A Novel Sparsity Measure for Tensor Recovery

Qian Zhao¹  Deyu Meng¹,²,*  Xu Kong³  Qi Xie¹  Wenfei Cao¹  Yao Wang¹  Zongben Xu¹,²
¹School of Mathematics and Statistics, Xi’an Jiaotong University
²Ministry of Education Key Lab of Intelligent Networks and Network Security, Xi’an Jiaotong University
³School of Mathematical Sciences, Liaocheng University

timmy.zhaqian@gmail.com  dy meng@mail.xjtu.edu.cn  xu.kong@hotmail.com
xq.liwu@stu.xjtu.edu.cn  { caowen2015, yao.s.wang} @gmail.com  zb xu@mail.xjtu.edu.cn

Abstract

In this paper, we propose a new sparsity regularizer for measuring the low-rank structure underneath a tensor. The proposed sparsity measure has a natural physical meaning which is intrinsically the size of the fundamental Kronecker basis to express the tensor. By embedding the sparsity measure into the tensor completion and tensor robust PCA frameworks, we formulate new models to enhance their capability in tensor recovery. Through introducing relaxation forms of the proposed sparsity measure, we also adopt the alternating direction method of multipliers (ADMM) for solving the proposed models. Experiments implemented on synthetic and multispectral image data sets substantiate the effectiveness of the proposed methods.

1. Introduction

Visual data from real applications are often generated by the interaction of multiple factors. For example, a hyperspectral image consists of a collection of images scattered over various discrete bands and thus includes three intrinsic constituent factors, i.e., spectrum and spatial width and height. Beyond the traditional vector or matrix, which is intrinsically the size of the fundamental Kronecker representation; (b) Relationship between the proposed sparsity measure and the existing lower-order sparsity measures.

mistake [1, 31, 37], the tensor data always can only be partially acquired from real data acquisition equipment. Also, some data values could be corrupted by gross noises or outliers [49]. Therefore, recovering tensor from corrupted observations has become a critical and inevitable challenge for tensor data analysis. Such a tensor recovery problem can be formally described in two ways. One is the tensor completion (TC), which focuses on inferring missing entries of the entire tensor from partially observed data; the other is the tensor robust PCA (TRPCA), which corresponds to separating the clean tensor from the corrupting noise. In the degenerated matrix cases, both problems have been thoroughly investigated for decades [7, 23, 34, 35, 42, 6]. However, for general higher-order tensors, until very recent years it begins to attract attention in computer vision and pattern recognition circle [12, 20, 30, 31, 51, 52, 21].

The important clue to handle this issue is to utilize the latent knowledge underlying the tensor structure. The most commonly utilized one is that the tensor intensities along each mode are always with evident correlation. For example, The images obtained across the spectrum of a multispectral image are generally highly correlated. This implies that the tensor along each mode resides on a low-rank subspace and the entire tensor corresponds to the affiliation of the subspaces along all the tensor modes. By fully utilizing such low-rank prior knowledge, the corrupted tensor values
are expected to be properly regularized and faithfully recovered from known ones. This forms the main methodology underlying most of the current tensor recovery methods.

In matrix cases, this low-rank prior isrationally measured by the rank of the training matrix. For practical computation, its relaxations such as the trace norm (also known as nuclear norm) and Schatten-$q$ norm are generally employed. Most current tensor recovery works directly extended this term to higher-order cases by easily ameliorating it as the sum of ranks (or its relaxations) along all tensor modes $\{30, 31, 9\}$. Albeit easy to implement, different from matrix scenarios, this simple rank-sum term is short of a clear physical meaning for general tensors. Besides, using same weights to penalize all dimensionality ranks of a tensor is not always rational. Still taking the multispectral image data as example, the tensor intensity along the spectral dimensionality is always significantly more highly correlated than those along the spatial dimensionality. This prior knowledge delivers the information that the rank of a multispectral image along its spectral dimensionality should be much lower than that along its spatial dimensionality. We thus should more largely penalize the spectral rank rather than spatial ones.

In this paper, our main argument is that instead of using the sum of tensor ranks along all its modes as conventional, we should more rationally use the multiplication of them to encode the low-rank prior inside the tensor data. This simple measurement can well ameliorate the limitations of the currently utilized ones. Specifically, we can mainly conclude its advantages as follows.

Firstly, it has a natural physical interpretation. As shown in Fig. 1 when the rank of a $d$-order tensor along its $n^{th}$ mode is $r_n$, this tensor can be finely represented by at most $\prod_{n=1}^{d} r_n$ Kronecker bases $\{25, 33\}$. This means that the multiplication of tensor ranks along all its dimensions can be well explained as a reasonable proxy for measuring the capacity of tensor space, in which the entire tensor located, by taking Kronecker basis as the fundamental representation component.

Secondly, it provides a possibly unified way to interpret the sparsity measures throughout vector to matrix. The sparsity of a vector is conventionally measured by the number of the bases (from a predefined dictionary) that can represent the vector as a linear combination of them $\{8, 14\}$. Since in vector case, a Kronecker basis is just a common vector, this measurement is just the number of Kronecker bases required to represent the vector, which complies with our proposed sparsity measure. The sparsity of a matrix is conventionally assessed by its rank $\{7, 34, 42\}$. Actually there are the following results $\{15\}$: (1) if the ranks of a matrix along its two dimensions are $r_1$ and $r_2$, respectively, then $r_1 = r_2 = r$; (2) if the matrix is with rank $r$, then it can be represented as $r$ Kronecker bases. The former result means that the proposed measure (by eliminating the square) is proportional to the conventional one, and the latter indicates that this measure also complies with our physical interpretation.

