Applied Mathematical Modelling 150 (2026) 116368

Contents lists available at ScienceDirect

Applied Mathematical Modelling

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

Tensor completion via total curvature variation and low-rank matrix factorization

Zhi Xu [a,b], Jing-Hua Yang [c,*], Xi-Le Zhao [d], Xi-hong Yan [a,b], Chuan-long Wang [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, Chengdu, Sichuan, 611756, PR China
[d] School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu, Sichuan, 611731, PR China

ARTICLE INFO

Keywords:
Low-rank matrix factorization
Total curvature variation
Low-rank tensor completion
Proximal alternating minimization algorithm

ABSTRACT

Curvature-based regularization has attracted growing concern in the field of image restoration, benefiting from its favorable geometric properties, such as preserving sharp edges, corners and contrast. Total variation regularization has the ability to promote piecewise smooth property and preserve edges in image processing. Inspired by the advantages of curvature regularization and total variation, in the paper, we first develop a regularization that combines curvature and total variation to explore the geometric characteristics inside high-dimensional data, called total curvature variation (TCV) regularization, which can better preserve local information of the underlying data. We present a new low-rank tensor completion model via TCV and low-rank matrix factorization, which can simultaneously exploits the global low-rank prior and local structure information of data. We solve the proposed minimization problem by using the effective proximal alternating minimization algorithm with guaranteed convergence. Results from experiments on color images, videos, and magnetic resonance images show the superior performance of the proposed method over the compared methods in terms of quantitative and qualitative evaluations.

1. Introduction

Being the higher-dimensional extension of vector and matrix, tensor is an important data form which can express more underlying complex structure inside the data. Tensor can be used as a powerful tool in the field of high-dimensional image processing, such as color image and video completion [1,2], magnetic resonance image (MRI) recovery [3,4], hyperspectral image (HSI) recovery [5,6], traffic measurement data recovery [7], and data clustering [8,9]. In this work, we concentrate on tensor completion problem.

Tensor completion is a crucial procedure to recovery a clean image from the incomplete observation. Low-rank constraint is the most common approach in low-rank tensor completion (LRTC) problem. LRTC can be taken as the generalization of low-rank matrix completion [10]. Mathematically, the LRTC problem can be modeled as:

* Corresponding author.
E-mail addresses: xuzhizhenyuan@163.com (Z. Xu), yangjinghua110@126.com (J.-H. Yang), xlzhao122003@163.com (X.-L. Zhao), xihong1@e.ntu.edu.sg (X.-h. Yan), clwang1964@163.com (C.-l. Wang).

https://doi.org/10.1016/j.apm.2025.116368
Received 27 February 2025; Received in revised form 25 June 2025; Accepted 7 August 2025

Available online 19 August 2025

$$\begin{aligned} \min_{\mathcal{Y}} \quad & \operatorname{rank}(\mathcal{Y}) \\ \text{s.t.} \quad & \mathcal{P}_{\Omega}(\mathcal{Y}) = \mathcal{O}, \end{aligned} \tag{1}$$

where $\mathcal{Y} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ is the underlying tensor, $\mathcal{O} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ denotes the initial incomplete tensor, Ω is the set in which the entries are known, $\mathcal{P}_{\Omega}$ represents a linear projection operator, it makes the elements of tensor $\mathcal{Y}$ in Ω equal to $\mathcal{O}$. So the key problem of LRTC is to solve the measure of tensor rank. Unlike the rank of matrix, there exist no uniqueness for the rank of tensor. The typical tensor ranks are CANDECOMP/PARAFAC (CP) rank [11] and Tucker rank [12]. In past few decades, some new low-rank approximation characterizations for tensor are proposed, like tubal rank [13], tensor train (TT) rank [14], tensor ring (TR) rank [15], and fully-connected tensor network (FCTN) rank [16,17]. Nevertheless, the minimization of tensor rank is NP-hard [18] because of the non-convexity and non-continuity of the rank. To cope with this question, Liu et al. [19] developed the sum of nuclear norm of the unfolding matrices as the convex surrogate of Tucker rank, and come up with the following LRTC approach:

$$\begin{aligned} \min_{\mathcal{Y}} \quad & \sum_{k=1}^{N} \alpha_k \|\mathbf{Y}_{(k)}\|_* \\ \text{s.t.} \quad & \mathcal{P}_{\Omega}(\mathcal{Y}) = \mathcal{O}, \end{aligned} \tag{2}$$

where $\|\cdot\|_*$ represents the nuclear norm, $\mathbf{Y}_{(k)} \in \mathbb{R}^{I_k \times \prod_{n \neq k} I_n}$ is the mode-k unfolding of tensor $\mathcal{Y}$, and each $\alpha_k \geq 0$, $\sum_{k=1}^{N} \alpha_k = 1$. In order to search for the efficient solution of solving problem (2), Liu et al. [19] proposed the high accuracy low rank tensor completion algorithm (HaLRTC), which is efficient to obtain the high accuracy solution. Over the last decade, numerous tensor completion methods based on (2) are put forward. Ji et al. [20] proposed a kernel LRTC method via mapping $\mathbf{Y}_{(k)}$ to a higher-order kernel space and then minimizing the sum of the Schatten-p quasi-norm [21]. However, such models need to calculate the singular value decomposition (SVD) of all $\mathbf{Y}_{(k)}$ during each iteration, with the result that the computation complexity increases. To address this matter, Xu et al. [22] executed low-rank matrix parallel factorization for each $\mathbf{Y}_{(k)}$, named TMac, which can be formulated as:

$$\begin{aligned} \min_{\mathbf{A}_k, \mathbf{X}_k, \mathcal{Y}} \quad & \sum_{k=1}^{N} \frac{\alpha_k}{2} \|\mathbf{Y}_{(k)} - \mathbf{A}_k \mathbf{X}_k\|_F^2 \\ \text{s.t.} \quad & \mathcal{P}_{\Omega}(\mathcal{Y}) = \mathcal{O}, \end{aligned} \tag{3}$$

where $\mathbf{A}_k \in \mathbb{R}^{I_k \times r_k}$, $\mathbf{X}_k \in \mathbb{R}^{r_k \times \prod_{n \neq k} I_n}$, and r_k presents the estimated rank of the matrix $\mathbf{Y}_{(k)}$. TMac has shown its better performance in the efficiency of computing. Note that models (2) and (3) only consider the global low-rank prior of tensor. However, this may cause some unsatisfactory results like losing important image details when facing high-dimensional image with complex structures or extremely low sampling rate.

