Applied Mathematics and Computation 483 (2024) 128980

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

Full Length Article

Tensor robust principal component analysis with total generalized variation for high-dimensional data recovery

Zhi Xu [a,b], Jing-Hua Yang [c,*], Chuan-long Wang [a,b], Fusheng Wang [a], Xi-hong Yan [a,b]

[a] School of Mathematics and Statistics, Taiyuan Normal University, Jinzhong, Shanxi, 030619, PR China
[b] Shanxi Key Laboratory for Intelligent Optimization Computing and Blockchain Technology, Taiyuan Normal University, Jinzhong, Shanxi, 030619, PR China
[c] School of Information Science and Technology, Southwest Jiaotong University, Chengdou, Sichuan, 611756, PR China

ARTICLE INFO

Keywords:
Tensor robust principal component analysis
Total generalized variation
Tensor singular value decomposition
Tensor nuclear norm
The alternating direction method of multiplier algorithm

ABSTRACT

In the past few years, tensor robust principal component analysis (TRPCA) which is based on tensor singular value decomposition (t-SVD) has got a lot of attention in recovering low-rank tensor corrupted by sparse noise. However, most TRPCA methods only consider the global structure of the image, ignoring the local details and sharp edge information of the image, resulting in the unsatisfactory restoration results. In this paper, to fully preserve the local details and edge information of the image, we propose a new TRPCA method by introducing a total generalized variation (TGV) regularization. The proposed method can simultaneously explore the global and local prior information of high-dimensional data. Specifically, the tensor nuclear norm (TNN) is employed to develop the global structure feature. Moreover, we introduce the TGV, a higher-order generalization of total variation (TV), to preserve the local details and edges of the underlying image. Subsequently, the alternating direction method of multiplier (ADMM) algorithm is introduced to solve the proposed model. Sufficient experiments on color images and videos have demonstrated that our method is superior to other comparison methods.

1. Introduction

The tensor, which is a higher-order form of vector and matrix, is available for various realms, e.g., communication systems [1,2], color images and videos processing [3–6], the hyperspectral data recovery [7–9], and data clustering [10]. Data is always unavoidably contaminated by noise in the retrieving and transmitting courses. Therefore, image denoising has become an essential step to get a clean image. Principal component analysis (PCA) [11], as a powerful tool for extracting the underlying low-dimensional structures, has received more and more attention. It can efficiently reconstruct data interfered by low noise. Whereas, one major weakness to PCA is that its performance degrades rapidly with respect to data which is grossly corrupted or outlying. Therefore, the research on making PCA robust has become an active topic in the last decades.

* Corresponding author.
E-mail addresses: xuzhizhenyuan@163.com (Z. Xu), yangjinghua110@126.com (J.-H. Yang), clwang1964@163.com (C.-l. Wang), fswang2005@163.com (F. Wang), xihong1@e.ntu.edu.sg (X.-h. Yan).

https://doi.org/10.1016/j.amc.2024.128980
Received 17 September 2023; Received in revised form 18 July 2024; Accepted 25 July 2024

Available online 5 August 2024

Motivated by theories about low-rank matrix analysis, Candès et al. [12] proposed the robust principal component analysis (RPCA) to decompose an acquired two-dimensional data $\mathbf{X} \in \mathbb{R}^{n_1 \times n_2}$ into a low-rank component $\mathbf{L}$ and sparse component $\mathbf{S}$. The RPCA is the mathematical solution for the following low-rank matrix restoration problem:

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

where $\|\mathbf{L}\|_* = \sum_i \sigma_i$ denotes the nuclear norm of $\mathbf{L}$, $\sigma_i (i = 1, 2, \cdots, \min(n_1, n_2))$ is the i-th singular value of $\mathbf{L}$, $\|\mathbf{S}\|_1 = \sum_{i,j} |s_{i,j}|$ represents the l_1 norm of sparse matrix $\mathbf{S}$, $s_{i,j}$ is the (i, j)-th entry of $\mathbf{S}$ and $\lambda > 0$ denotes a trade-off parameter. RPCA-based methods have been successfully employed in numerous machine learning and computer vision problems, such as image recovery [13,14], surveillance video processing [15], and image alignment [16]. Nevertheless, the main weakness to RPCA is its limited feasibility for processing two-dimensional data. As a matter of fact, most real-world data, like color images, videos and multispectral remote sensing images, are generally high-dimensional. For application's sake of RPCA, one must first permute the high-dimensional data to form a matrix. This matricization pretreatment can generally lead to information loss or the destruction of the inherent structures of data. So studying a variant method of RPCA, which is appropriate for high-dimensional data, is significant.

To effectively handle the recovery of high-dimensional data, tensor RPCA (TRPCA) methods have been exploited [17,18]. The tensor case TRPCA can be seen as the high-order form of the matrix case RPCA. In tensor case, TRPCA is proposed to recover a clean high-dimension data corrupted by sparse noise. Compared with RPCA, TRPCA can vividly express the essential spatial and spectral information inside multi-dimensional data. Given a tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_m}$, TRPCA can be summarized as the following low-rank tensor recovery problem:

$$\min_{\mathcal{L},\mathcal{S}} \operatorname{rank}(\mathcal{L}) + \lambda \|\mathcal{S}\|_1, \quad \text{s.t.} \quad \mathcal{X} = \mathcal{L} + \mathcal{S}. \tag{2}$$

Unlike matrix rank, tensor rank has several definitions and measurements in TRPCA methods. Different definitions for tensor rank can be induced by different kinds of tensor decompositions [19]. The CANDECOMP/PARAFAC (CP) rank [20,21], defined as the minimum number of the rank-1 tensors generated from CP decomposition of a tensor, is NP-hard [22]. With no transparent algorithms to determine CP rank, the TRPCA methods based on CP decomposition are difficult to solve [23]. Tucker rank, which is produced by Tucker decomposition, is a vector made up of ranks of the unfolding matrices along each mode of the tensor. The sum of nuclear norm (SNN) was proposed to be a convex substitute for Tucker rank [24]. A SNN-based TRPCA was proposed in [25] and has presented the precise recover guarantee. However, the operations of unfolding the tensor along all modes may ignore the internal relationship between adjacent elements of high-dimensional data. To preserve the low-rankness of high-dimensional data without destroying the structure, tensor singular value decomposition(t-SVD)-based tensor tubal rank [26,27] is proposed. The tensor tubal rank can maintain more structural information. The tensor nuclear norm (TNN) [28–31] can be regarded as the convex surrogate of tensor tubal rank. Lu et al. [32,33] have proposed a TNN-based TRPCA program and it can be mathematically expressed as:

$$\min_{\mathcal{L},\mathcal{S}} \|\mathcal{L}\|_* + \lambda \|\mathcal{S}\|_1, \quad \text{s.t.} \quad \mathcal{X} = \mathcal{L} + \mathcal{S}, \tag{3}$$

where $\mathcal{X}$, $\mathcal{L}$, and $\mathcal{S}$ denote respectively observed data, the underlying low-rank tensor and sparse component. $\|\mathcal{L}\|_*$ is the TNN of $\mathcal{L}$ (see section 2.2), which can force $\mathcal{L}$ to keep low tubal rank, and $\|\mathcal{S}\|_1 = \sum_{i,j,k} |s_{i,j,k}|$ is l_1 norm of $\mathcal{S}$, whose function is to guarantee the reasonable sparsity of $\mathcal{S}$. In recent years, tensor network decompositions, such as tensor train (TT) decomposition [34], tensor ring (TR) decomposition [35], and fully-connected tensor network (FCTN) decomposition [36,37], have attracted much attention for analysis of high-dimensional tensor data. Thus, tensor network decomposition-based TRPCA methods have also been studied. For example, Yang et al. [38] proposed a TRPCA model which minimizes the TT nuclear norm. Xu et al. [39] provided a TR decomposition-based program for the mixed noise removing of hyperspectral image. Liu et al. [40] leveraged the superior expression of FCTN for robust tensor completion problem. Although these TRPCA approaches work well in recovering corrupted images, they ignore the importance of local detail information due to only taking into account the global characteristic of images.

To preserve local detail information of image, some total variation (TV) regularization-based image recovery methods have extensively sprung up [41–43]. TV regularization, as a popular tool in image recovery, has the ability to effectively maintain edge information and facilitate piecewise smooth features of images. The work [44] proposed the low-rank matrix method based on TV regularization (LRTV) to remove mixed noise in images. Similarly, in [45], a model which denotes to minimizing TV and weighted nuclear norm (WNMM) was introduced. Via extending traditional TV to 3-D spatio-spectral total variation (SSTV) [46], Wang et al. [47] incorporated the anisotropic SSTV regularization into low-rank tucker decomposition (LRTDTV) for recovering images. Likewise, Chen et al. [48] developed a TRPCA approach based on TNN and SSTV. Besides, an image recovery method that used local low-rank matrix recovery and global spatial-spectral total variation (LLRGTV) is presented in [49]. Although the above mentioned denoising methods based on TV regularization show a series of advantages, TV still faces some problems, such as the undesirable staircase effect in smooth region. For the sake of getting over the unfavorable influence, total generalized variation (TGV) regularization was developed in [50], and it has been proven to reduce staircase effect caused by TV regularization, keep the sharp edges, and have the ability to keep the detail features in images. Unlike TV, which is calculated by first-order derivative, TGV involves first-order and higher-order derivatives. TGV has already been successfully applied in vari-

Fig. 1. The recovered results by TRPCA [33], LRTDTV [47], and the proposed TRPCA-TGV on color image *pallon* with noise level 10%.

ous applications, e.g., removing the additive noise [51–53] and gaussian noise [54,55], deblurring [56], and compressive imaging [57].

Enlightened by TGV, we exploit a novel TRPCA method by introducing TGV into TNN (TRPCA-TGV) to fully consider the global structural information and local detail information of high-dimensional data. Specifically, we employ TNN to preserve the global low-rank prior information. Moreover, we introduce TGV to preserve the local detail information and sharp edge information. To show the advantages of TRPCA-TGV, in Fig. 1, taking color image *pallon* as an example, we show the restoration results of our method TRPCA-TGV and two representative comparison methods (one TRPCA-based method and one TV-based method) under the noise level 10%. We observe that due to only considering the global prior of the image, the recovery result obtained by TRPCA loses some image details and preserves poorer edge effects. LRTDTV obtains the suboptimal result, but its detail features are not so well and it has the obvious staircase effect in smooth region. Significantly, the proposed method TRPCA-TGV takes into account both the global and local prior information of the image, so the overall structure, edges, and some details of the image are well restored. These results of the experiment confirm the outstanding performance of the proposed method. The main contributions of this paper are summarized as three aspects.