Thirdly, it provides an insightful understanding for rank weighting mechanism in the traditional rank-sum framework. In specific, since the proposed sparsity measure can be equivalently reformulated as:

$$\prod_{n=1}^{d} r_n = \sum_{n=1}^{d} w_n r_n, \quad w_n = \frac{1}{d} \prod_{l=1, l \neq n}^{d} r_l,$$

then it is easy to see that the subspace located in the tensor dimensionality with lower rank will have a relatively larger weight, and vice versa. E.g., for a hyperspectral image, denote $r_s$ the rank of its spectral dimension and $r_w$ and $r_h$ as ranks of its spatial width and height dimensions. Since generally $r_w, r_h \gg r_s$ in this case, our regularization will more largely penalize $r_s$ (with weight $w_n r_n/3$) than $r_w$ and $r_h$. This finely accords with the prior knowledge underlying this tensor kind.

In this paper, by embedding this sparsity measure into the TC and TRPCA frameworks, we formulate new models for both issues to enhance their capability in tensor recovery. Throughout the paper, we denote scalars, vectors, matrices and tensors by the non-bold letters, bold lower case letters, bold upper case letters and calligraphic upper case letters, respectively.

2. Related work

Matrix recovery: There are mainly two kinds of matrix recovery problems, i.e., matrix completion (MC) and robust PCA (RPCA), both of which have been extensively studied. MC problem arises in machine learning scenarios, like collaborative filtering and latent semantic analysis $\{35\}$. In 2009, Candès and Recht $\{7\}$ prompted a new surge for this problem by showing that the matrix can be exactly recovered from an incomplete set of entries through solving a convex semidefinite programming. Similar exact-recovery theory was simultaneously presented by Recht et al. $\{34\}$, under a certain restricted isometry property, for the linear transformation defined constraints. To alleviate the heavy computational cost, various methods have been proposed to solve the trace norm minimization problem induced by the MC model $\{4, 23, 26, 28\}$. Due to the development of these efficient algorithms, MC has been readily applied to computer vision and pattern recognition problems, such as the depth enhancement $\{32\}$ and image tag completion $\{18\}$.

RPCA model was initially formulated by Wright et al. $\{42\}$, with the theoretical guarantee to be able to recover the ground truth tensor from grossly corrupted one under...
certain assumptions \[6\]. Some variants have been further proposed, e.g., Xu et al. \[43\] used the L_{1,2}\text{-norm to handle data corrupted by column. The iterative thresholding method can be used to solve the RPCA model \[6\], but is generally slow. To speed up the computation, Lin et al. proposed the accelerated proximal gradient (APG) \[27\] and the augmented Lagrangian multiplier (ALM) \[29\] methods. ALM leads to state-of-the-art performance in terms of both speed and accuracy. Bayesian approaches to RPCA have also been investigated. Ding et al. \[13\] modeled the singular values of the low-rank matrix and the entries of the sparse matrix with beta-Bernoulli priors, and used a Markov chain Monte Carlo (MCMC) sampling scheme to perform inference. Babacan et al. \[2\] adopted the automatic relevance determination (ARD) approach to RPCA modeling, and utilized the variational Bayes method to do inference.

**Tensor recovery:** In recent years, tensor recovery has been attracting much attention. Generalized from matrix case, tensor recovery can also be categorized into two lines of researches: tensor completion (TC) and tensor robust PCA (TRPCA). Different from the natural sparsity measure (rank) for matrices, it is more complicated to construct a rational sparsity measure to describe the intrinsic correlations along various tensor modes. By measuring sparsity of a tensor with the sum of the ranks of all unfolding matrices along all modes and relaxing with trace norms, Liu et al. \[30\] firstly extended the MC model to TC cases and designed efficient HaLRTC algorithm by applying ADMM to solve it \[31\]. Goldfarb and Qin \[21\] applied the same sparsity measure to TRPCA problem and also solved it by ADMM. Romera-Paredes and Pontil \[36\] promoted this “sum of ranks” measure by relaxing it to a tighter convex form. Recently, Zhang et al. \[51\] proposed a worst case measure, i.e., measuring the sparsity of a tensor by its largest rank of all unfolding matrices, and relaxed it with a sum of exponential forms. Designed mainly for videos, Zhang et al. \[52\] developed a tensor-SVD based sparsity measure for both TC and TRPCA problems. Xu et al. \[44\] investigated factorization based method for TC problem.

It can be seen that most of the currently utilized tensor sparsity measures can be recognized as certain relaxations of the sum of the unfolding matrices ranks along all modes. They on one hand lack a clear physical interpretation, and on the other hand have not a consistent relationship with previous defined sparsity measures for vector/matrix. We thus propose a new measure to alleviate this issue.

### 3. Notions and preliminaries

A tensor corresponds to a multi-dimensional data array. A tensor of order \(d\) is denoted as \(A \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_d}\). Elements of \(A\) are denoted as \(a_{i_1i_2\cdots i_d}\), where \(1 \leq i_n \leq I_n, 1 \leq n \leq d\). For a \(d\)-order tensor \(A\), its \(n\)th unfolding matrix is denoted by \(A_{(n)} = \text{unfold}_n(A) \in \mathbb{R}^{I_n \times (I_1 \cdots I_{n-1}I_{n+1} \cdots I_d)}\), \(1 \leq n \leq d\), whose columns compose of all \(I_n\)-dimensional vectors along the \(n\)th mode of \(A\). Conversely, the unfolding matrix along the \(n\)th mode can be transformed back to the tensor by \(A = \text{fold}_n(A_{(n)})\), \(1 \leq n \leq d\). The \(n\)-rank of \(A\), denoted as \(r_n\), is the dimension of the vector space spanned by all columns of \(A_{(n)}\).