An important but often overlooked fact is that most high-dimensional images are characteristic of piecewise smoothness. As one of the best-known regularization to characterize the piecewise smooth property, total variation (TV) has the effects of preserving edges and promoting local piecewise smoothness in image processing [23,24]. A lot of researches based on TV have been done for different image processing problems [25,26]. Zhao et al. [27] introduced TV regularization into sparse unmixing for image restoration. Inspired by the work [27], Ji et al. [28] applied TV to $\mathbf{X}_3$ in optimization problem (3), and presented a low-rank matrix factorization-based project (MF-TV), it can be given as follows:

$$\begin{aligned} \min_{\mathbf{A}_k, \mathbf{X}_k, \mathcal{Y}} \quad & \sum_{k=1}^{N} \frac{\alpha_k}{2} \|\mathbf{Y}_{(k)} - \mathbf{A}_k \mathbf{X}_k\|_F^2 + \mu \mathrm{TV}(\mathbf{X}_3) \\ \text{s.t.} \quad & \mathcal{P}_{\Omega}(\mathcal{Y}) = \mathcal{O}, \end{aligned} \tag{4}$$

where μ is a positive regularization parameter. MF-TV can get a better completion result compared to TMac. Li et al. [29] incorporated TV into the unfolding matrices along each mode for LRTC problem (LRTC-TV). Yokota et al. [30] developed a convex programming for tensor completion and denoising simultaneously which devotes to minimize the objective function consisting of tensor nuclear norm and TV in the condition of noise inequality. Chen et al. [31] proposed an asymmetric three-dimensional TV and integrated it into framelet regularized low-rank tensor completion.

Although TV regularization has been proved to have the ability of preserving sharp edges in image restoration, it also suffers from some undesirable effects, such as the staircase effect and losing image contrast [32]. To overcome these drawbacks, some curvature-based methods have been introduced for two-dimensional grayscale image recovery [33,34]. As a significant geometric concept, curvature can characterize the degree to which a curve or a surface deviates from a straight line or a flat plan. The curvature regularization is outstanding in image processing field due to its fine geometric interpretability and strong prior in the continuity, and it has the ability to preserve sharp edges, corners, and contrast of image [35]. Zhu and Chan [36] introduced the L_1 norm of mean curvature [37] of the image surface to recover noisy images, and got better edges and corners than the classical Rudin-Osher-Fatemi model [23]. Brito-Loeza et al. [38] considered the L_1 norm of Gaussian curvature as the regularization term in image restoration problem, and verified the capability of curvature regularization to preserve sharp edges and contrast of image. Dong et al. [39] incorporated the low dimensional manifold regularization with a higher-order curvature regularization in grayscale image completion. Zhong et al. [40] minimized the normal curvatures along different directions for image completion. Nevertheless, almost all curvature-based methods are used in two-dimensional image processing.

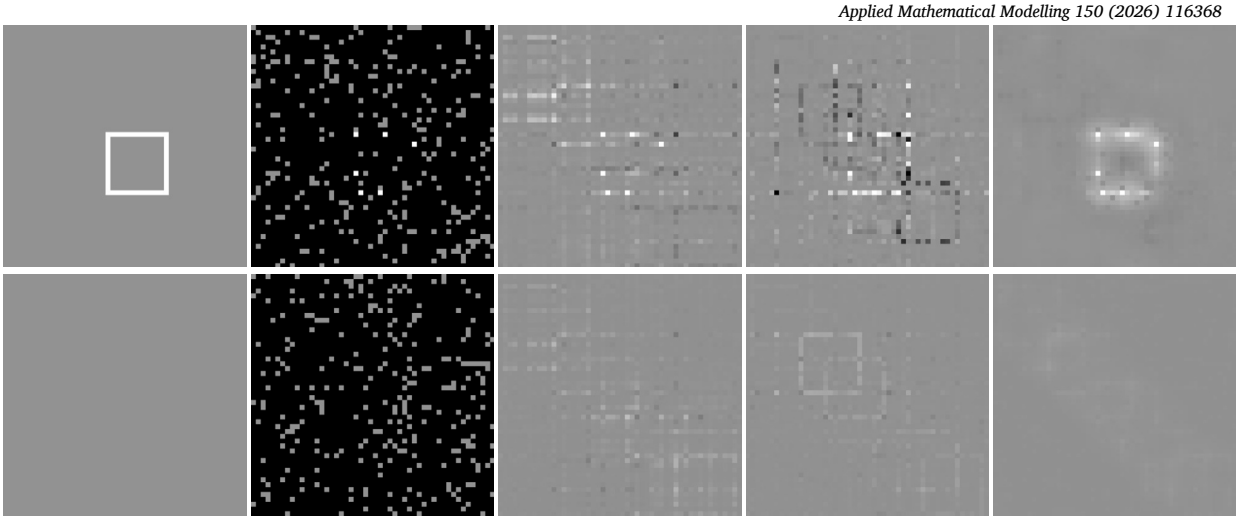

Fig. 1. The results recovered by different methods. The first and second rows respectively show the 24-th and 46-th slices of the recovered synthetic $50 \times 50 \times 50$ data with 90% entries missing. From left to right: original data, observed data, the recovered results by TMac [22], MF-TV [28], and the proposed method TCV-LRTC, respectively.