- We develop a new TRPCA program by incorporating TGV regularization. The proposed TRPCA-TGV model simultaneously explores the global and local prior information of high-dimensional images.
- The well-know alternating direction method of multiplier (ADMM) [58] framework is introduced to convert the original complex problem into several easily solvable subproblems.
- Extensive numerical experiments on color images and videos demonstrate the superiority of our proposed method over the compared methods, from the aspects of both quantitative assessments and visual effects.

The rest of this paper organizes as follows. Section 2 gives some preliminary knowledge. We introduce the proposed model and algorithm in Section 3. Section 4 evaluates the performance of the experimental results. We present an overview of this work in Section 5.

2. Notations and preliminaries

In this section, we present the basic notations and the necessary definitions throughout this work.

2.1. Notations

We respectively employ lowercase letter (e.g., a), bold uppercase letter (e.g., $\mathbf{A}$), and calligraphic letter (e.g., $\mathcal{A}$) to represent vector, matrix, and tensor. Given a third-order tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, we use the symbols $\mathcal{A}(i,:,:)$, $\mathcal{A}(:,i,:)$, and $\mathcal{A}(:,:,i)$ to express its i-th horizontal, lateral, and frontal slices. For convenience, $\mathcal{A}(:,:,i)$ is written as $\mathcal{A}^{(i)}$. $\mathcal{A}(i,j,:)$ denotes the mode-3 fiber of $\mathcal{A}$. The element of $\mathcal{A}$ is represented by $a_{i,j,k}$. The inner product of tensors $\mathcal{A}$ and $\mathcal{B}$ of size $n_1 \times n_2 \times n_3$ is computed as $\langle \mathcal{A}, \mathcal{B} \rangle = \sum_{i,j,k} a_{i,j,k} \cdot b_{i,j,k}$. The Frobenius and l_1 norms of $\mathcal{A}$ are respectively defined as $\|\mathcal{A}\|_F = \sqrt{\langle \mathcal{A}, \mathcal{A} \rangle} = \sqrt{\sum_{i,j,k} a_{i,j,k}^2}$ and $\|\mathcal{A}\|_1 = \sum_{i,j,k} |a_{i,j,k}|$. $\hat{\mathcal{A}} = \mathrm{fft}(\mathcal{A}, [\,], 3)$ represents the Fourier transformation along the third mode of $\mathcal{A}$. Conversely, the inverse Fourier transformation is expressed as $\mathcal{A} = \mathrm{ifft}(\hat{\mathcal{A}}, [\,], 3)$. We use $\lceil s \rceil$ to represent the smallest integral value that is s or more.

2.2. Tensor nuclear norm (TNN)

Definition 2.1 *(t-product [33]).* Let $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ and $\mathcal{B} \in \mathbb{R}^{n_2 \times n_4 \times n_3}$, the t-product $C = \mathcal{A} * \mathcal{B} \in \mathbb{R}^{n_1 \times n_4 \times n_3}$ is defined as follows:

$$C(i,j,:) = \sum_{k=1}^{n_2} \mathcal{A}(i,k,:) \star \mathcal{B}(k,j,:),$$

where the notation $\star$ represents the circular convolution of $\mathcal{A}(i,k,:)$ and $\mathcal{B}(k,j,:)$.

Definition 2.2 *(tensor conjugate transpose [33]).* For $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, its conjugate transpose is a tensor $\mathcal{A}^H \in \mathbb{R}^{n_2 \times n_1 \times n_3}$, which can be got via conjugate transposing all frontal slices and then reversing the order of transposed frontal slices from 2 to n_3.

Definition 2.3 *(identify tensor [33]).* $\mathcal{I} \in \mathbb{R}^{n_1 \times n_1 \times n_3}$ is a identity tensor if its first frontal slice is the $n_1 \times n_1$ identify matrix, and all other frontal slices are zeros.

Definition 2.4 *(orthogonal tensor [33]).* If a tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_1 \times n_3}$ satisfies

$$\mathcal{A}^H * \mathcal{A} = \mathcal{A} * \mathcal{A}^H = \mathcal{I},$$

then $\mathcal{A}$ is orthogonal.

Definition 2.5 *(f-diagonal tensor [33]).* A third-order tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is f-diagonal whose all frontal slices $\mathcal{A}^{(i)}, i = 1, \cdots, n_3$ are diagonal matrices.

Theorem 2.1 *(t-SVD [33]). For $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, then $\mathcal{A}$ can be decomposed into*

$$\mathcal{A} = \mathcal{U} * \mathcal{S} * \mathcal{V}^H, \tag{4}$$

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

Definition 2.6 *(tensor tubal-rank and multi-rank [33]).* For $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, the tubal-rank of $\mathcal{A}$, denoted as $\text{rank}_t(\mathcal{A})$, is defined to be the maximum rank of all frontal slices $\mathcal{S}^{(i)}$,

$$\text{rank}_t(\mathcal{A}) = \#\{i : \mathcal{S}(i,i,:) \neq 0\}, \tag{5}$$

where $\mathcal{S}$ is f-diagonal, and can be obtained by t-SVD of $\mathcal{A}$. The multi-rank of $\mathcal{A}$, denoted by $\text{rank}_m(\mathcal{A}) \in \mathbb{R}^{n_3}$, is a vector whose i-th entry is the rank of i-th frontal slice of $\hat{\mathcal{A}}$. And $\text{rank}_t(\mathcal{A}) = \max(\text{rank}_m(\mathcal{A}))$.

Definition 2.7 *(TNN [33]).* The tensor nuclear norm of $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is the average of the sum of singular values of all the frontal slices of $\hat{\mathcal{A}}$, i.e.,

$$\|\mathcal{A}\|_* := \frac{1}{n_3} \sum_{i=1}^{n_3} \|\hat{\mathbf{A}}^{(i)}\|_*, \tag{6}$$

where $\hat{\mathbf{A}}^{(i)}$ is the i-th frontal slice of $\hat{\mathcal{A}}$.

2.3. Total generalized variation (TGV)

To illustrate the definition of TGV, assuming that $\Omega \in \mathbb{R}^d, d \in \mathbf{N}, d \geq 1$ is a domain which is a non-empty and bounded domain. TGV [50] of order k with weights $\boldsymbol{\alpha} = (\alpha_0, \alpha_1, \cdots, \alpha_{k-1}) > 0$ for an image $u : \Omega \to \mathbb{R}^+$ is defined as:

$$\text{TGV}_{\boldsymbol{\alpha}}^k(u) = \sup\left\{ \int_\Omega u \text{div}^k v dx | v \in C_c^k(\Omega, \text{Sym}^k(\mathbb{R}^d)), \|\text{div}^j v\|_\infty \leq \alpha_j, j = 0, 1, \cdots, k-1 \right\}, \tag{7}$$

where $C_c^k(\Omega, \text{Sym}^k(\mathbb{R}^d))$ denotes the space of symmetric tensor fields with the compact support in Ω, and $\text{Sym}^k(\mathbb{R}^d)$ is the space of symmetric kth-order tensors on $\mathbb{R}^d$ which can be formulated as:

$$\text{Sym}^k(\mathbb{R}^d) = \{\xi : \underbrace{\mathbb{R}^d \times \cdots \times \mathbb{R}^d}_{k} \longrightarrow \mathbb{R} | \xi \text{ is multilinear and symmetric}\}.$$

Clearly, $\text{TGV}_1^1(u)$ is the TV of u when $k = 1$, $\boldsymbol{\alpha} = 1$, $\text{Sym}^1(\mathbb{R}^d) = \mathbb{R}^d$. In the case of $k = 2$, $\text{Sym}^2(\mathbb{R}^d)$ is the space $\mathbf{S}^{d \times d}$ made up of all symmetric $d \times d$ matrices.

Since $\text{TGV}_{\boldsymbol{\alpha}}^k(u)$ contains high-order regularization compared with traditional TV, it can better preserve sharp image edges and detail information [59]. Inspired by the above, we use $k = 2$ in this work. The second-order TGV can be formulated as:

$$\mathrm{TGV}^2_{\boldsymbol{\alpha}}(u) = \sup\left\{ \int_\Omega u \mathrm{div}^2 v dx \,|\, v \in C_c^2(\Omega, \mathrm{Sym}^2(\mathbb{R}^d)), \|v\|_\infty \le \alpha_0, \|\mathrm{div} v\|_\infty \le \alpha_1 \right\}. \tag{8}$$

For $v \in \mathbf{S}^{d\times d}$, the divergences are given by:

$$(\mathrm{div} v)_i = \sum_{j=1}^{d} \frac{\partial v_{ij}}{\partial x_j}, 1 \le i \le d, \quad \mathrm{div}^2 v = \sum_{i,j=1}^{d} \frac{\partial^2 v_{ij}}{\partial x_i \partial x_j},$$

and the infinity norms of v and $\mathrm{div} v$ can be expressed as:

$$\|v\|_\infty = \max_{x\in\Omega} \left(\sum_{i,j=1}^{d} |v_{i,j}(x)|^2 \right)^{1/2}, \quad \|\mathrm{div} v\|_\infty = \max_{x\in\Omega} \left(\sum_{i=1}^{d} |(\mathrm{div} v)_i(x)|^2 \right)^{1/2}.$$

We derive another form of $\mathrm{TGV}^2_{\boldsymbol{\alpha}}(u)$ to solve it efficiently. We have $d = 2$ for images. Here we define $\mathbf{U}, \mathbf{V}, \mathbf{W}$ as:

$$\mathbf{U} = C_c^2(\Omega, \mathbb{R}), \quad \mathbf{V} = C_c^2(\Omega, \mathbb{R}^2), \quad \mathbf{W} = C_c^2(\Omega, \mathbf{S}^{2\times 2}). \tag{9}$$

According to [51,59], $\mathrm{TGV}^2_{\boldsymbol{\alpha}}(u)$ can be simply reformulated as:

$$\mathrm{TGV}^2_{\boldsymbol{\alpha}}(u) = \min_{p\in\mathbf{V}} \alpha_1 \|Du - p\|_1 + \alpha_0 \|\mathcal{E}(p)\|_1, \tag{10}$$

where $\mathcal{E}(p) = \frac{1}{2}(Dp + (Dp)^T)$, $D = (D_1; D_2)$, D_1 and D_2 respectively denote the first-order forward finite difference operators in x-direction and y-direction. In particular, the operations $D : \mathbf{U} \to \mathbf{V}$ and $\mathcal{E} : \mathbf{V} \to \mathbf{W}$ can be expressed as:

$$Du = \begin{pmatrix} D_1 u \\ D_2 u \end{pmatrix}, \quad \mathcal{E}(p) = \begin{pmatrix} D_1 p_1 & \frac{1}{2}(D_2 p_1 + D_1 p_2) \\ \frac{1}{2}(D_2 p_1 + D_1 p_2) & D_2 p_2 \end{pmatrix}.$$

More details can be seen in [50,54].