The product of two matrices can be generalized to the product of a tensor and a matrix. The \(n\)-product of a tensor \(A \in \mathbb{R}^{I_1 \times \cdots \times I_n \times \cdots \times I_d}\) by a matrix \(B \in \mathbb{R}^{J_n \times I_n}\), denoted by \(A \times_n B\), is also an \(d\)-order tensor \(C \in \mathbb{R}^{I_1 \times \cdots \times I_N \times \cdots \times I_d}\), whose entries are computed by

\[
C_{i_1 \cdots i_{n-1}i_ni_{n+1} \cdots i_d} = \sum_{j_n} a_{i_1 \cdots i_{n-1}j_ni_{n+1} \cdots i_d} b_{j_ni_n}.
\]

One important decomposition for tensors is Tucker decomposition \[25\] \[38\] (see Fig. 1(a) for visualization), by which any \(d\)-order tensor \(A \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_d}\) can be written as

\[
A = S \times_1 U_1 \times_2 U_2 \times \cdots \times_d U_d,
\]

where \(S \in \mathbb{R}^{r_1 \times r_2 \times \cdots \times r_d}\) is called the core tensor, and \(U_n \in \mathbb{R}^{I_n \times r_n}\) (\(1 \leq n \leq d\)) is composed by the \(r_n\) orthonogonal bases along the \(n\)th mode of \(A\). The Frobenius norm of a tensor \(A = (a_{i_1\cdots i_d}) \in \mathbb{R}^{I_1 \times \cdots \times I_d}\) is defined as \(\|A\|_F = \left(\sum_{i_1, \ldots, i_d=1}^{I_1, \ldots, I_d} a_{i_1\cdots i_d}^2\right)^{1/2}\), and the \(L_1\) norm is defined as \(\|A\|_1 = \sum_{i_1, \ldots, i_d=1}^{I_1, \ldots, I_d} |a_{i_1\cdots i_d}|\).

### 4. New tensor sparsity measure

Our sparsity measure is motivated by the Tucker model of tensors. We know that a \(d\)-order tensor \(X \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_d}\) can be written as \(X = \sum_{i_1, \ldots, i_d=1}^{r_1, \ldots, r_d} s_{i_1\cdots i_d} u_{i_1} \circ u_{i_2} \circ \cdots \circ u_{i_d}\), where \(r_n\) is the \(n\)-rank of \(X\) along its \(n\)th mode, and \(u_{i_n}\) is the \(i_n\)th column of \(U_n\). In other words, tensor \(X\) can be represented by at most \(\prod_{n=1}^{d} r_n\) Kronecker bases. Thus, \(\prod_{d=1}^{d} r_n\) provides a natural way to measure the capacity of the tensor space, by taking Kronecker basis as the fundamental representation component. We denote the new tensor sparsity measure as

\[
S(X) = \prod_{n=1}^{d} \text{rank} (X_{(n)}).
\]

Note that rank \((X_{(n)})\) will lead to combinatorial optimization problems in applications. Therefore, relaxation for \(S(X)\) is needed. In matrix case, the tightest convex relaxation for rank is the trace norm, defined as

\[
\|X\|_* = \sum_{i} \sigma_i (X),
\]

where \(\sigma_i (X)\) is the \(i\)th singular value of the matrix \(X\). Thus, it is natural to relax rank \((X_{(n)})\) to \(\|X_{(n)}\|_*\), and obtain the following relaxation for \(S(X)\):

\[
S_{\text{trace}}(X) = \prod_{n=1}^{d} \|X_{(n)}\|_*.
\]
Nonconvex relaxations for sparsity measures have also been investigated in literatures [16, 43, 11, 48, 17]. Among them, the so-called folded-concave penalties, such as SCAD [16] and MCP [48] have been shown to have nice statistical properties, and have been successfully applied to matrix or tensor recovery problems [40, 29, 9]. Motivated by these, we also relax the proposed tensor sparsity measure using two-folded-concave penalties as follows:

\[ S_{\text{mcp}}(\mathcal{X}) = \prod_{n=1}^{d} P_{\text{mcp}}(X_{(n)}) \quad (4) \]

and

\[ S_{\text{scad}}(\mathcal{X}) = \prod_{n=1}^{d} P_{\text{scad}}(X_{(n)}) \quad (5) \]

where

\[ P_{\text{mcp}}(X) = \sum_{i} \psi_{\text{mcp}}(\sigma_i(X)) \quad \text{and} \quad P_{\text{scad}}(X) = \sum_{i} \psi_{\text{scad}}(\sigma_i(X)), \]

with

\[ \psi_{\text{mcp}}(t) = \begin{cases} \lambda |t| - \frac{a^2}{2a^2}, & \text{if } |t| < a\lambda \\ \frac{a^2}{2}, & \text{if } |t| \geq a\lambda \end{cases} \]

and

\[ \psi_{\text{scad}}(t) = \begin{cases} \lambda |t|, & \text{if } |t| \leq \lambda \\ \frac{a\lambda(t - 0.5 |t|^2 + t^2)}{a^2 - 1}, & \text{if } \lambda < |t| < a\lambda \\ \frac{\lambda^2(a^2 - 1)}{2(a - 1)}, & \text{if } |t| \geq a\lambda \end{cases} \]

respectively. Here, \( \lambda \) and \( a \) are the parameters involved of the folded concave penalties, and are empirically specified as 1 and 6 in our experiments.

4.1. Relation to lower-order sparsity measures

Here we briefly discuss the relationship between the proposed tensor sparsity measure and the existing lower-order sparsity measures, as illustrated in Fig. 1b.