To preserve the geometric properties of high-dimensional image and avoid the staircase effect in smooth region, we make the corresponding curvature at a point as the weight of TV, and develop the total curvature variation (TCV) regularization for tensor completion. Then we present a new LRTC model based on low-rank matrix factorization and TCV (TCV-LRTC),

$$\begin{aligned} \min_{\mathbf{A}_k, \mathbf{X}_k, \mathcal{Y}} \quad & \sum_{k=1}^{N} \frac{\alpha_k}{2} \|\mathbf{Y}_{(k)} - \mathbf{A}_k \mathbf{X}_k\|_F^2 + \lambda \langle \phi(\kappa), |\nabla \mathbf{Y}_{(1)}| \rangle \\ \text{s.t.} \quad & \mathcal{P}_\Omega(\mathcal{Y}) = \mathcal{O}, \end{aligned} \tag{5}$$

where $\lambda > 0$ is the regularization parameter, $\mathbf{Y}_{(1)}$ represents the mode-1 unfolding of tensor $\mathcal{Y}$, $|\nabla \mathbf{Y}_{(1)}|$ denotes the TV of matrix $\mathbf{Y}_{(1)}$, $< \cdot, \cdot >$ denotes the inner product, $\phi(\kappa)$ is a function about the curvature κ. The difference between our model and two other methods, i.e., TMac [22] and MF-TV [28], lies on the strong prior of curvature to keep local features. Our motivation can be seen in Fig. 1. The test data is synthetic with size $50 \times 50 \times 50$, there is a white square moving across the main diagonal in the first 40 frames, its last 10 frames only have background with intensity 0.5. The recovered results exhibited in Fig. 1 are respectively the 24-th and 46-th frames of the synthetic data. Apparently, the proposed method ameliorates the restoration effects for global and local characteristics. The proposed regularization-enhanced framework achieves dual-scale recovery, simultaneously preserving global structures coherence through low-rank constraints while capturing fine-grained local details via the novel regularizer, relative to matrix factorization-based tensor completion methods [22]. Compared to TV-based approaches [28,29], our curvature-aware formulation maintains sharp edges, corners and contrast of the image and eliminates the staircase artifacts caused by TV. Furthermore, this work is the first successful extension of traditional curvature regularization [33,34,40] to high-dimensional image restoration, overcoming the inherent dimensional limitations of the two-dimensional grayscale applications.

To sum up, the main contributions of this paper consist of three aspects.

• The TCV regularization based on TV and curvature is designed to capture the geometric properties in the high-dimensional image completion problem. To the best of our knowledge, TCV is first extended to explore local details of high-dimensional image. TCV can overcome the staircase effect caused by TV, and can also preserve sharp edges, corners and contrast of the image.

• We produce a novel LRTC model by incorporating TCV into the low-rank matrix factorization, which is capable of concurrently capturing both the global low-rank properties and local detailed features of the high-dimensional image.

• The proximal alternating minimization (PAM) algorithm is employed to convert the original optimization problem with coupled variables to some solvable subproblems. The convergence of the algorithm based on PAM framework is theoretically guaranteed. Numerical experiments are performed on color images, videos and MRIs, and verify that the proposed method achieves better recovery results compared with five tensor completion methods.

The outline of this work organizes as follows. We introduce some notations and preliminaries about tensor and TCV in Section 2. Section 3 describes the TCV regularized LRTC model and then presents the solving algorithm. Section 4 gives extensive experiments and comparable results to verify the performance of the proposed method. Section 5 is the conclusion of this paper.

2. Notations and preliminaries

In this section, some basic notations for tensor and preliminaries of total curvature variation and curvature are given.

2.1. Tensor basics

In this paper, we respectively represent vector by boldface lowercase letter (e.g., $\mathbf{a}$), matrix by capital letter (e.g., $\mathbf{A}$), and tensor by calligraphic letter (e.g., $\mathcal{A}$). Given an N-order tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$, its $(i_1, i_2, \cdots, i_N)$-th entry is expressed by $a_{i_1,i_2,\cdots,i_N}$.

The inner product of two tensors $\mathcal{A}$ and $\mathcal{B}$ of the same size is denoted as

$$\langle \mathcal{A}, \mathcal{B} \rangle = \sum_{i_1,i_2,\cdots,i_N} a_{i_1,i_2,\cdots,i_N} b_{i_1,i_2,\cdots,i_N}.$$

The Frobenius norm of $\mathcal{A}$ is $\|\mathcal{A}\|_F = \sqrt{\langle \mathcal{A}, \mathcal{A} \rangle}$.

For a N-order tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$, the mode-$k$ unfolding of $\mathcal{A}$ is expressed as $\mathbf{A}_{(k)} \in \mathbb{R}^{I_k \times \prod_{n \neq k} I_n}$, and its (i_k, j)-th entry equals to the $(i_1, i_2, \cdots, i_N)$-th entry of tensor $\mathcal{A}$, where

$$j = 1 + \sum_{n=1, n \neq k}^{N} (i_n - 1) J_n \quad \text{with} \quad J_n = \prod_{m=1, m \neq k}^{n-1} I_m.$$

The inverse operator of unfolding is expressed by "fold", i.e., $\mathcal{A} = \text{fold}_{(k)}(\mathbf{A}_{(k)})$. Besides, $\|\mathcal{A}\|_F = \|\mathbf{A}_{(k)}\|_F$ for $k = 1, \cdots, N$. We define the Tucker rank of tensor $\mathcal{A}$ as an array:

$$\text{rank}(\mathcal{A}) = (\text{rank}(\mathbf{A}_{(1)}), \text{rank}(\mathbf{A}_{(2)}), \cdots, \text{rank}(\mathbf{A}_{(N)})). \tag{6}$$