3. The proposed model and algorithm

We develop a novel TRPCA model via incorporating the advantage of TGV. Specifically, giving an observed tensor $\mathcal{X} \in \mathbb{R}^{n_1\times n_2\times n_3}$ which is corrupted by sparse noise, the proposed optimization program can be formulated as:

$$\begin{aligned} &\min_{\mathcal{L},\mathcal{S}} \beta\|\mathcal{L}\|_* + \lambda\|\mathcal{S}\|_1 + \mathrm{TGV}^2_{\boldsymbol{\alpha}}(\mathcal{L}) \\ &\text{s.t. } \mathcal{X} = \mathcal{L} + \mathcal{S}, \end{aligned} \tag{11}$$

where $\mathcal{L}$ and $\mathcal{S}$ are low-rank and sparse components with the same size to $\mathcal{X}$, β, and λ are trade-off parameters to adjust the proportion of the above three terms. Our proposed model is formed by three parts. The first part uses TNN to capture the global features of underlying image $\mathcal{L}$. The second part is to suppress sparse noise $\mathcal{S}$ by l_1 norm. The third part introduces the TGV to preserve local detail information. In summary, our method has the following advantages:

- Compared with the general TRPCA recovery models which only consider the global prior information, our model considers both the global and local prior information of high-dimensional data for achieving satisfactory restoration performance.
- Compared with the common TV-based restoration methods which often come with some unfavorable staircase effects in smooth regions of the image, we adopt TGV which can not only fully preserve the local detail information of the image but also effectively suppress the staircase effect.

Solving the minimization problem (11) directly is difficult. Since ADMM [58] is efficient in solving the large-scale optimization problem, we exploit ADMM to solve this minimization problem (11).

Specifically, by introducing the auxiliary variable $\mathcal{Z}$, (11) can be reformulated as:

$$\begin{aligned} &\min_{\mathcal{L},\mathcal{S},\mathcal{Z}} \beta\|\mathcal{L}\|_* + \lambda\|\mathcal{S}\|_1 + \mathrm{TGV}^2_{\boldsymbol{\alpha}}(\mathcal{Z}) \\ &\text{s.t. } \mathcal{X} = \mathcal{L} + \mathcal{S}, \mathcal{L} = \mathcal{Z}. \end{aligned} \tag{12}$$

And then, the augmented Lagrangian function of (12) is

$$L(\mathcal{L},\mathcal{S},\mathcal{Z},\mathcal{W},\mathcal{B}) = \beta\|\mathcal{L}\|_* + \lambda\|\mathcal{S}\|_1 + \mathrm{TGV}^2_{\boldsymbol{\alpha}}(\mathcal{Z}) + <\mathcal{W}, \mathcal{L}+\mathcal{S}-\mathcal{X}> + \frac{\gamma}{2}\|\mathcal{L}+\mathcal{S}-\mathcal{X}\|_F^2 + <\mathcal{B}, \mathcal{L}-\mathcal{Z}> + \frac{\gamma}{2}\|\mathcal{L}-\mathcal{Z}\|_F^2, \tag{13}$$

where $\mathcal{W}$ and $\mathcal{B}$ are the Lagrangian multipliers, $\gamma > 0$ is a penalty parameter.

In ADMM scheme, we alternately update each variable at a time, while fixing the others. The solutions to solving a sequence of subproblems are given as follows:

1)$\mathcal{L}$-subproblem: Retaining all terms about $\mathcal{L}$, we can get

$$\begin{aligned}\mathcal{L}_{k+1} &= \arg\min_{\mathcal{L}} \beta\|\mathcal{L}\|_* + \frac{\gamma_k}{2}\left(\|\mathcal{L}+\mathcal{S}_k-\mathcal{X}+\frac{\mathcal{W}_k}{\gamma_k}\|_F^2 + \|\mathcal{L}-\mathcal{Z}_k+\frac{\mathcal{B}_k}{\gamma_k}\|_F^2\right)\\ &= \arg\min_{\mathcal{L}} \frac{\beta}{\gamma_k}\|\mathcal{L}\|_* + \frac{1}{2}\|\mathcal{L}-C_k\|_F^2,\end{aligned} \tag{14}$$

where $C_k = \frac{\gamma_k(\mathcal{X}+\mathcal{Z}_k-\mathcal{S}_k)-(\mathcal{W}_k+\mathcal{B}_k)}{2\gamma_k}$. Assume $\mathcal{U}*\mathcal{D}*\mathcal{V}^H$ is the t-SVD of C_k, (14) can be solved under tensor singular value thresholding (t-SVT) [33] as follows:

$$\mathcal{L}_{k+1} = \mathcal{U} * \mathcal{D}_{\frac{\beta}{\gamma_k}} * \mathcal{V}^H, \tag{15}$$

where

$$\mathcal{D}_{\frac{\beta}{\gamma_k}} = \mathrm{ifft}((\hat{\mathcal{D}} - \frac{\beta}{\gamma_k})_+, [\,], 3).$$

Note that $\hat{\mathcal{D}} = \mathrm{fft}(\mathcal{D}, [\,], 3)$, and $a_+ = \max(a, 0)$.

2)$\mathcal{S}$-subproblem: Extracting all terms containing $\mathcal{S}$ from (13), we can update $\mathcal{S}$ by solving:

$$\begin{aligned}\mathcal{S}_{k+1} &= \arg\min_{\mathcal{S}} \lambda\|\mathcal{S}\|_1 + \frac{\gamma_k}{2}\|\mathcal{L}_{k+1}+\mathcal{S}-\mathcal{X}+\frac{\mathcal{W}_k}{\gamma_k}\|_F^2\\ &= \arg\min_{\mathcal{S}} \frac{\lambda}{\gamma_k}\|\mathcal{S}\|_1 + \frac{1}{2}\|\mathcal{S}-(\mathcal{X}-\mathcal{L}_{k+1}-\frac{\mathcal{W}_k}{\gamma_k})\|_F^2.\end{aligned} \tag{16}$$

$\mathcal{S}_{k+1}$ can be updated via soft-thresholding operator [60] as:

$$\mathcal{S}_{k+1} = \mathrm{shrink}(\mathcal{X}-\mathcal{L}_{k+1}-\frac{\mathcal{W}_k}{\gamma_k}, \frac{\lambda}{\gamma_k}), \tag{17}$$

where $\mathrm{shrink}(p,\sigma) = \mathrm{sgn}(p).*\max(|p|-\sigma, 0)$, $\mathrm{sgn}(\cdot)$ represents the sign function.

3)$\mathcal{Z}$-subproblem: Keeping all terms relative to $\mathcal{Z}$ in (13), the solution of $\mathcal{Z}$ can be formed as:

$$\mathcal{Z}_{k+1} = \arg\min_{\mathcal{Z}} \frac{\gamma_k}{2}\|\mathcal{L}_{k+1}-\mathcal{Z}+\frac{\mathcal{B}_k}{\gamma_k}\|_F^2 + \mathrm{TGV}_{\boldsymbol{\alpha}}^2(\mathcal{Z}). \tag{18}$$

To solve (18), we can transform it into solving each frontal slice $\mathcal{Z}_{k+1}^{(i)}$ of $\mathcal{Z}_{k+1}$, $i = 1, 2, \cdots, n_3$,

$$\mathcal{Z}_{k+1}^{(i)} = \arg\min_{\mathcal{Z}^{(i)}} \frac{\gamma_k}{2}\|\mathcal{Z}^{(i)} - (\mathcal{L}_{k+1}^{(i)} + \frac{\mathcal{B}_k^{(i)}}{\gamma_k})\|_F^2 + \mathrm{TGV}_{\boldsymbol{\alpha}}^2(\mathcal{Z}^{(i)}). \tag{19}$$

According to equation (10) in Section 2, we rewrite (19) as:

$$\mathcal{Z}_{k+1}^{(i)} = \arg\min_{\mathcal{Z}^{(i)},p} \frac{\gamma_k}{2}\|\mathcal{Z}^{(i)} - (\mathcal{L}_{k+1}^{(i)} + \frac{\mathcal{B}_k^{(i)}}{\gamma_k})\|_F^2 + \alpha_1\|D\mathcal{Z}^{(i)} - p\|_1 + \alpha_0\|\mathcal{E}(p)\|_1. \tag{20}$$

Numerous achievements have been made to explore the effective solutions for TGV problem [51,55,61]. Here, we employ the primal-dual method [53] to solve (20).

Besides, the Lagrangian multipliers and penalty parameter can be updated through:

$$\mathcal{W}_{k+1} = \mathcal{W}_k + \gamma_k(\mathcal{L}_{k+1}+\mathcal{S}_{k+1}-\mathcal{X}), \tag{21}$$

$$\mathcal{B}_{k+1} = \mathcal{B}_k + \gamma_k(\mathcal{L}_{k+1}-\mathcal{Z}_{k+1}), \tag{22}$$

$$\gamma_{k+1} = \min\{\theta\gamma_k, \gamma_{max}\}. \tag{23}$$

Algorithm 1 presents the optimization procedure of TRPCA-TGV.

4. Numerical experiments

We give some experimental results to display the recovery performance of our TRPCA-TGV for color images and videos denoising. Our method is compared with one method based on t-SVD (TRPCA [33]) and two methods based on TV (LLRGTV [49] and LRTDTV [47]).

Algorithm 1 TRPCA-TGV solver.

Input: $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, ϵ, λ, β, θ, maximum iteration K.
Output: The recovered image $\mathcal{L}_{k+1}$.
Initialize: $\mathcal{L}_0 = \mathcal{S}_0 = \mathcal{W}_0 = \mathcal{Z}_0 = \mathcal{B}_0 = \{0\}^{n_1 \times n_2 \times n_3}$, $\gamma_0 = 10^{-2}$, $\gamma_{max} = 10^6$, $k = 0$.
1: **While** not converged and $k < K$ **do**
2: update $\mathcal{L}_{k+1}$ via (15).
3: update $\mathcal{S}_{k+1}$ via (17).
4: update $\mathcal{Z}_{k+1}$ via (18).
5: update $\mathcal{W}_{k+1}$, $\mathcal{B}_{k+1}$, and γ_{k+1} by (21), (22), and (23).
6: check the convergence conditions.
7: **end While**

Fig. 2. The original color images.

