zero mean and variance  $\sigma^2 \|\mathcal{X}(:, j, :)\|_F^2$ . In this experiment,  $\sigma^2$  ranges from 0.1 to 0.4 and for each  $\sigma^2$ , we generate  $\mathcal{X}^{\text{noise}}$  10 times. We can again observe from Table 11 that OR-TLRR-DFT achieves the highest clustering accuracy, which suggests the effectiveness of the proposed method for clustering the corrupted data.

Table 12: Clustering results (ACC, NMI and PUR) on the noisy USPS dataset.

Methods	SR = 0.3			SR = 0.5			SR = 0.7		
	ACC	NMI	PUR	ACC	NMI	PUR	ACC	NMI	PUR
R-PCA	0.4633	0.4100	0.4912	0.4572	0.4044	0.4876	0.4566	0.3999	0.4816
OR-PCA	0.4489	0.4015	0.4807	0.4482	0.4014	0.4806	0.4468	0.3953	0.4773
LRR	0.2911	0.2146	0.3111	0.2503	0.1603	0.2632	0.2540	0.1375	0.2664
SSC	0.3890	0.4065	0.4618	0.4025	0.4119	0.4670	0.4266	0.4125	0.4690
SSC-OMP	0.1424	0.0984	0.1939	0.1623	0.1149	0.2147	0.2105	0.1517	0.2624
TRPCA-DFT	0.3417	0.2902	0.3767	0.3362	0.2797	0.3744	0.3506	0.2756	0.3739
TRPCA-DCT	0.3399	0.2903	0.3768	0.3380	0.2795	0.3748	0.3564	0.2817	0.3781
OR-TPCA	0.3219	0.2533	0.3468	0.3342	0.2685	0.3573	0.3588	0.2787	0.3754
TLRR-DFT	0.2762	0.1624	0.2892	0.2129	0.0899	0.2233	0.1795	0.0518	0.1870
TLRR-DCT	0.3526	0.2688	0.3965	0.3203	0.2190	0.3557	0.2610	0.1386	0.2802
<b>OR-TLRR-DFT</b>	<b>0.4876</b>	<b>0.4228</b>	<b>0.5251</b>	<b>0.5055</b>	<b>0.4263</b>	<b>0.5348</b>	<b>0.4929</b>	<b>0.4252</b>	<b>0.5317</b>
OR-TLRR-DCT	0.4069	0.3452	0.4460	0.4074	0.3532	0.4555	0.4246	0.3519	0.4638

We finally consider the USPS dataset, which contains 9298 images of  $16 \times 16$  handwritten digits. In this experiment, 100 images are randomly selected from each of the 10 digits; hence  $n_2 = 1000$ . For each image, we randomly shift the digit horizontally with respect to the center by 3 pixels either side (left or right). Then we generate noisy data by adding white Gaussian noise to different percentages of the shifted image samples, where the sampling rate (SR) is set to be 0.3, 0.5 and 0.7. The noise level  $\sigma^2$  is now fixed to be 0.2. One can infer from Table 12 that OR-TLRR-DFT consistently outperforms other compared methods. By introducing column-sparse noise, the robustness of the proposed method is also verified.