2.2. Total curvature variation (TCV) regularization

This subsection concretely presents the discrete formulation of TCV regularization. Let u be a two-dimensional image, $u(i, j)$ is the gray value at point (i, j). In this paper, we develop the regularization based on curvature [40] and TV to capture the local information, named total curvature variation (TCV) regularization, which can be formulated as following form:

$$R(u) = \langle \phi(\kappa), |\nabla u| \rangle, \tag{7}$$

where $|\nabla u|$ is the isotropic TV of u, $\phi(\kappa)$ is a suitable function of curvature κ of the point (i, j). Following [40], in this work, $\phi(\kappa) = 1 + \omega|\kappa|$ is selected. For an $m \times n$ image, the discrete gradient operator at the (i, j)-th pixel is defined as:

$$(\nabla u)_{i,j} = ((\mathbf{D}_1 u)_{i,j}, (\mathbf{D}_2 u)_{i,j}), \quad i = 1, \cdots, m, j = 1, \cdots, n, \tag{8}$$

where $\mathbf{D}_1$ and $\mathbf{D}_2$ are the first-order difference operators in horizontal and vertical directions.

2.3. Discrete calculation of curvature

In this subsection, we present herein a discretization scheme for curvature estimation. For a two-dimensional image $u : \Theta \to \mathbb{R}, \Theta \subset \mathbb{R}^2$, assume that $\mathcal{S} : (x, u(x))$ denotes the regular surface in $\mathbb{R}^3$ for $x \in \Theta$. The normal curvature at a point can be represented as the ratio of the second fundamental form II to the first fundamental form I:

$$\kappa_n = \frac{\text{II}}{\text{I}}, \tag{9}$$

where $\text{I} = ds^2$ denotes the square of the arc-length between center point and its neighbor point, II can be approximated by the following proposition.

Proposition 1. *[40] Assuming that $\mathcal{S} : (x, u(x))$ denotes an image surface, $O(i, j, u_{i,j})$ is any point on $\mathcal{S}$, the second fundamental form* II *at O can be approximately calculated as:*

$$\text{II} \approx 2d_l, \tag{10}$$

where d_l is the distance to the tangent plane of O from its neighbor point $P(i + \triangle i, j + \triangle j, u_{i+\triangle i, j+\triangle j})$, $\triangle i$ and $\triangle j$ respectively express the increments for variables i and j.

To simplify the symbol for computing normal curvature, we present the image surface in $\mathbb{R}^3$ with function $z = u_{i,j}, (i, j) \in \Theta$. The domain Θ of size $m \times n$ can be expressed by the grid $\Theta = \{(i, j) : 1 \leq i \leq m, 1 \leq j \leq n\}$. The normal curvature at the central point O can be calculated numerically though a 3×3 patch in discrete form, see Fig. 2(a), where $\triangle$ denotes a half point. Take the calculation of normal curvature of $O(i, j, u_{i,j})$ along the direction $\overrightarrow{OX}$ as an example, we use the plane T_{XYZ} spanned by three points, $X(i-1, j, u_{i-1,j})$, $Y(i, j-1, u_{i,j-1})$ and $Z(i, j+1, u_{i,j+1})$, to estimate the tangent plane of O. The normal vector $\mathbf{V}$ of plane T_{XYZ} can be obtained by computing the outer product of the vectors $\overrightarrow{XY}$ and $\overrightarrow{XZ}$,

$$\mathbf{V} = \overrightarrow{XY} \times \overrightarrow{XZ} = (2u_{i-1,j} - u_{i,j-1} - u_{i,j+1}, u_{i,j-1} - u_{i,j+1}, 2). \tag{11}$$

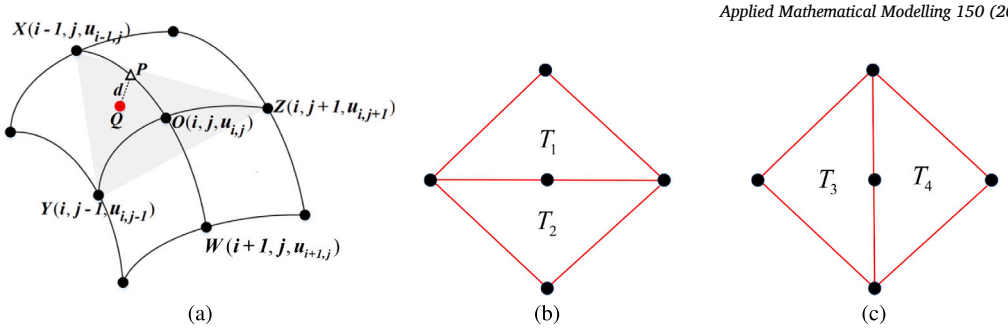

Fig. 2. Illustration of the tangent plane. (a) The distance from half point P to tangent plane T_{XYZ}, (b) tangent planes T_1 and T_2, (c) tangent planes T_3 and T_4.

Then the distance d_l in (10) can be estimated by the projection distance from half point $P(i-\frac{1}{2},j,u_{i-\frac{1}{2},j})$ to the tangent plane T_{XYZ},

$$d_l = \frac{\overrightarrow{XP}\cdot\mathbf{V}}{|\mathbf{V}|} = \frac{2u_{i,j}-u_{i,j-1}-u_{i,j+1}}{2\sqrt{(2u_{i-1,j}-u_{i,j-1}-u_{i,j+1})^2+(u_{i,j-1}-u_{i,j+1})^2+4}}, \tag{12}$$

where $u_{i-\frac{1}{2},j}$ can be approximated by the average value of its neighbor points O and X, i.e., $u_{i-\frac{1}{2},j}=\frac{u_{i-1,j}+u_{i,j}}{2}$.