In general, we employ the peak signal-to-noise ratio (PSNR) and the structural similarity index (SSIM) [62] to measure the quality of the recovered results. The average PSNR and SSIM values of all bands are calculated and reported. Higher quality recovery result usually corresponds to the larger PSNR and SSIM values.

We set the maximum number of iteration $K = 500$. For our stopping criterion, let

$$chg\mathcal{L} = \|\mathcal{L}_{k+1} - \mathcal{L}_k\|_\infty,$$
$$chg\mathcal{S} = \|\mathcal{S}_{k+1} - \mathcal{S}_k\|_\infty,$$
$$chg\mathcal{Z} = \|\mathcal{Z}_{k+1} - \mathcal{Z}_k\|_\infty,$$
$$chg\mathcal{X} = \|\mathcal{X} - \mathcal{L}_{k+1} - \mathcal{S}_{k+1}\|_\infty,$$
$$chg\mathcal{Y} = \|\mathcal{L}_{k+1} - \mathcal{Z}_{k+1}\|_\infty,$$

where $\|\mathcal{A}\|_\infty = \max_{i,j,k} |a_{i,j,k}|$ for a third-order tensor. In iterative procedure, the algorithm will be stopped once the following final criterion is met:

$$\max\{chg\mathcal{L}, chg\mathcal{S}, chg\mathcal{Z}, chg\mathcal{X}, chg\mathcal{Y}\} \le \varepsilon, \tag{24}$$

where ε is the convergence tolerance. In all experiments, ε is set to 10^{-5}.

All codes are implemented using MATLAB R2021a on a laptop with Intel core i5-12500 CPU (2.5GHz) and 16.00GB RAM, and NVIDIA GeForce RTX2050 4GB.

4.1. Color images recovery

In this subsection, we evaluate the performance of our method with the compared methods on ten color images. All test images have the same size $256 \times 256 \times 3$, as shown in Fig. 2.

The noisy images are obtained by randomly setting 10%, 20%, 30%, and 40% of pixels to random values in $[0, 255]$. Table 1 is the quantitative comparison of different approaches for ten color images. We have marked the highest PSNR and SSIM values and least execution times using black bold font. It is evident that our method has the maximum PSNR and SSIM values for all images. Even if the noise level has reached 40%, our method still performs better than the compared meth-

Table 1
Quantitative evaluation of results obtained by TRPCA, LLRGTV, LRTDTV and TRPCA-TGV for color images with different noise levels.

Image	Noise Levels	10%			20%			30%			40%		
	Method	PSNR	SSIM	time	PSNR	SSIM	time	PSNR	SSIM	time	PSNR	SSIM	time
peppers	TRPCA	27.03	0.8944	3.41	25.12	0.7904	3.41	22.29	0.5682	3.27	18.40	0.3170	3.19
	LLRGTV	23.73	0.7392	2.93	21.27	0.5383	3.14	19.23	0.4240	3.17	17.15	0.4454	3.23
	LRTDTV	28.66	0.8430	5.40	24.14	0.6887	5.49	20.81	0.5486	5.61	17.62	0.3678	8.06
	TRPCA-TGV	**29.60**	**0.9045**	**1.50**	**25.44**	**0.7981**	**2.69**	**22.59**	**0.6391**	**2.72**	**19.50**	**0.5070**	**2.98**
pallon	TRPCA	31.14	0.9310	3.45	29.62	0.8783	3.37	26.96	0.7098	3.40	22.40	0.3916	3.25
	LLRGTV	26.82	0.7161	3.09	23.44	0.5468	3.10	21.68	0.4496	3.11	19.66	0.4164	3.06
	LRTDTV	31.65	0.8430	5.74	26.61	0.6749	5.62	22.31	0.5669	5.47	19.93	0.3928	5.42
	TRPCA-TGV	**34.17**	**0.9423**	**1.54**	**30.25**	**0.8793**	**1.54**	**27.12**	**0.7716**	**2.39**	**24.13**	**0.6508**	**2.69**
lena	TRPCA	29.92	0.9291	3.46	28.04	0.8616	3.46	22.23	0.6786	3.20	22.29	0.4117	3.09
	LLRGTV	26.47	0.7673	2.98	23.54	0.7146	3.06	21.67	0.5806	2.96	20.11	0.5383	2.98
	LRTDTV	31.37	0.8763	5.62	26.97	0.7323	5.49	23.60	0.5670	5.64	20.98	0.5079	5.27
	TRPCA-TGV	**33.03**	**0.9347**	**1.54**	**28.32**	**0.8676**	**1.46**	**25.49**	**0.7433**	**1.51**	**22.87**	**0.6581**	**1.51**
starfish	TRPCA	30.33	0.9421	3.47	27.14	0.8530	3.29	22.98	0.6337	3.18	18.80	0.3804	3.11
	LLRGTV	26.33	0.8545	2.98	23.11	0.7684	2.95	20.53	0.6510	3.03	18.60	0.5299	2.96
	LRTDTV	28.82	0.8727	5.42	24.86	0.7711	5.52	21.70	0.6463	5.51	19.37	0.5191	5.52
	TRPCA-TGV	**33.09**	**0.9507**	**1.64**	**27.43**	**0.8578**	**2.06**	**23.45**	**0.6988**	**2.32**	**20.29**	**0.5423**	**2.47**
leaf	TRPCA	31.69	0.9635	3.59	28.78	0.9086	3.42	25.60	0.7544	3.29	21.44	0.4768	3.11
	LLRGTV	29.26	0.8853	3.01	26.20	0.7688	3.01	23.48	0.6804	2.97	21.04	0.5531	2.97
	LRTDTV	33.02	0.9669	5.56	28.59	0.8538	5.60	25.08	0.7298	5.87	22.06	0.5982	5.69
	TRPCA-TGV	**35.01**	**0.9714**	**1.44**	**28.97**	**0.9107**	**2.17**	**25.84**	**0.7816**	**2.38**	**22.87**	**0.6419**	**2.62**
palace	TRPCA	28.25	0.9427	3.48	25.95	0.8765	3.48	22.96	0.7079	3.20	19.10	0.4712	3.20
	LLRGTV	27.80	0.8617	**2.95**	22.95	0.7331	3.16	19.66	0.6138	**3.00**	17.99	0.5130	**2.99**
	LRTDTV	28.62	0.8969	4.74	25.32	0.7958	6.12	22.47	0.7071	5.73	22.12	0.6119	5.39
	TRPCA-TGV	**31.70**	**0.9490**	3.35	**26.20**	**0.8789**	**2.83**	**23.15**	**0.7364**	4.09	**22.14**	**0.6177**	3.84
resort	TRPCA	30.51	0.9381	3.49	28.57	0.8814	3.42	25.87	0.7107	3.24	21.98	0.4360	3.12
	LLRGTV	30.17	0.8599	2.91	26.41	0.7442	2.96	23.71	0.6478	2.92	21.74	0.5626	3.01
	LRTDTV	30.23	0.8583	5.49	27.24	0.7463	5.49	24.81	0.6348	5.45	22.66	0.5250	5.49
	TRPCA-TGV	**33.95**	**0.9553**	**1.46**	**29.05**	**0.8821**	**1.45**	**26.14**	**0.7691**	**2.22**	**23.80**	**0.6489**	**2.58**
mountains	TRPCA	38.74	0.9748	3.62	36.70	0.9546	3.35	34.31	0.9009	3.31	28.30	0.6089	3.23
	LLRGTV	33.38	0.9235	2.94	29.27	0.8293	2.90	24.75	0.7083	2.98	21.55	0.7522	2.98
	LRTDTV	36.36	0.9045	5.62	31.62	0.8019	5.54	27.10	0.6700	5.60	23.67	0.5509	5.58
	TRPCA-TGV	**42.89**	**0.9827**	**1.40**	**37.69**	**0.9558**	**1.43**	**34.83**	**0.9225**	**1.45**	**31.50**	**0.8753**	**2.29**
pyramid	TRPCA	34.07	0.9422	3.41	32.26	0.8954	3.35	29.81	0.7721	3.28	25.51	0.4978	3.17
	LLRGTV	31.74	0.7911	2.94	26.88	0.6690	2.93	23.00	0.5293	3.01	20.30	0.4289	3.02
	LRTDTV	31.66	0.8058	5.40	27.50	0.6554	5.45	24.02	0.5147	5.10	21.51	0.4130	5.63
	TRPCA-TGV	**37.25**	**0.9637**	**1.40**	**32.76**	**0.9003**	**1.41**	**30.25**	**0.8055**	**1.57**	**27.78**	**0.7140**	**1.66**
church	TRPCA	32.85	0.9652	3.47	31.44	0.9460	3.43	29.35	0.8850	3.31	24.57	0.5341	3.22
	LLRGTV	32.34	0.9157	3.01	27.49	0.8012	2.96	23.49	0.6746	2.92	20.31	0.5613	2.96
	LRTDTV	35.89	0.9211	5.33	30.88	0.8068	5.48	27.08	0.6917	5.49	23.80	0.5761	5.36
	TRPCA-TGV	**36.89**	**0.9749**	**1.47**	**32.24**	**0.9496**	**1.41**	**29.45**	**0.8933**	**1.43**	**25.68**	**0.8145**	**1.46**

ods. Meanwhile, the iteration times of TRPCA-TGV are less than the compared methods except the execution times for image *palace*.

In Fig. 3, some restored images with different noise levels are exhibited. We have selected the same subdomain of each image and enlarged it so as to better showcase the comparison effect in vision. It is clear that our method has successfully removed noise and can capture more realistic details in recovering images. Under the condition of noise level 10%, TRPCA can eliminate the noise, but its detail features are unsharp. The chief reason is that TRPCA has focused just on the global low-rank prior information, but ignored the importance of local information for detail features. The LLRGTV has used the method of local low-rank matrix, this has essentially broken the interior structure of tensor. So the recovery results of LLRGTV are unsatisfactory. The recovery of LRTDTV has the staircase effect due to the usage of SSTV which is the form of first-oder derivative. Overall, the proposed method has achieved a more satisfactory result, and it can preserve both the underlying global information and better local detail features. In addition, the proposed method also has the ability to powerfully denoise when noise level is 40%.