We know that a one-order tensor, i.e., a vector \( x \), can be represented as a linear combination of atoms (one-order Kronecker bases) from a dictionary:

\[ x = D\alpha = \sum_{i} \alpha_i d_i, \quad (6) \]

where \( D = (d_1, \ldots, d_n) \) is the predefined dictionary and \( \alpha = (\alpha_1, \ldots, \alpha_n)^T \) is the coefficients to represent \( x \). In applications, we often seek the \( \alpha \) with least sparsity \( \|\alpha\|_0 \), i.e., with the least non-zero elements, which corresponds to the least Kronecker bases used to represent \( x \).

A two-order tensor, i.e., a matrix, can be represented as the following singular value decomposition (SVD) form:

\[ X = U\Sigma V^T = \sum_{i} \sigma_i u_i v_i^T, \quad (7) \]

where \( U \) and \( V \) are orthogonal matrices, and \( \Sigma = \text{diag}(\sigma) \) with \( \sigma = (\sigma_1, \ldots, \sigma_n)^T \) being the singular values of \( X \). The sparsity measure of \( X \) is known as its rank, i.e., the number of non-zero elements of \( \sigma \). This sparsity intrinsically corresponds to the number of two-order Kronecker bases \( u_i v_i^T \) used to represent \( X \), and thus also complies with our definition for general tensors.

From the above discussion, we can see that our tensor sparsity measure provides a possibly unified way to interpret the sparsity measures throughout vector to matrix.

It should be noted that the atomic norm [10] provides a similar unification in a more mathematically rigorous way. However, it corresponds to the conventional rank for tensors, which is generally intractable in practice. In contrast, the proposed sparsity measure is based on the Tucker model, and thus can be easily computed.

5. Application to tensor recovery

5.1. Tensor completion

Tensor completion (TC) aims to recover the tensor corrupted by missing values from incomplete observations. Using the proposed sparsity measure, we can mathematically formulate the TC model as

\[ \min_{\mathcal{X}} S(\mathcal{X}), \quad s.t. \quad \mathcal{X}_\Omega = \mathcal{T}_\Omega, \quad (8) \]

where \( \mathcal{X}, \mathcal{T} \in \mathbb{R}^{l_1 \times l_2 \times \cdots \times l_d} \) are the reconstructed and observed tensors, respectively, and the elements of \( \mathcal{T} \) indexed by \( \Omega \) are given while the remaining are missing; \( S(\cdot) \) is the tensor sparsity measure as defined in (4). We refer to the model as rank product tensor completion or RP-TC.

In practice, we do not directly adopt model (8), since \( S(\mathcal{X}) \) makes the optimization difficult to solve. Instead, applying relaxations (5) to \( S(\mathcal{X}) \), we can get the following three TC models:

\[ \min_{\mathcal{X}} \frac{1}{c_1} S_{\text{trace}}(\mathcal{X}), \quad s.t. \quad \mathcal{X}_\Omega = \mathcal{T}_\Omega, \quad (9) \]

\[ \min_{\mathcal{X}} \frac{1}{c_2} S_{\text{mcp}}(\mathcal{X}), \quad s.t. \quad \mathcal{X}_\Omega = \mathcal{T}_\Omega, \quad (10) \]

\[ \min_{\mathcal{X}} \frac{1}{c_3} S_{\text{scad}}(\mathcal{X}), \quad s.t. \quad \mathcal{X}_\Omega = \mathcal{T}_\Omega, \quad (11) \]

where \( c_i (i = 1, 2, 3) \) are taken for computational convenience (we will discuss this later). We denote the three models as RP-TC_{\text{trace}}, RP-TC_{\text{mcp}} and RP-TC_{\text{scad}}, respectively.

The above TC models are not convex, and therefore more difficult to solve. Fortunately, we can apply the alternating direction method of multipliers (ADMM) [3, 28] to them. By virtue of ADMM, the original models can be divided into simpler subproblems with closed-form solutions, and thus be effectively solved. To do this, taking RP-TC_{\text{trace}} model (9) as an example, we need to first introduce \( d \) auxiliary tensors \( \mathcal{M}_n \) \( (1 \leq n \leq d) \) and equivalently reformulate the problem as follows:

\[ \min_{\mathcal{X}, \mathcal{M}_1, \ldots, \mathcal{M}_d} \frac{1}{c_1} \prod_{n=1}^{d} \| \mathcal{M}_{n(\Omega)} \|_*, \quad \text{s.t.}, \quad \mathcal{X}_\Omega = \mathcal{M}_\Omega, \quad \mathcal{X} = \mathcal{M}_n, \quad 1 \leq n \leq d, \quad (12) \]

where \( \mathcal{M}_{n(\Omega)} = \text{unfolds}(\mathcal{M}_n) \). Then we consider the augmented Lagrangian function for (12) as:

\[ \begin{aligned}
L_{\mu}(\mathcal{X}, \mathcal{M}_1, \ldots, \mathcal{M}_d, \mathcal{Y}_1, \ldots, \mathcal{Y}_d) &= \frac{1}{c_1} \prod_{n} \| \mathcal{M}_{n(\Omega)} \|_* + \sum_{n} \langle \mathcal{X} - \mathcal{M}_n, \mathcal{Y}_n \rangle + \frac{\mu}{2} \sum_{n} \| \mathcal{X} - \mathcal{M}_n \|_F^2, \\
\end{aligned} \]
where \( Y_n \) (1 ≤ \( n \) ≤ \( d \)) are the Lagrange multipliers and \( \rho \) is a positive scalar. Now we can solve the problem within the ADMM framework. With \( M_j \) (\( j \neq i \)) fixed, \( M_n \) can be updated by solving the following problem:

\[
\min_{M_n} L_{\rho}(\mathcal{X}, M_1, \cdots, M_d, Y_1, \cdots, Y_d),
\]

which has the closed-form solution:

\[
M_n = \text{fold}_n \left( D_{\rho} \left( \text{unfold}_n (\mathcal{X} + \frac{1}{\rho} Y_n) \right) \right),
\]

where \( D_{\rho}(X) = U \Sigma_{\rho} V^T \) is a shrinkage operator with \( \Sigma_{\rho} = \text{diag}(\max(\sigma_i(X) - \tau, 0)) \), and

\[
\alpha_n = \frac{1}{C_1} \prod_{i=1, l \neq n}^d \|M_{(i)}\|_1, \quad 1 \leq n \leq d.
\]