According to the geometric characteristics of a surface, different direction vectors at one point correspond to different normal curvatures. We compute four normal curvatures at point O along four directions, i.e., $\overrightarrow{OX}$, $\overrightarrow{OY}$, $\overrightarrow{OZ}$ and $\overrightarrow{OW}$. As shown in Fig. 2(b) and 2(c), four triangular planes (i.e., T_1-T_4) are applied to approach the tangent planes along four directions. Then the four distances (i.e., $d_q, q=1,2,3,4$) from half points to tangent planes can be calculated as follows:

$$d_1 = \frac{2u_{i,j}-u_{i,j-1}-u_{i,j+1}}{2\sqrt{(2u_{i-1,j}-u_{i,j-1}-u_{i,j+1})^2+(u_{i,j-1}-u_{i,j+1})^2+4}};$$

$$d_2 = \frac{u_{i,j-1}+u_{i,j+1}-2u_{i,j}}{2\sqrt{(2u_{i+1,j}-u_{i,j-1}-u_{i,j+1})^2+(u_{i,j+1}-u_{i,j-1})^2+4}};$$

$$d_3 = \frac{u_{i-1,j}-u_{i+1,j}-2u_{i,j}}{2\sqrt{(u_{i+1,j}-u_{i-1,j})^2+(u_{i-1,j}-u_{i+1,j}-2u_{i,j-1})^2+4}};$$

$$d_4 = \frac{2u_{i,j}-u_{i-1,j}-u_{i+1,j}}{2\sqrt{(u_{i-1,j}-u_{i+1,j})^2+(u_{i-1,j}-u_{i+1,j}-2u_{i,j+1})^2+4}}.$$

The first fundamental form I (the square of the arc-length $\widehat{OP}$) can be approximately calculated using the square of straight-line distance from O to P,

$$\mathrm{I} = ds^2 = \widehat{OP}^2 \approx (u_{i-\frac{1}{2},j}-u_{i,j})^2+h^2, \tag{13}$$

where h denotes the grid step. Hence the normal curvatures at point O along four direction vectors can be denoted as:

$$\kappa_{n,q} \approx \frac{2d_q}{ds_q^2}, \quad q=1,2,3,4. \tag{14}$$

To make the best use of geometric measurements along all direction vectors, we employ the discrete total curvature at a given point $x\in\Theta$, which can be estimated by:

$$\kappa(x) \approx \sum_{q=1}^{4}|\kappa_{n,q}|. \tag{15}$$

3. The proposed model and algorithm

This section introduces the formulated LRTC model along with its corresponding algorithm with convergence guarantees.

3.1. Proposed model

For a N-order tensor $\mathcal{Y} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$, the objective function of the proposed model (5) can be reformulated as follows:

$$\min_{\mathbf{A},\mathbf{X},\mathcal{Y}} f(\mathbf{A},\mathbf{X},\mathcal{Y}) = \min_{\mathbf{A},\mathbf{X},\mathcal{Y}} \sum_{k=1}^{N} \frac{\alpha_k}{2} \|\mathbf{Y}_{(k)} - \mathbf{A}_k \mathbf{X}_k\|_F^2 + \lambda \langle \phi(\kappa(\mathbf{Y}_{(1)})), |\nabla \mathbf{Y}_{(1)}| \rangle, \tag{16}$$

where $\mathbf{A} = (\mathbf{A}_1, \cdots, \mathbf{A}_N)$, $\mathbf{X} = (\mathbf{X}_1, \cdots, \mathbf{X}_N)$, λ is the regularization parameter, $\sum_{k=1}^{N} \alpha_k = 1, \alpha_k \geq 0 (k = 1, \cdots, N)$, and $\kappa(\mathbf{Y}_{(1)})$ is the curvature of one point on the surface formed by $\mathbf{Y}_{(1)}$.

The proposed model has two parts, the first term is the low-rank matrix factorizations of all unfolding matrices $\mathbf{Y}_{(k)}$ that promotes the global low-rankness of the underlying tensor $\mathcal{Y}$. We assume that the Tucker rank of $\mathcal{Y}$ is $(r_1, \cdots, r_N)$ and $r_k < I_k$ $(k = 1, \cdots, N)$, therefore, $\mathbf{A}_k \in \mathbb{R}^{I_k \times r_k}$ and $\mathbf{X}_k \in \mathbb{R}^{r_k \times \prod_{n \neq k} I_n}$ are the low-rank matrix factorization of $\mathbf{Y}_{(k)}$.

In this work, $|\nabla \mathbf{Y}_{(1)}|$ is the isotropic TV of $\mathbf{Y}_{(1)}$. Generally speaking, the more uneven a point on the surface is, the larger curvature it will have. If a point on the surface has a larger TV value, then it must be characterized by larger curvature. Giving the same weight to the TV at all points is illogical. Hence we assign different weights to TV using the corresponding curvatures so that the staircase effects in smooth region can be avoided and the geometric properties of the image can be preserved.

Remark 1. The second term is total curvature variation (TCV) regularization whose role is keeping the geometric properties of the image, such as piece-wise smoothness, edges, corners, and contrast. Since the mode-1 unfolding $\mathbf{Y}_{(1)}$ and mode-2 unfolding $\mathbf{Y}_{(2)}$ of $\mathcal{Y}$ have similar spatial information like potential edges and texture features. To reduce the computational complexity, we only employ the regularization about $\mathbf{Y}_{(1)}$ to preserve spatial local information.

3.2. Proposed algorithm

In this subsection, the proximal alternating minimization (PAM) algorithm [41] is used to solve the proposed model.