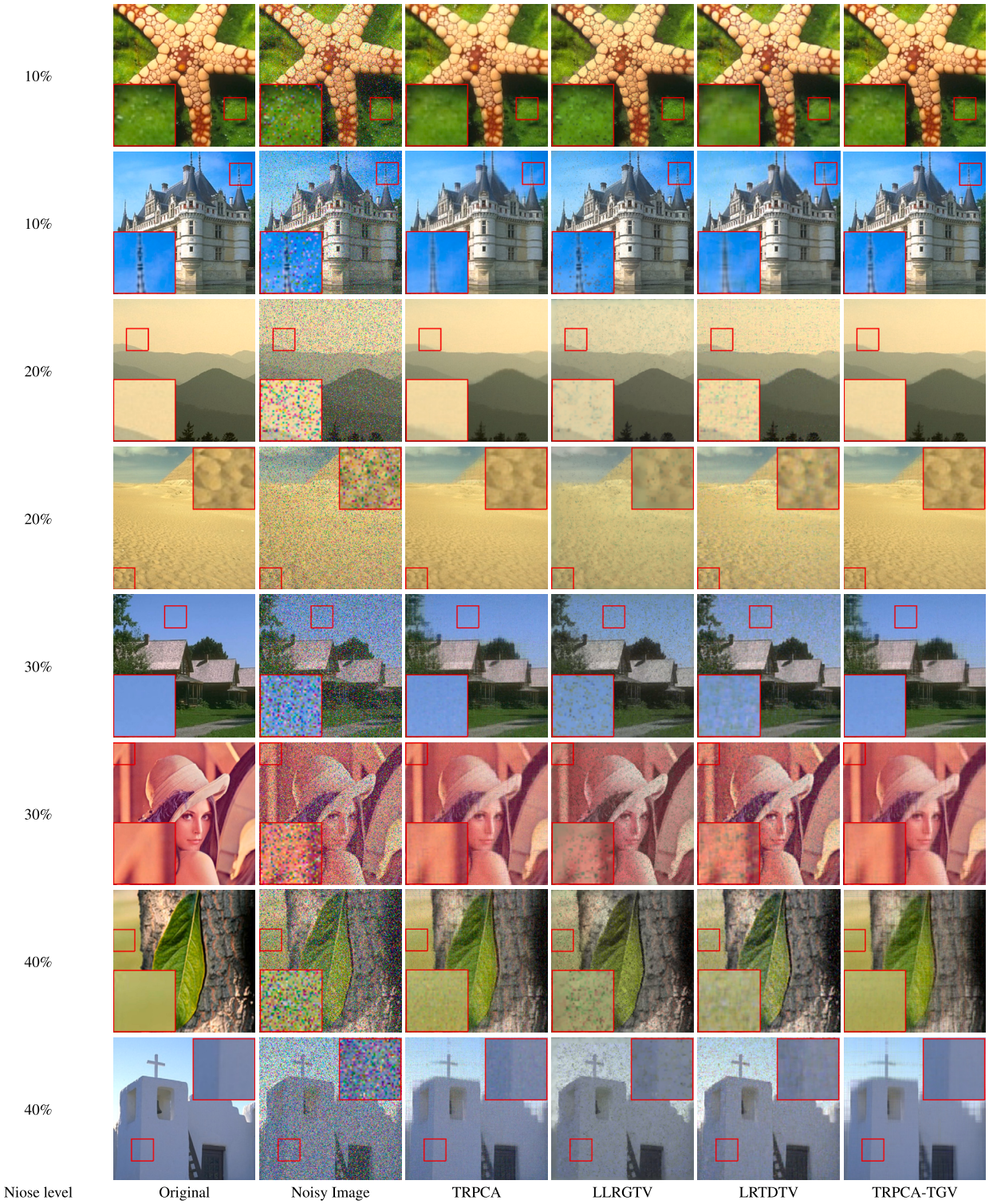

Fig. 3. The recovered images by different methods with noise levels 10%, 20%, 30%, and 40%.

Table 2
Quantitative evaluation of results obtained by TRPCA, LLRGTV, LRTDTV, and TRPCA-TGV for videos with different noise levels.

Video	Noise Levels	10%			20%			30%		
	Method	PSNR	SSIM	time	PSNR	SSIM	time	PSNR	SSIM	time
	TRPCA	26.27	0.9057	28.08	25.04	0.8453	27.42	23.22	0.7123	26.76
bus	LLRGTV	20.44	0.5341	33.48	18.62	0.4149	31.76	19.16	0.4464	32.71
$256 \times 256 \times 30$	LRTDTV	21.18	0.6273	29.94	21.03	0.5924	28.60	20.83	0.5807	24.78
	TRPCA-TGV	**29.41**	**0.9209**	**12.29**	**25.61**	**0.8489**	**12.14**	**23.45**	**0.7513**	**12.30**
	TRPCA	33.05	0.9345	26.35	31.97	0.9191	25.40	30.67	0.8899	24.76
suzie	LLRGTV	28.42	0.8137	83.92	28.33	0.8109	81.16	28.16	0.8052	82.41
$144 \times 176 \times 75$	LRTDTV	30.21	0.8730	22.30	30.00	0.8679	22.19	29.73	0.8609	23.03
	TRPCA-TGV	**36.40**	**0.9559**	**11.07**	**33.28**	**0.9239**	**11.50**	**31.04**	**0.8901**	**11.43**
	TRPCA	33.72	0.9698	29.35	33.63	0.9630	27.68	31.48	0.9533	25.90
salesman	LLRGTV	31.44	0.9490	88.10	31.29	0.9471	93.86	30.91	0.9433	82.49
$144 \times 176 \times 75$	LRTDTV	33.41	0.9634	20.94	32.77	0.9584	26.85	31.67	0.9478	23.07
	TRPCA-TGV	**38.25**	**0.9859**	**18.68**	**34.83**	**0.9726**	**12.12**	**32.22**	**0.9567**	**11.96**
	TRPCA	38.22	0.9843	10.59	36.76	0.9808	10.41	34.44	0.9733	10.61
hall	LLRGTV	39.63	0.9832	30.07	37.98	0.9817	30.06	35.20	**0.9747**	28.43
$144 \times 176 \times 30$	LRTDTV	38.86	0.9806	**9.78**	36.96	0.9744	**9.77**	33.33	0.9651	17.23
	TRPCA-TGV	**41.55**	**0.9889**	12.01	**38.69**	**0.9824**	10.75	**35.29**	0.9744	**10.19**
	TRPCA	34.02	0.9603	11.15	32.22	0.9395	10.73	30.20	0.8982	10.34
coastguard	LLRGTV	28.84	0.8452	30.05	28.43	0.8336	30.22	27.71	0.8093	30.24
$144 \times 176 \times 30$	LRTDTV	32.44	0.9192	11.90	31.13	0.8995	12.23	29.46	0.8655	12.45
	TRPCA-TGV	**36.88**	**0.9747**	**4.96**	**33.17**	**0.9443**	**5.00**	**30.50**	**0.9046**	**4.99**
	TRPCA	35.23	0.9802	13.05	34.23	0.9764	12.89	32.92	0.9695	11.52
news	LLRGTV	35.52	0.9819	30.64	34.88	0.9795	30.23	33.51	**0.9739**	30.32
$144 \times 176 \times 30$	LRTDTV	38.57	0.9889	**12.71**	34.64	0.9739	12.61	31.22	0.9407	12.78
	TRPCA-TGV	**40.24**	**0.9896**	13.14	**36.81**	**0.9809**	**12.15**	**33.63**	0.9715	**5.02**

4.2. Videos recovery

In this subsection, six gray videos[1] are tested by TRPCA, LLRGTV, LRTDTV, and our proposed method TRPCA-TGV. Table 2 displays the average PSNR and SSIM values and execution times of the recovered videos by different methods with noise levels 10%, 20%, and 30%. Clearly, our method achieves best recovery results and least iteration times for most test videos. All PSNR values of the videos recovered by our method are highest.

In order to see more recovery details for videos, we select some cases to show the visual contrast. If the number of frames of a video is m, we typically show the $\lceil \frac{m}{2} \rceil$-th frames of the recovery results for videos with noise levels 10% and 20% in Fig. 4. It is evident that, our model is superior to others in the aspect of denoising and detail preservation. For example, in the recovery results of *bus*, the letters on the bus restored by our method are clearer than other comparative methods. Fig. 5 displays the PSNR and SSIM values of each frame for videos *bus* and *suzie* with noise level 10%. Obviously, our method has achieved the best quality indexes for almost all frames. Fig. 6 is the pixel values of tubes randomly selected from the reconstructed videos via different methods with noise level 10%. We can draw that the pixel values of videos recovered by our method are nearest to original pixel values. Therefore, our method is more efficient.

4.3. Discussions

Parameters analysis. In the TRPCA-TGV model, there exist the following parameters: β capturing the low-rankness of the underlying tensor, the sparse regularization parameter λ, penalty parameter γ, θ used for updating γ, the positive weights α_0 and α_1 in TGV. During our experiments, to improve the speed of convergence effectively, we set $\theta = 1.1$. The initial penalty parameter γ_0 is set to 10^{-2}. Inspired by [53], we set the parameters $\alpha_0 = 2$ and $\alpha_1 = 1$ in TGV. According to [33], we also set $\lambda = 1/\sqrt{max(n_1, n_2) \times n_3}$ in TRPCA-TGV. Fig. 7 shows the PSNR and SSIM values of the results restored via TRPCA-TGV for color image *lena* in regard to β. It can be concluded that the recovery results are closely affected by β. On color images recovery, the parameter β is relatively sensitive to different noise levels. For noise levels 10%, 20%, 30% and 40%, βs are respectively selected from the sets $\{0.30, 0.35, 0.45\}$, $\{0.40, 0.45, 0.50\}$, $\{0.50, 0.55, 0.60\}$, and $\{0.60, 0.65, 0.70\}$. Since the video data in our experiments are of different size, the proper parameter βs are shown in Table 3. The βs corresponding to noise levels 10%, 20%, and 30% show an increasing trend.