Similarly, with \( M_n \) fixed, \( \mathcal{X} \) can be updated by solving

\[
\min_{\mathcal{X}} L_{\rho}(\mathcal{X}, M_1, \cdots, M_d, Y_1, \cdots, Y_d),
\]

which also has the closed-form solution:

\[
\mathcal{X} = \frac{1}{\rho d} \sum_{n=1}^d (\rho M_n - Y_n).
\]

Then the Lagrangian multipliers are updated by

\[
Y_n := Y_n - \rho (M_n - \mathcal{X}), \quad 1 \leq n \leq d,
\]

and \( \rho \) is increased to \( \rho \rho \) with some constant \( \mu > 1 \). Note that the above procedure and its derivation are similar to HaLRTC in [31]. The main difference is that the \( \alpha_i \) in our algorithm can be varied during the iterations instead of fixed, and thus adaptively give proper penalizations along each mode. This, from the algorithm point of view, explains the advantage of our method beyond previous work.

For models (10) and (11), the general ADMM framework can also be applied. However, due to the more complex folded-concave penalties, we need to slightly modify the algorithm. We leave details in supplementary material. Since models (9)–(11) are not convex, the algorithm cannot be guaranteed to achieve global optima, but empirically perform well in all our experiments.

### 5.2. Tensor robust PCA

Tensor robust PCA (TRPCA) aims to recover the tensor from grossly corrupted observations. The observed tensor can then be written in the following form:

\[
\mathcal{T} = \mathcal{L} + \mathcal{E},
\]

where \( \mathcal{L} \) is the main tensor with higher-order sparsity, and \( \mathcal{E} \) corresponds the sparse noise/outliers embedded in data. Using the proposed tensor sparsity measure, we can get the following TRPCA model:

\[
\min_{\mathcal{L}, \mathcal{E}} S(\mathcal{L}) + \lambda \|\mathcal{E}\|_1, \quad \text{s.t.} \quad \mathcal{T} = \mathcal{L} + \mathcal{E},
\]

where \( \lambda \) is a tuning parameter compromising the recovery tensor and noise terms.

However, for real data such as hyperspectral image [49], the embedded noise is always not solely sparse, but probably a mixture of sparse and Gaussian. We thus ameliorate the observation expression as:

\[
\mathcal{T} = \mathcal{L} + \mathcal{E} + \mathcal{N},
\]

where \( \mathcal{N} \) denotes the Gaussian noise. We then propose the following TRPCA model, called rank product TRPCA (RP-TRPCA):

\[
\min_{\mathcal{L}, \mathcal{E}} S(\mathcal{L}) + \lambda \|\mathcal{E}\|_1 + \gamma \|\mathcal{T} - (\mathcal{L} + \mathcal{E})\|_F^2,
\]

where \( \gamma \) is the tuning parameter related to the intensity of Gaussian noise. Replacing \( S(\mathcal{L}) \) with the three relaxations proposed in Section 4, we obtain the following TRPCA models:

\[
\begin{align*}
\min_{\mathcal{L}, \mathcal{E}} & \quad \frac{1}{C_1} \prod_{i=1}^d \|M_{(i)}\|_1 + \lambda \|\mathcal{E}\|_1 + \gamma \|\mathcal{T} - (\mathcal{L} + \mathcal{E})\|_F^2, \\
\min_{\mathcal{L}, \mathcal{E}} & \quad \frac{1}{C_2} S_{\text{mcp}}(\mathcal{L}) + \lambda \|\mathcal{E}\|_1 + \gamma \|\mathcal{T} - (\mathcal{L} + \mathcal{E})\|_F^2, \\
\min_{\mathcal{L}, \mathcal{E}} & \quad \frac{1}{C_3} S_{\text{scad}}(\mathcal{L}) + \lambda \|\mathcal{E}\|_1 + \gamma \|\mathcal{T} - (\mathcal{L} + \mathcal{E})\|_F^2,
\end{align*}
\]

which are referred as RP-TRPCA_{trace}, RP-TRPCA_{mcp} and RP-TRPCA_{scad}, respectively.

Similar to the TC case, we also apply the ADMM to solving these models. Taking RP-TRPCA_{trace} model as an example, by introducing auxiliary tensors \( M_n \) (1 ≤ \( n \) ≤ \( d \)), we can equivalently reformulate it as:

\[
\begin{align*}
\min_{\mathcal{L}, \mathcal{E}, M_1, \cdots, M_d} & \quad \frac{1}{C_1} \prod_{n=1}^d \|M_{(n)}\|_1 + \lambda \|\mathcal{E}\|_1 + \gamma \|\mathcal{T} - (\mathcal{L} + \mathcal{E})\|_F^2, \\
\text{s.t.} & \quad \mathcal{L} = M_n, \quad 1 \leq n \leq d.
\end{align*}
\]