Obviously, the objective function (16) is not jointly convex for $(\mathbf{A}, \mathbf{X}, \mathcal{Y})$. Let $\mathcal{Z} = (\mathbf{A}, \mathbf{X}, \mathcal{Y})$, $\mathcal{Z}^l = (\mathbf{A}^l, \mathbf{X}^l, \mathcal{Y}^l)$, by utilizing the proximal operator under the PAM algorithmic framework, (16) can be approximated by the following expression:

$$\min_{\mathcal{Z}} h(\mathcal{Z}, \mathcal{Z}^l) = \min_{\mathcal{Z}} f(\mathcal{Z}) + \frac{\gamma}{2} \|\mathcal{Z} - \mathcal{Z}^l\|_F^2, \tag{17}$$

where $\gamma > 0$ is the proximal parameter. From (17), the subproblems for $\mathbf{A}$, $\mathbf{X}$ and $\mathcal{Y}$ can be calculated respectively by:

$$\begin{cases} \mathbf{A}^{l+1} = \arg\min_{\mathbf{A}} f(\mathbf{A}, \mathbf{X}^l, \mathcal{Y}^l) + \frac{\gamma}{2} \|\mathbf{A} - \mathbf{A}^l\|_F^2, \\ \mathbf{X}^{l+1} = \arg\min_{\mathbf{X}} f(\mathbf{A}^{l+1}, \mathbf{X}, \mathcal{Y}^l) + \frac{\gamma}{2} \|\mathbf{X} - \mathbf{X}^l\|_F^2, \\ \mathcal{Y}^{l+1} = \arg\min_{\mathcal{Y}} f(\mathbf{A}^{l+1}, \mathbf{X}^{l+1}, \mathcal{Y}) + \frac{\gamma}{2} \|\mathcal{Y} - \mathcal{Y}^l\|_F^2. \end{cases} \tag{18}$$

Since $\mathbf{A}$-subproblem and $\mathbf{X}$-subproblem can be respectively regarded as N independent subproblems, we can solve them easily as follows:

$$\begin{aligned} \mathbf{A}_k^{l+1} &= \arg\min_{\mathbf{A}_k} \frac{\alpha_k}{2} \|\mathbf{Y}_{(k)}^l - \mathbf{A}_k \mathbf{X}_k^l\|_F^2 + \frac{\gamma}{2} \|\mathbf{A}_k - \mathbf{A}_k^l\|_F^2 \\ &= (\alpha_k \mathbf{Y}_{(k)}^l (\mathbf{X}_k^l)^T + \gamma \mathbf{A}_k^l)(\alpha_k \mathbf{X}_k^l (\mathbf{X}_k^l)^T + \gamma \mathbf{I}_1)^\dagger, \end{aligned} \tag{19}$$

and

$$\begin{aligned} \mathbf{X}_k^{l+1} &= \arg\min_{\mathbf{X}_k} \frac{\alpha_k}{2} \|\mathbf{Y}_{(k)}^l - \mathbf{A}_k^{l+1} \mathbf{X}_k\|_F^2 + \frac{\gamma}{2} \|\mathbf{X}_k - \mathbf{X}_k^l\|_F^2 \\ &= (\alpha_k (\mathbf{A}_k^{l+1})^T \mathbf{A}_k^{l+1} + \gamma \mathbf{I}_2)^\dagger (\alpha_k (\mathbf{A}_k^{l+1})^T \mathbf{Y}_{(k)}^l + \gamma \mathbf{X}_k^l), \end{aligned} \tag{20}$$

where $\mathbf{I}_1 \in \mathbb{R}^{r_k \times r_k}$ and $\mathbf{I}_2 \in \mathbb{R}^{r_k \times r_k}$ are identity matrices, and $(\cdot)^\dagger$ denotes the Moore-Penrose pseudoinverse of matrix $(\cdot)$. The complexity of computing $\mathbf{A}_k$ and $\mathbf{X}_k$ is $O(I_k r_k^2 + I_k r_k s_k + r_k^2 s_k)$ for $k = 1, \cdots, N$, where $s_k = \prod_{n=1, n \neq k}^{N} I_n$.

The $\mathcal{Y}$- subproblem can be expressed as:

$$\mathcal{Y}^{l+1} = \arg\min_{\mathcal{P}_\Omega(\mathcal{Y}) = \mathcal{O}} \sum_{k=1}^{N} \frac{\alpha_k}{2} \|\mathbf{Y}_{(k)} - \mathbf{A}_k^{l+1} \mathbf{X}_k^{l+1}\|_F^2 + \frac{\gamma}{2} \|\mathcal{Y} - \mathcal{Y}^l\|_F^2 + \lambda \langle \phi(\kappa(\mathbf{Y}_{(1)})), |\nabla \mathbf{Y}_{(1)}| \rangle. \tag{21}$$

Then we transform solving the problem of tensor $\mathcal{Y}$ into the effective solution method for unfolding matrices $\mathbf{Y}_{(k)}^{l+1}$, $k = 1, \cdots, N$. We have:

$$\mathbf{Y}_{(k)}^{l+1} = \frac{\alpha_k \mathbf{A}_k^{l+1} \mathbf{X}_k^{l+1} + \gamma \mathbf{Y}_{(k)}^l}{\alpha_k + \gamma}, \quad k = 2, \cdots, N. \tag{22}$$

The complexity of computing $\mathbf{Y}_{(k)}^{l+1}, k = 2, \cdots, N$ is $O(\sum_{k=2}^{N} r_k \prod_{n=1}^{N} I_n)$.