Convergency behaviors. Taking the color image *peppers* as an example, we show the numerical convergence of TRPCA-TGV. As can be seen in Fig. 8, the relative errors of $\mathcal{L}$ and $\mathcal{S}$, which can be expressed as $\frac{\|\mathcal{L}_{k+1} - \mathcal{L}_k\|_F}{\|\mathcal{L}_k\|_F}$ and $\frac{\|\mathcal{S}_{k+1} - \mathcal{S}_k\|_F}{\|\mathcal{S}_k\|_F}$, rapidly decrease

[1] http://trace.eas.asu.edu/yuv/.

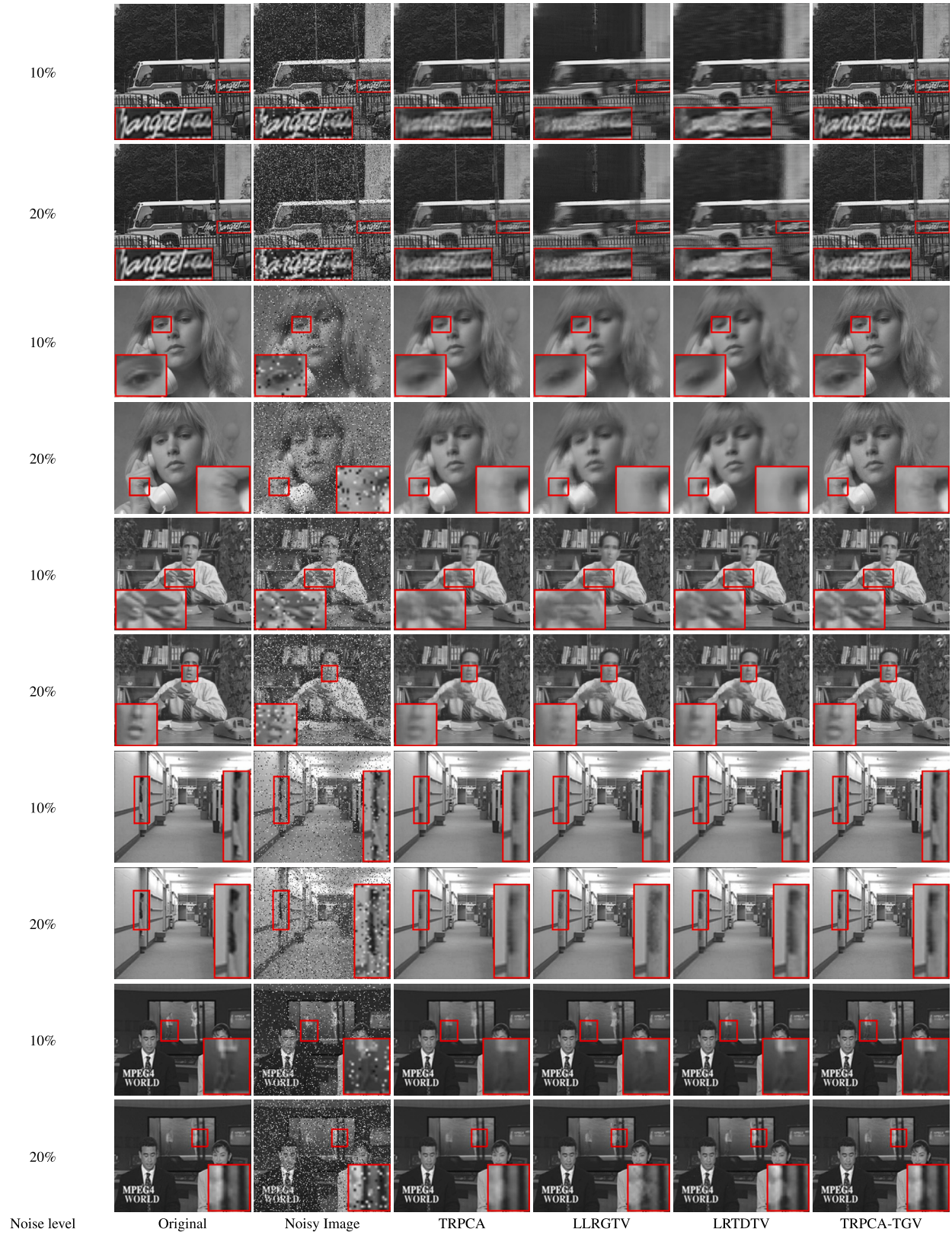

Fig. 4. The recovered videos by different methods with noise levels 10% and 20%. From top to bottom: *bus*, *suzie*, *salesman*, *hall*, and *news*.

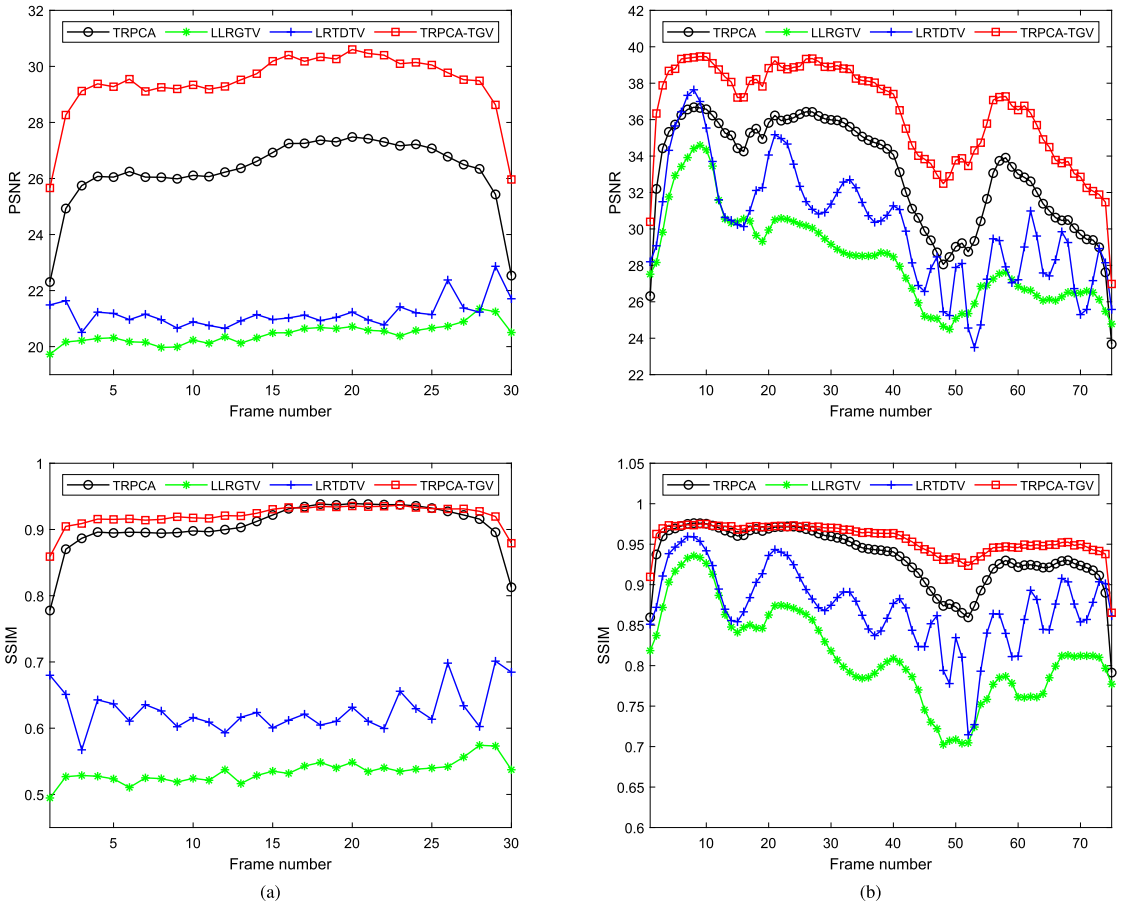

Fig. 5. The PSNR and SSIM values of each frame of videos recovered by different methods with noise level 10%. (a) *bus* and (b) *suzie*.

Table 3
The proper parameter βs for videos with noise levels 10%, 20%, and 30%.

Noise Levels	*bus*	*suzie*	*salesman*	*hall*	*coastguard*	*news*
10%	0.33	0.31	0.28	0.31	0.35	0.27
20%	0.46	0.40	0.37	0.38	0.42	0.36
30%	0.53	0.48	0.48	0.45	0.49	0.47

during the iterations for different noise levels. This clearly verifies the numerical convergence behavior of the proposed algorithm.

5. Conclusions

In this work, we exploited a hybrid model for tensor recovery, in which we introduce total generalized variation into tensor robust principal component analysis for catching the global prior information and local detail features. The ADMM algorithm was employed to solve the proposed problem. A large quantity of numerical experiments have indicated that our method TRPCA-TGV performs superior than the comparative methods TRPCA, LLRGTV and LRTDTV.

Data availability

Data will be made available on request.