Then the augmented Lagrangian function for (26) becomes

\[
\begin{align*}
L_{\rho}(\mathcal{L}, \mathcal{E}, \mathcal{M}_1, \cdots, \mathcal{M}_d, Y_1, \cdots, Y_d) &= \frac{1}{C_1} \prod_{n=1}^d \|M_{(n)}\|_1 + \lambda \|\mathcal{E}\|_1 + \gamma \|\mathcal{T} - (\mathcal{L} + \mathcal{E})\|_F^2 \\
&\quad + \sum_{n=1}^d \beta \|\mathcal{X} - \mathcal{M}_n, Y_n\|_2^2 + \frac{\rho}{2} \sum_{n=1}^d \|\mathcal{X} - \mathcal{M}_n\|_2^2.
\end{align*}
\]

Within the ADMM framework, the original problem can be solved via alternatingly solving a sequence of subproblems with respect to \( L_{\rho} \). Besides, all the subproblems involved can be efficiently solved similar to solving RP-TC models. We omit the details due to space limitations.

### 5.3. Discussion for setting \( C_i \)

Now we briefly discuss how to determine the parameter \( C_i \) involved in the proposed TC and TRPCA models. For the TC models, since there is only one term in the objective function, \( C_i \) mainly guarantees the computation stability. Therefore, we can fix it as a relative large constant. In the TRPCA models, the situation is more complicated, since there are three terms in the objective function, and \( C_i \), together with \( \lambda \) and \( \gamma \), balances these terms. We empirically found that the algorithm performs consistently well under the setting \( C_i = \frac{1}{d} \sum_{n=1}^d \prod_{i \neq n} P(L(i)) \), \( \lambda \) to be with order
of $1/\sqrt{\max\{L_{i}^n\}}$ and $\gamma = 0.1/\sigma$, where $L_{(i)}$ is the $i$th unfolding matrix of the ground truth tensor $L$, $P(\cdot)$ corresponds to the matrix sparsity measure and $\sigma$ is the standard deviation of the Gaussian noise. Since the ground truth tensor $L$ is unknown, we first run another easy-to-be-implemented TRPCA method, such as that proposed in [21], to initialize $C_i$ based on the obtained result, and then implement our algorithms under such initialization. This strategy performs well throughout our experiments.

6. Experiments

In this section, we evaluate the effectiveness of the proposed sparsity measure for tensor recovery problems, including TC and TRPCA, on both synthetic and real data. All experiments were implemented in Matlab 8.4(R2014b) on a PC with 3.50GHz CPU and 32GB RAM.

6.1. Tensor completion experiments

Synthetic tensor completion. The synthetic data were generated as follows: first, the ground truth tensor was yielded from the Tucker model, i.e., $T = S \times_1 U_1 \times_2 U_2 \times_3 U_3$, where the core tensor $S \in \mathbb{R}^{r_1 \times r_2 \times r_3}$ was randomly generated from the standard Gaussian distribution, and all $U_n \in \mathbb{R}^{I_n \times r_n}$ were randomly generated column-orthogonal matrices; then a portion of elements was randomly sampled as observed data while the rest were left as missing components. We set $I_n (n = 1, \ldots, 3)$ to 50, respectively, resulting the ground truth tensor with size $50 \times 50 \times 50$. For the rank parameters $r_n$ along each mode, we considered two settings, $(10, 20, 40)$ and $(20, 40, 40)$. These settings are designed to simulate the ranks along different modes with diversity, which is always encountered in practice, e.g., for multispectral image, the rank along the spectral mode is always much lower than those along the spatial mode. We then varied the percentage of sampled elements from 20% to 80% and reconstructed the tensor using the proposed methods and the state-of-the-arts along this line, including ALM based matrix completion (denoted as MC-ALM) [26], HaLRTC [31], McpLRTC [9], ScadLRTC [9], tensor-SVD based TC method (t-SVD) [52] and factorization based TC method (TMac) [44]. The performance in terms of relative reconstruction error (RRE) averaged over 20 realizations, was summarized in Table 1. For matrix-based method MC-ALM, we applied it to the unfolded matrix along each mode of the tensor, obtaining 3 RREs, and report the best one. The average computation time are also listed. The best and the second best results in terms of RRE are highlighted in bold with and without underline, respectively. It can be seen from Table 1 that, compared with other competing methods, the proposed RP-TC methods can accurately recover the tensor with very few observations (20% or 30%). Specifically, under the rank setting $(10, 20, 40)$, the proposed methods, together with three other methods can recover the tensor with 20% observations, and the proposed methods perform slightly better than the methods in [9]. Under the more challenging rank setting $(20, 40, 40)$, all methods failed with 20% observations, while only the proposed methods can accurately recover the ground truth tensor with 30% elements observed. It is also observed that RP-TCtrace does not perform very well in this tensor completion task. This is due to the fact that, as a relaxation for the rank of the unfolding matrix, trace norm is not so tight, and thus makes the relaxation for the proposed sparsity measure even looser. As for computational cost, we can see that the proposed methods with folded-concave relaxations have similar amount of running time as the methods in [9], which also used folded-concave penalties. Considering their better performance, the cost is acceptable.

Multispectral image completion. Then we test the performance of the proposed methods using multispectral image data. Two well-known data sets, Natural Scenes 2002 [19] and Columbia Multispectral Image Database [46] were considered. The former contains multispectral images captured from 8 scenes, with 31 bands (varying wavelength from 410nm to 710nm in 10nm steps) and spatial resolution $820 \times 820$ natural scene; the latter contains 32 real-world scenes, each with spatial resolution $512 \times 512$ and 31 bands (varying from 400nm to 700nm in 10nm steps).

We used 2 images from Natural Scenes 2002 (Scene 4

Table 1. Performance comparison of 9 competing TC methods on synthetic data with rank setting $(10, 20, 40)$ (upper) and $(20, 40, 40)$ (lower). The best and the second best results in terms of RRE are highlighted in bold with and without underline, respectively.