When $k = 1$, the $\mathbf{Y}_{(1)}$-subproblem is expressed as:

$$\mathbf{Y}_{(1)}^{l+1} = \arg\min_{\mathbf{Y}_{(1)}} \frac{\alpha_1}{2}\|\mathbf{Y}_{(1)} - \mathbf{A}_1^{l+1}\mathbf{X}_1^{l+1}\|_F^2 + \frac{\gamma}{2}\|\mathbf{Y}_{(1)} - \mathbf{Y}_{(1)}^{l}\|_F^2 + \lambda\langle\phi(\kappa(\mathbf{Y}_{(1)})), |\nabla\mathbf{Y}_{(1)}|\rangle. \tag{23}$$

We solve the $\mathbf{Y}_{(1)}$-subproblem via the alternating direction method of multipliers (ADMM) algorithm [42,43]. To refrain from computing the gradient of the term pertaining to curvature, we consider $\phi(\kappa)$ in (23) as a weighted matrix of $|\nabla\mathbf{Y}_{(1)}|$. Since $\nabla = (\mathbf{D}_1, \mathbf{D}_2)$, by introducing two auxiliary variables $\mathbf{Q}_1$ and $\mathbf{Q}_2$, (23) can be rewritten as:

$$\begin{aligned} \arg\min_{\mathbf{Y}_{(1)},\mathbf{Q}_1,\mathbf{Q}_2} \quad & \frac{\alpha_1}{2}\|\mathbf{Y}_{(1)} - \mathbf{A}_1^{l+1}\mathbf{X}_1^{l+1}\|_F^2 + \frac{\gamma}{2}\|\mathbf{Y}_{(1)} - \mathbf{Y}_{(1)}^{l}\|_F^2 + \lambda\langle\phi(\kappa), |(\mathbf{Q}_1, \mathbf{Q}_2)|\rangle \\ \text{s.t.} \quad & \mathbf{Q}_1 = \mathbf{D}_1\mathbf{Y}_{(1)},\ \mathbf{Q}_2 = \mathbf{D}_2\mathbf{Y}_{(1)}. \end{aligned} \tag{24}$$

The corresponding augmented Lagrangian function of (24) is represented as:

$$\begin{aligned} L(\mathbf{Y}_{(1)}, \mathbf{Q}_1, \mathbf{Q}_2, \mathbf{W}_1, \mathbf{W}_2) = & \frac{\alpha_1}{2}\|\mathbf{Y}_{(1)} - \mathbf{A}_1^{l+1}\mathbf{X}_1^{l+1}\|_F^2 + \frac{\gamma}{2}\|\mathbf{Y}_{(1)} - \mathbf{Y}_{(1)}^{l}\|_F^2 + \lambda\langle\phi(\kappa), |(\mathbf{Q}_1, \mathbf{Q}_2)|\rangle \\ & + \langle\mathbf{W}_1, \mathbf{Q}_1 - \mathbf{D}_1\mathbf{Y}_{(1)}\rangle + \frac{\beta}{2}\|\mathbf{Q}_1 - \mathbf{D}_1\mathbf{Y}_{(1)}\|_F^2 \\ & + \langle\mathbf{W}_2, \mathbf{Q}_2 - \mathbf{D}_2\mathbf{Y}_{(1)}\rangle + \frac{\beta}{2}\|\mathbf{Q}_2 - \mathbf{D}_2\mathbf{Y}_{(1)}\|_F^2, \end{aligned} \tag{25}$$

where $\mathbf{W}_1$ and $\mathbf{W}_2$ denote the Lagrangian multipliers and $\beta > 0$ is the penalty parameter. Then, the problem (25) can be updated as follows:

$$\begin{cases} \mathbf{Y}_{(1)}^{l+1,c+1} & = \arg\min_{\mathbf{Y}_{(1)}} L(\mathbf{Y}_{(1)}, \mathbf{Q}_1^c, \mathbf{Q}_2^c, \mathbf{W}_1^c, \mathbf{W}_2^c), \\ (\mathbf{Q}_1^{c+1}, \mathbf{Q}_2^{c+1}) & = \arg\min_{(\mathbf{Q}_1,\mathbf{Q}_2)} L(\mathbf{Y}_{(1)}^{l+1,c+1}, \mathbf{Q}_1, \mathbf{Q}_2, \mathbf{W}_1^c, \mathbf{W}_2^c), \\ \mathbf{W}_1^{c+1} & = \mathbf{W}_1^c + \beta(\mathbf{Q}_1^{c+1} - \mathbf{D}_1\mathbf{Y}_{(1)}^{l+1,c+1}), \\ \mathbf{W}_2^{c+1} & = \mathbf{W}_2^c + \beta(\mathbf{Q}_2^{c+1} - \mathbf{D}_2\mathbf{Y}_{(1)}^{l+1,c+1}). \end{cases} \tag{26}$$

Next, we give the solving procedure of the first two steps in (26).

For $\mathbf{Y}_{(1)}$-subproblem in (26), we solve the following problem:

$$\begin{aligned} \mathbf{Y}_{(1)}^{l+1,c+1} = & \arg\min_{\mathbf{Y}_{(1)}} \frac{\alpha_1}{2}\|\mathbf{Y}_{(1)} - \mathbf{A}_1^{l+1}\mathbf{X}_1^{l+1}\|_F^2 + \frac{\gamma}{2}\|\mathbf{Y}_{(1)} - \mathbf{Y}_{(1)}^{l}\|_F^2 \\ & + \frac{\beta}{2}\|\mathbf{Q}_1^c - \mathbf{D}_1\mathbf{Y}_{(1)} + \frac{\mathbf{W}_1^c}{\beta}\|_F^2 + \frac{\beta}{2}\|\mathbf{Q}_2^c - \mathbf{D}_2\mathbf{Y}_{(1)} + \frac{\mathbf{W}_2^c}{\beta}\|_F^2. \end{aligned} \tag{27}$$