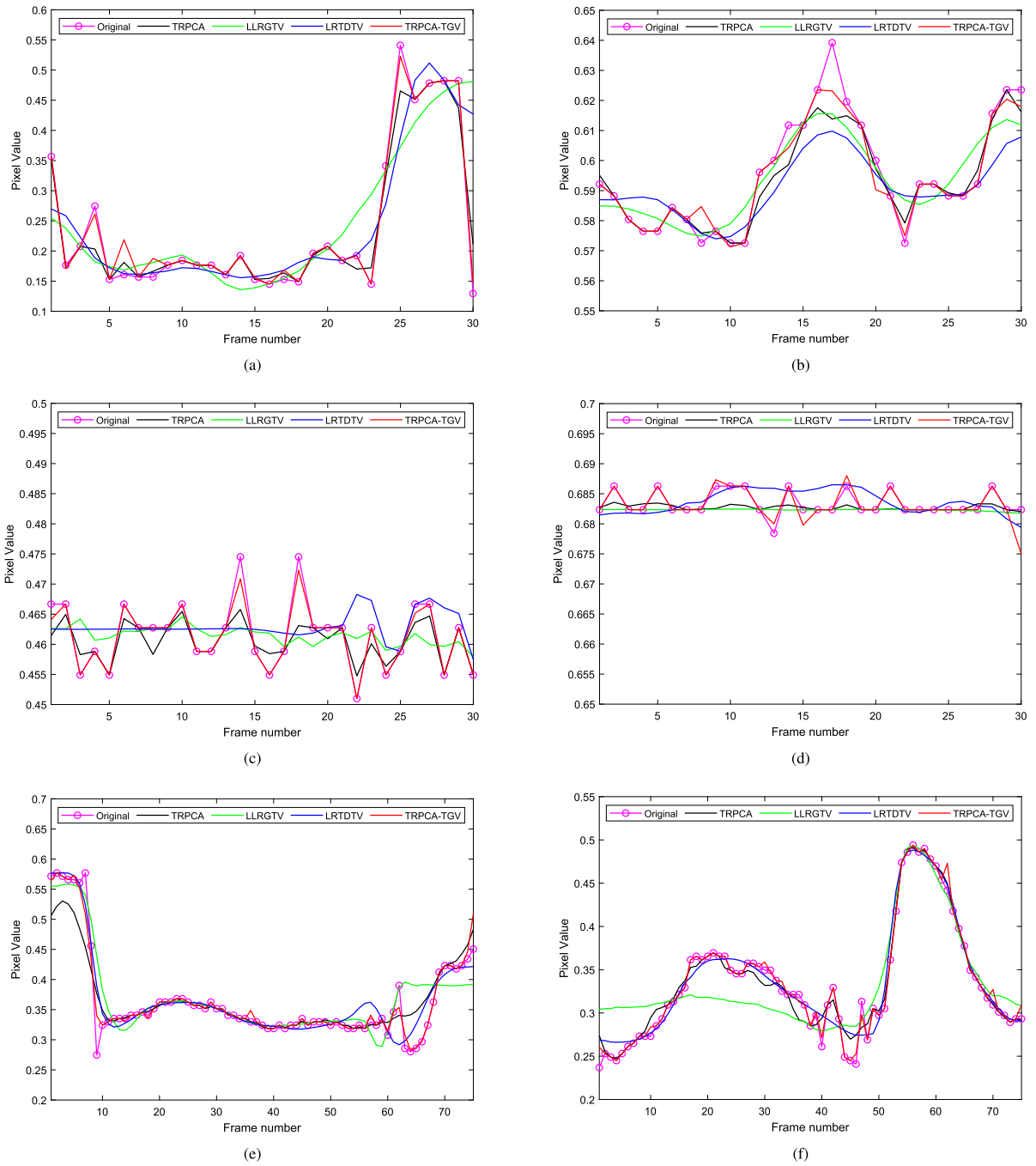

Fig. 6. The pixel values of randomly selected tubes of the recovered videos by different methods with noise level 10%. (a) *bus*, (b) *coastguard*, (c) *hall*, (d) *news*, (e) *salesman*, and (f) *suzie*.

Acknowledgements

The research is supported by Scientific and Technological Innovation Programs of Higher Education Institutions in Shanxi (2023L241), the special fund for Science and Technology Innovation Teams of Shanxi Province (202204051002018), Basic Research Program of Shanxi Province (Free exploration) project (202103021224303), Natural Science Foundation of Sichuan Province (2024NSFSC1467), Postdoctoral Fellowship Program of CPSF (GZC20232198), Fundamental Research Funds for the Central Universities (2682024CX017), and the China Postdoctoral Science Foundation (2024M752661).

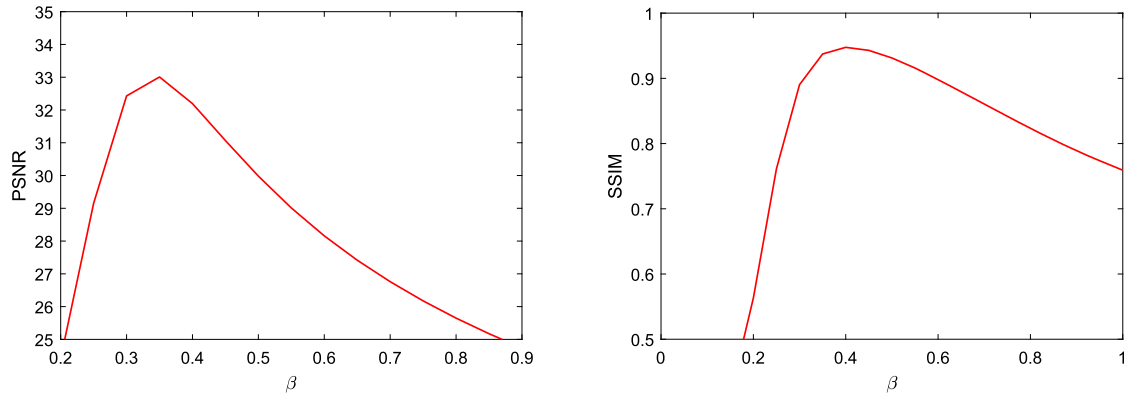

Fig. 7. The PSNR and SSIM values with respect to parameter β on color image *lena* with noise level 10%.

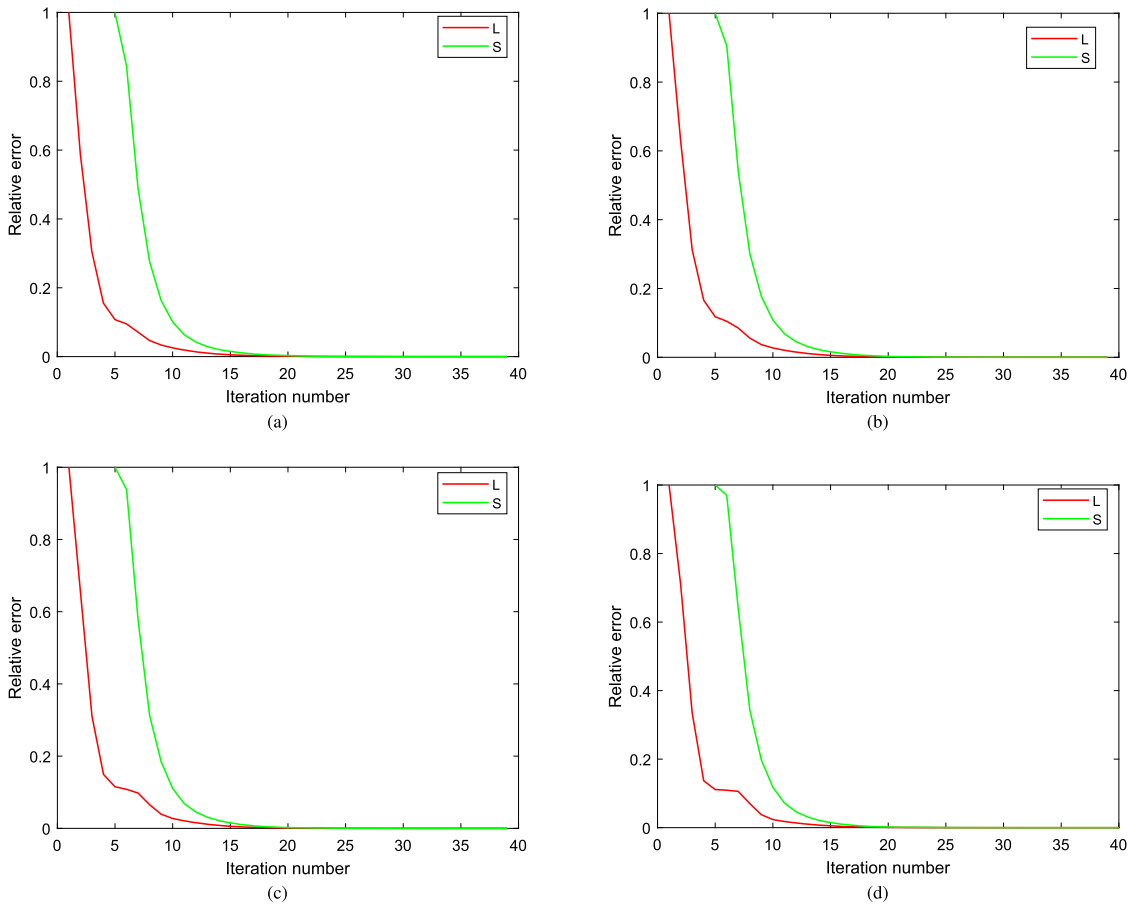

Fig. 8. The relative error curves of our method on color image *peppers* with different noise levels: (a) 10%, (b) 20%, (c) 30%, and (d) 40%.

References

[1] N.D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E.E. Papalexakis, C. Faloutsos, Tensor decomposition for signal processing and machine learning, IEEE Trans. Signal Process. 65 (13) (2017) 3551–3582.

[2] H. Chen, F. Ahmad, S. Vorobyov, F. Porikli, Tensor decompositions in wireless communications and mimo radar, IEEE J. Sel. Top. Signal Process. 15 (3) (2021) 438–453.

[3] X.-L. Zhao, W.-H. Xu, T.-X. Jiang, Y. Wang, M.K. Ng, Deep plug-and-play prior for low-rank tensor completion, Neurocomputing 400 (2020) 137–149.

[4] T.-X. Jiang, M.K. Ng, X.-L. Zhao, T.-Z. Huang, Framelet representation of tensor nuclear norm for third-order tensor completion, IEEE Trans. Image Process. 29 (2020) 7233–7244.