<table>
<thead>
<tr>
<th>Method</th>
<th>20%</th>
<th>30%</th>
<th>40%</th>
<th>Avg. Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC-ALM</td>
<td>0.567</td>
<td>0.306</td>
<td>5.54e-2</td>
<td>1.76e-8</td>
</tr>
<tr>
<td>HaLRTC</td>
<td>0.835</td>
<td>0.704</td>
<td>0.543</td>
<td>2.46e-8</td>
</tr>
<tr>
<td>t-SVD</td>
<td>0.815</td>
<td>0.628</td>
<td>0.427</td>
<td>1.48e-4</td>
</tr>
<tr>
<td>TMac</td>
<td>0.405e-3</td>
<td>1.44e-3</td>
<td>6.95e-4</td>
<td>1.86e-4</td>
</tr>
<tr>
<td>McpLRTC [9]</td>
<td>2.93e-5</td>
<td>2.26e-8</td>
<td>7.85e-12</td>
<td>5.23e-16</td>
</tr>
<tr>
<td>ScadLRTC [9]</td>
<td>2.94e-5</td>
<td>2.26e-8</td>
<td>7.85e-12</td>
<td>5.23e-16</td>
</tr>
<tr>
<td>RP-TC_trace</td>
<td>0.833</td>
<td>0.697</td>
<td>0.517</td>
<td>1.52e-8</td>
</tr>
<tr>
<td>RP-TC_mcp</td>
<td>0.930e-6</td>
<td>4.70e-9</td>
<td>1.80e-12</td>
<td>5.20e-16</td>
</tr>
<tr>
<td>RP-TC_scad</td>
<td>9.28e-6</td>
<td>4.26e-9</td>
<td>9.76e-13</td>
<td>5.18e-16</td>
</tr>
<tr>
<td>Method</td>
<td>20%</td>
<td>30%</td>
<td>40%</td>
<td>Avg. Time (s)</td>
</tr>
<tr>
<td>--------</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>--------------</td>
</tr>
<tr>
<td>MC-ALM</td>
<td>0.783</td>
<td>0.646</td>
<td>0.503</td>
<td>1.36e-7</td>
</tr>
<tr>
<td>HaLRTC</td>
<td>0.875</td>
<td>0.794</td>
<td>0.703</td>
<td>0.160</td>
</tr>
<tr>
<td>t-SVD</td>
<td>0.937</td>
<td>0.838</td>
<td>0.726</td>
<td>9.72e-2</td>
</tr>
<tr>
<td>TMac</td>
<td>0.943</td>
<td>0.192</td>
<td>5.55e-3</td>
<td>3.31e-4</td>
</tr>
<tr>
<td>McpLRTC [9]</td>
<td>0.973</td>
<td>0.498</td>
<td>1.85e-4</td>
<td>2.57e-14</td>
</tr>
<tr>
<td>ScadLRTC [9]</td>
<td>0.973</td>
<td>0.498</td>
<td>1.85e-4</td>
<td>2.57e-14</td>
</tr>
<tr>
<td>RP-TC_trace</td>
<td>0.875</td>
<td>0.793</td>
<td>0.701</td>
<td>0.119</td>
</tr>
<tr>
<td>RP-TC_mcp</td>
<td>0.836</td>
<td>7.05e-4</td>
<td>6.66e-5</td>
<td>1.70e-14</td>
</tr>
<tr>
<td>RP-TC_scad</td>
<td>0.833</td>
<td>5.23e-4</td>
<td>6.85e-5</td>
<td>1.74e-14</td>
</tr>
</tbody>
</table>

2We have also tried the setting that all the modes are with the same rank as mostly investigated in previous work [20, 30, 31], while most of the methods, including ours, perform similarly well. Therefore, we do not include the results to save space.

3Defined as $RRE = \|X - T\|_F/\|T\|_F$, where $T$ and $X$ denote the ground truth and reconstructed tensors, respectively.

http://personalpages.manchester.ac.uk/staff/d.h.foster/

http://www1.cs.columbia.edu/CAVE/databases/multispectral/
and Scene 8) and 6 images from Columbia Multispectral Image Database (balloons, beads, cloth, jellybeans, peppers and watercolors) in our experiments, and each image was downsampled with half length and width of the original one. Then we varied the sampling rate from 10% to 50%, and applied TC methods to recovering the images. Similar to the performance evaluation utilized by Peng et al. [33], we employ the following five quantitative picture quality indices (PQI) to assess the performance of a TC method: peak signal-to-noise ratio (PSNR), structure similarity (SSIM) [41], feature similarity (FSIM) [50], erreur relative globale adimensionnelle de synthèse (ERGAS) [39] and spectral angle mapper (SAM) [47]. Good completion results correspond to larger values in PSNR, SSIM and FSIM, and smaller values in ERGAS and SAM.

The averaged performances of each method over the 8 images under different sampling rates are summarized in Table 2 (detailed results on each image are presented in supplementary material). Each image is resized to $256 \times 256$ for all spectral bands, and rescale to $[0, 1]$. Gaussian noise with standard deviation $\sigma = 0.1$ was added, and 10% of the pixels were further randomly chosen and added to salt-and-pepper noise. Mixing the two kinds of noises aims to simulate real noise that can be found in remote hyperspectral images [49]. We implemented the proposed TRPCA models. The idea is as follows: for a 3D tensor patch, we aggregate them to restore the whole multispectral image. We used 10 images from Columbia Multispectral Image Database, including balloons, beads, chart and stuffed toy, cloth, egyptian statue, feathers, flowers, photo and face, pompoms and watercolors, for testing (more results are presented in supplementary material). Each image is resized to $256 \times 256$ for all spectral bands, and rescale to $[0, 1]$. Gaussian noise with standard deviation $\sigma = 0.1$ was added, and 10% of the pixels were further randomly chosen and added to salt-and-pepper noise. Mixing the two kinds of noises aims to simulate real noise that can be found in remote hyperspectral images [49]. We implemented the proposed TRPCA models. The idea is as follows: for a 3D tensor patch, we aggregate them to restore the whole multispectral image.