[5] X.-L. Zhao, J.-H. Yang, T.-H. Ma, T.-X. Jiang, M.K. Ng, T.-Z. Huang, Tensor completion via complementary global, local, and nonlocal priors, IEEE Trans. Image Process. 31 (2021) 984–999.
[6] S. Zhang, L. Han, Robust tensor recovery with nonconvex and nonsmooth regularization, Appl. Math. Comput. 438 (2023) 127566.
[7] J. Lin, T.-Z. Huang, X.-L. Zhao, T.-X. Jiang, L. Zhuang, A tensor subspace representation-based method for hyperspectral image denoising, IEEE Trans. Geosci. Remote Sens. 59 (9) (2020) 7739–7757.
[8] J. Xue, Y. Zhao, W. Liao, J.C.-W. Chan, Nonlocal low-rank regularized tensor decomposition for hyperspectral image denoising, IEEE Trans. Geosci. Remote Sens. 57 (7) (2019) 5174–5189.
[9] M. Ding, X. Fu, T.-Z. Huang, J. Wang, X.-L. Zhao, Hyperspectral super-resolution via interpretable block-term tensor modeling, IEEE J. Sel. Top. Signal Process. 15 (3) (2020) 641–656.
[10] J.-H. Yang, C. Chen, H.-N. Dai, M. Ding, Z.-B. Wu, Z. Zheng, Robust corrupted data recovery and clustering via generalized transformed tensor low-rank representation, IEEE Trans. Neural Netw. Learn. Syst. 35 (7) (2024) 8839–8853.
[11] H. Abdi, L.J. Williams, Principal component analysis, Wiley Interdiscip. Rev.: Comput. Stat. 2 (4) (2010) 433–459.
[12] E.J. Candès, X. Li, Y. Ma, J. Wright, Robust principal component analysis?, J. ACM 58 (3) (2011) 1–37.
[13] Y. Chen, T.Z. Huang, X.L. Zhao, Destriping of multispectral remote sensing image using low-rank tensor decomposition, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 11 (12) (2018) 4950–4967.
[14] S. Liao, S. Fu, Y. Li, H. Han, Image inpainting using non-convex low rank decomposition and multidirectional search, Appl. Math. Comput. 452 (2023) 128048.
[15] J. Wright, A. Ganesh, S. Rao, Y. Peng, Y. Ma, Robust principal component analysis: exact recovery of corrupted low-rank matrices via convex optimization, Adv. Neural Inf. Process. Syst. 22 (2009) 2080–2088.
[16] Y. Peng, A. Ganesh, J. Wright, W. Xu, Y. Ma, Rasl: robust alignment by sparse and low-rank decomposition for linearly correlated images, IEEE Trans. Pattern Anal. Mach. Intell. 34 (11) (2012) 2233–2246.
[17] Y. Liu, L. Chen, C. Zhu, Improved robust tensor principal component analysis via low-rank core matrix, IEEE J. Sel. Top. Signal Process. 12 (6) (2018) 1378–1389.
[18] W. Cao, Y. Wang, J. Sun, D. Meng, C. Yang, A. Cichocki, Z. Xu, Total variation regularized tensor RPCA for background subtraction from compressive measurements, IEEE Trans. Image Process. 25 (9) (2016) 4075–4090.
[19] J.-H. Yang, X.-L. Zhao, T.-Y. Ji, T.-H. Ma, T.-Z. Huang, Low-rank tensor train for tensor robust principal component analysis, Appl. Math. Comput. 367 (2020) 124783.
[20] C. Battaglino, G. Ballard, T.G. Kolda, A practical randomized CP tensor decomposition, SIAM J. Matrix Anal. Appl. 39 (2) (2018) 876–901.
[21] Y. Xu, Z. Wu, J. Chanussot, P. Comon, Z. Wei, Nonlocal coupled tensor CP decomposition for hyperspectral and multispectral image fusion, IEEE Trans. Geosci. Remote Sens. 58 (1) (2019) 348–362.
[22] C.J. Hillar, L.-H. Lim, Most tensor problems are NP-hard, J. ACM 60 (6) (2013) 1–39.
[23] M. Zhou, Y. Liu, Z. Long, L. Chen, C. Zhu, Tensor rank learning in CP decomposition via convolutional neural network, Signal Process. Image Commun. 73 (2019) 12–21.
[24] J. Liu, P. Musialski, P. Wonka, J. Ye, Tensor completion for estimating missing values in visual data, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2012) 208–220.
[25] B. Huang, C. Mu, D. Goldfarb, J. Wright, Provable models for robust low-rank tensor completion, Pac. J. Optim. 11 (2) (2015) 339–364.
[26] K. Braman, Third-order tensors as linear operators on a space of matrices, Linear Algebra Appl. 433 (7) (2010) 1241–1253.
[27] Z. Zhang, G. Ely, S. Aeron, N. Hao, M. Kilmer, Novel methods for multilinear data completion and de-noising based on tensor-SVD, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 3842–3849.
[28] X.-L. Zhao, H. Zhang, T.-X. Jiang, M.K. Ng, X.-J. Zhang, Fast algorithm with theoretical guarantees for constrained low-tubal-rank tensor recovery in hyperspectral images denoising, Neurocomputing 413 (2020) 397–409.
[29] R. Dian, S. Li, Hyperspectral image super-resolution via subspace-based low tensor multi-rank regularization, IEEE Trans. Image Process. 28 (10) (2019) 5135–5146.
[30] B.-Z. Li, X.-L. Zhao, T.-Y. Ji, X.-J. Zhang, T.-Z. Huang, Nonlinear transform induced tensor nuclear norm for tensor completion, J. Sci. Comput. 92 (3).
[31] W. Hu, D. Tao, W. Zhang, Y. Xie, Y. Yang, The twist tensor nuclear norm for video completion, IEEE Trans. Neural Netw. Learn. Syst. 28 (12) (2016) 2961–2973.
[32] C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, S. Yan, Tensor robust principal component analysis: exact recovery of corrupted low-rank tensors via convex optimization, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 5249–5257.
[33] C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, S. Yan, Tensor robust principal component analysis with a new tensor nuclear norm, IEEE Trans. Pattern Anal. Mach. Intell. 42 (4) (2019) 925–938.
[34] I.V. Oseledets, Tensor-train decomposition, SIAM J. Sci. Comput. 33 (5) (2011) 2295–2317.
[35] Q. Zhao, G. Zhou, S. Xie, L. Zhang, A. Cichocki, Tensor ring decomposition, arXiv preprint, arXiv:1606.05535.
[36] Y.-B. Zheng, T.-Z. Huang, X.-L. Zhao, Q. Zhao, T.-X. Jiang, Fully-connected tensor network decomposition and its application to higher-order tensor completion, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, 2021, pp. 11071–11078.
[37] Y.-B. Zheng, T.-Z. Huang, X.-L. Zhao, Q. Zhao, Tensor completion via fully-connected tensor network decomposition with regularized factors, J. Sci. Comput. 92 (1) (2022) 8.
[38] J.-H. Yang, X.-L. Zhao, T.-H. Ma, M. Ding, T.-Z. Huang, Tensor train rank minimization with hybrid smoothness regularization for visual data recovery, Appl. Math. Model. 81 (2020) 711–726.
[39] X. Xu, R. Gao, Y. Qing, J. Feng, Z. Zeng, M. Wang, Hyperspectral image mixed noise removal via tensor robust principal component analysis with tensor-ring decomposition, Int. J. Remote Sens. 44 (5) (2023) 1556–1578.
[40] Y.-Y. Liu, X.-L. Zhao, G.-J. Song, M.K. Ng, Y.-B. Zheng, T.-Z. Huang, Fully-connected tensor network decomposition for robust tensor completion problem, Inverse Probl. Imaging 18 (1) (2024) 208–238.
[41] L.I. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, Phys. D: Nonlinear Phenom. 60 (1–4) (1992) 259–268.
[42] H.A. Aly, E. Dubois, Image up-sampling using total-variation regularization with a new observation model, IEEE Trans. Image Process. 14 (10) (2005) 1647–1659.
[43] M.K. Ng, H. Shen, E.Y. Lam, L. Zhang, A total variation regularization based super-resolution reconstruction algorithm for digital video, EURASIP J. Adv. Signal Process. 2007 (2007) 1–16.
[44] W. He, H. Zhang, L. Zhang, H. Shen, Total-variation-regularized low-rank matrix factorization for hyperspectral image restoration, IEEE Trans. Geosci. Remote Sens. 54 (1) (2015) 178–188.
[45] Z. Wu, Q. Wang, Z. Wu, Y. Shen, Total variation-regularized weighted nuclear norm minimization for hyperspectral image mixed denoising, J. Electron. Imaging 25 (1) (2016) 013037.
[46] H.K. Aggarwal, A. Majumdar, Hyperspectral image denoising using spatio-spectral total variation, IEEE Geosci. Remote Sens. Lett. 13 (3) (2016) 442–446.
[47] Y. Wang, J. Peng, Q. Zhao, Y. Leung, X.-L. Zhao, D. Meng, Hyperspectral image restoration via total variation regularized low-rank tensor decomposition, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 11 (4) (2017) 1227–1243.
[48] Y. Chen, S. Wang, Y. Zhou, Tensor nuclear norm-based low-rank approximation with total variation regularization, IEEE J. Sel. Top. Signal Process. 12 (6) (2018) 1364–1377.
[49] W. He, H. Zhang, H. Shen, L. Zhang, Hyperspectral image denoising using local low-rank matrix recovery and global spatial–spectral total variation, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 11 (3) (2018) 713–729.

[50] K. Bredies, K. Kunisch, T. Pock, Total generalized variation, SIAM J. Imaging Sci. 3 (3) (2010) 492–526.
[51] W. Guo, J. Qin, W. Yin, A new detail-preserving regularization scheme, SIAM J. Imaging Sci. 7 (2) (2014) 1309–1334.
[52] S. Wang, W. Guo, T.-Z. Huang, Weighted total generalized variation for compressive sensing reconstruction, in: 2015 International Conference on Sampling Theory and Applications (SampTA), IEEE, 2015, pp. 244–248.
[53] W. Feng, H. Lei, Y. Gao, Speckle reduction via higher order total variation approach, IEEE Trans. Image Process. 23 (4) (2014) 1831–1843.
[54] K. Bredies, Y. Dong, M. Hintermüller, Spatially dependent regularization parameter selection in total generalized variation models for image restoration, Int. J. Comput. Math. 90 (1) (2013) 109–123.
[55] T.-H. Ma, T.-Z. Huang, X.-L. Zhao, Spatially dependent regularization parameter selection for total generalized variation-based image denoising, Comput. Appl. Math. 37 (2018) 277–296.
[56] M.-G. Shama, T.-Z. Huang, J. Liu, S. Wang, A convex total generalized variation regularized model for multiplicative noise and blur removal, Appl. Math. Comput. 276 (2016) 109–121.
[57] F. Knoll, K. Hammernik, C. Zhang, S. Moeller, T. Pock, D.K. Sodickson, M. Akcakaya, Deep-learning methods for parallel magnetic resonance imaging reconstruction: a survey of the current approaches, trends, and issues, IEEE Signal Process. Mag. 37 (1) (2020) 128–140.
[58] B. He, M. Tao, X. Yuan, Alternating direction method with Gaussian back substitution for separable convex programming, SIAM J. Optim. 22 (2) (2012) 313–340.
[59] Florian Knoll, Kristian Bredies, Thomas Pock, Rudolf Stollberger, Second order total generalized variation (TGV) for MRI, Magn. Reson. Med. 65 (2) (2010) 480–491.
[60] I. Loris, C. Verhoeven, On a generalization of the iterative soft-thresholding algorithm for the case of non-separable penalty, Inverse Probl. 27 (12) (2011) 125007.
[61] J.J. Mei, T.Z. Huang, S. Wang, X.L. Zhao, Second order total generalized variation for speckle reduction in ultrasound images, J. Franklin Inst. (2017) 1–26.
[62] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Trans. Image Process. 13 (4) (2004) 600–612.