### 6.2. Tensor robust PCA

#### Simulated multispectral image restoration.

We first test the proposed TRPCA models by restoring multispectral images corrupted by artificially added noise. The idea is motivated by the application of low-rank matrix based method to image denoising [22]. In [22], Gu et al. proposed an image denoising method by exploiting the low-rank property of the matrix formed by nonlocal similar patches. Compared with gray image studied there, multispectral image has an additional spectral dimension. Besides, the different spectral bands are generally highly correlated. Therefore, we can generalize the matrix based denoising method to tensor case using the proposed TRPCA models. The idea is as follows: for a 3D tensor patch, we search for its nonlocal similar patches across the multispectral image, and reshape these patches to matrices; these matrices are then stacked into a new 3D tensor, which is the input of our TRPCA methods for denoising; we further extract the cleaned patches from the output of TRPCA and aggregate them to restore the whole multispectral image.

We used 10 images from Columbia Multispectral Image Database, including balloons, beads, chart and stuffed toy, cloth, egyptian statue, feathers, flowers, photo and face, pompoms and watercolors, for testing (more results are presented in supplementary material). Each image is resized to $256 \times 256$ for all spectral bands, and rescale to $[0, 1]$. Gaussian noise with standard deviation $\sigma = 0.1$ was added, and 10% of the pixels were further randomly chosen and added to salt-and-pepper noise. Mixing the two kinds of noises aims to simulate real noise that can be found in remote hyperspectral images [49]. We implemented the proposed TRPCA models. The idea is as follows: for a 3D tensor patch, we aggregate them to restore the whole multispectral image.
Table 3. The average performance comparison of 7 competing methods on 10 noisy multispectral images.

<table>
<thead>
<tr>
<th>Method</th>
<th>PSNR</th>
<th>SSIM</th>
<th>FSIM</th>
<th>ERGAS</th>
<th>SAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noisy image</td>
<td>13.01</td>
<td>0.116</td>
<td>0.387</td>
<td>1084.9</td>
<td>0.946</td>
</tr>
<tr>
<td>TDL [33]</td>
<td>26.03</td>
<td>0.631</td>
<td>0.887</td>
<td>241.4</td>
<td>0.287</td>
</tr>
<tr>
<td>RPCA [26]</td>
<td>28.66</td>
<td>0.670</td>
<td>0.836</td>
<td>170.7</td>
<td>0.491</td>
</tr>
<tr>
<td>HoRPCA [21]</td>
<td>28.77</td>
<td>0.829</td>
<td>0.880</td>
<td>174.0</td>
<td>0.247</td>
</tr>
<tr>
<td>t-SVD [52]</td>
<td>29.81</td>
<td>0.713</td>
<td>0.868</td>
<td>152.9</td>
<td>0.370</td>
</tr>
<tr>
<td>RP-TRPCA ___ _trace</td>
<td>28.86</td>
<td>0.832</td>
<td>0.881</td>
<td>172.5</td>
<td>0.251</td>
</tr>
<tr>
<td>RP-TRPCA ___ _mcp</td>
<td>32.70</td>
<td>0.832</td>
<td>0.922</td>
<td>112.9</td>
<td>0.237</td>
</tr>
<tr>
<td>RP-TRPCA ___ _scad</td>
<td>29.27</td>
<td>0.818</td>
<td>0.914</td>
<td>116.3</td>
<td>0.242</td>
</tr>
</tbody>
</table>

Figure 3. (a) The original image from beads; (b) The corresponding noisy image; (c)-(i) The recovered images by TDL [33], RPCA [26], HoRPCA [21], t-SVD [52], RP-TRPCA \_\_\_ \_trace, RP-TRPCA \_\_\_ \_mcp and RP-TRPCA \_\_\_ \_scad, respectively.

Figure 4. (a) The original images from HYDICE urban; (b)-(h) The restored images by MC-ALM [26], HaLR TC [31], t-SVD [52], TMac [44], RP-TC \_\_\_ \_trace, RP-TC \_\_\_ \_mcp and RP-TC \_\_\_ \_scad, respectively.

Real hyperspectral image restoration. Now we apply the proposed TRPCA models to real hyperspectral image restoration. We used a HYDICE urban image [6] for demonstration. The original image is of size $307 \times 307 \times 210$, while contains several bands seriously polluted by the atmosphere and water absorption. After removing these seriously polluted bands, we obtain the image of size $307 \times 307 \times 188$. Using the similar method described before, we can apply the competing methods to restoring this image. For TRPCA based methods, patches of size $8 \times 8$ with step of 4 pixels were extracted, and the group size was set to 20. The restoration results on sample bands are shown in Fig. 3.

7. Conclusion

In this paper, we present a new tensor sparsity measure, together with its three relaxations. Then we apply it to TC and TRPCA models, and design the ADMM algorithms to solve them. The experiments on synthetic and multispectral image data substantiate the rationality of the proposed sparsity measure and the effectiveness of our models on tensor recovery tasks.

Acknowledgments. This research was supported by 973 Program of China with No. 3202013CB329404, and the NSFC projects with No. 61373114, 11401286, 91330204, 11131006, 61503263.
References