Assuming the periodic boundary condition in operators $\mathbf{D}_1$ and $\mathbf{D}_2$, the solution of the $\mathbf{Y}_{(1)}$-subproblem can be formed as:

$$\mathbf{Y}_{(1)}^{l+1,c+1} = \mathcal{F}^{-1}\left(\frac{\mathcal{F}(\mathbf{R})}{\mathcal{F}(\mathbf{S})}\right), \tag{28}$$

where $\mathbf{R} = \alpha_1\mathbf{A}_1^{l+1}\mathbf{X}_1^{l+1} + \gamma\mathbf{Y}_{(1)}^{l} + \beta\mathbf{D}_1^T\mathbf{Q}_1^c + \mathbf{D}_1^T\mathbf{W}_1^c + \beta\mathbf{D}_2^T\mathbf{Q}_2^c + \mathbf{D}_2^T\mathbf{W}_2^c$, $\mathbf{S} = (\alpha_1 + \gamma)\mathbf{I} + \beta\mathbf{D}_1^T\mathbf{D}_1 + \beta\mathbf{D}_2^T\mathbf{D}_2$, $\mathcal{F}$ is the fast Fourier transform and $\mathcal{F}^{-1}$ is the inverse Fourier transform. The cost of computing $\mathbf{Y}_{(1)}^{l+1,c+1}$ is provided mainly by the fast Fourier transforms on $I_1 \times \prod_{k=2}^{N} I_k$ matrix. Therefore, the computing complexity is $O(\prod_{k=1}^{N} I_k \log \prod_{k=1}^{N} I_k)$.

For $(\mathbf{Q}_1, \mathbf{Q}_2)$-subproblem in (26), we have

$$\begin{aligned} (\mathbf{Q}_1^{c+1}, \mathbf{Q}_2^{c+1}) = & \arg\min_{(\mathbf{Q}_1,\mathbf{Q}_2)} \frac{\beta}{2}\|\mathbf{Q}_1 - \mathbf{D}_1\mathbf{Y}_{(1)}^{l+1,c+1} + \frac{\mathbf{W}_1^c}{\beta}\|_F^2 + \frac{\beta}{2}\|\mathbf{Q}_2 - \mathbf{D}_2\mathbf{Y}_{(1)}^{l+1,c+1} + \frac{\mathbf{W}_2^c}{\beta}\|_F^2 \\ & + \lambda\langle\phi(\kappa), |(\mathbf{Q}_1, \mathbf{Q}_2)|\rangle. \end{aligned} \tag{29}$$

(29) can be solved via the 2-D shrinkage operator [44] and updated as:

$$\left((\mathbf{Q}_1^{c+1})_{i,j}, (\mathbf{Q}_2^{c+1})_{i,j}\right) = \max\left\{\|\mathbf{P}_{i,j}\|_2 - \frac{\lambda\phi(\kappa(\mathbf{Y}_{(1)}^{l+1,c+1}))}{\beta}, 0\right\}\frac{\mathbf{P}_{i,j}}{\|\mathbf{P}_{i,j}\|_2},\ 1 \le i \le I_1, 1 \le j \le s_1, \tag{30}$$

where $\mathbf{P}_{i,j} = \left((\mathbf{D}_1\mathbf{Y}_{(1)}^{l+1,c+1} - \frac{\mathbf{W}_1^c}{\beta})_{i,j}, (\mathbf{D}_2\mathbf{Y}_{(1)}^{l+1,c+1} - \frac{\mathbf{W}_2^c}{\beta})_{i,j}\right)$, and we assign $0 \cdot (0/0) = 0$. The complexity of getting $(\mathbf{Q}_1, \mathbf{Q}_2)$ is $O(\prod_{k=1}^{N} I_k)$.

After obtaining all $\mathbf{Y}_{(k)}^{l+1}, k = 1, \cdots, N$, we can update tensor $\mathcal{Y}^{l+1}$ by:

$$\mathcal{Y}^{l+1} = \mathcal{P}_{\Omega^C}\left(\sum_{k=1}^{N} \alpha_k \text{fold}_{(k)}(\mathbf{Y}_{(k)}^{l+1})\right) + \mathcal{O}. \tag{31}$$

The computing complexity in inner loop of solving $\mathbf{Y}_{(1)}$-subproblem is $O(\prod_{k=1}^{N} I_k \log \prod_{k=1}^{N} I_k)$. In summary, the complexity of solving all variables at each iteration is $O(\sum_{k=1}^{N}(I_k r_k^2 + r_k^2 s_k) + (\sum_{k=1}^{N} r_k + \log \prod_{k=1}^{N} I_k) \prod_{k=1}^{N} I_k)$.

Finally, the proposed algorithm is described in Algorithm 1.

Algorithm 1 The PAM algorithm for solving (5).

Input: The incomplete tensor $\mathcal{O} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$, index set Ω, the estimated Tucker rank $r_1, \cdots, r_N$.
Output: The recovered tensor $\mathcal{Y}$.
Parameters: $\alpha_k, k = 1, \cdots, N$, λ, ω, γ, β, l_{max}, c_{max}.
Initialization: $\mathbf{A}_k^0 = \text{rand}(I_k \times r_k)$, $\mathbf{X}_k^0 = \text{rand}(r_k \times \prod_{n \neq k}^{N} I_n)$, $k = 1, \cdots, N$.
Out loop: while not converged and $l < l_{max}$ **do**
  Update $\mathbf{A}_k^l$, $k = 1, 2, \cdots, N$ via (19);
  Update $\mathbf{X}_k^l$, $k = 1, 2, \cdots, N$ via (20);
  Inner loop: while not converged and $c < c_{max}$ **do**
    Update $\mathbf{Y}_{(1)}^l$ via (28);
    Update κ via (15);
    Update $(\mathbf{Q}_1, \mathbf{Q}_2)$ via (30);
    Update $\mathbf{W}_1$ and $\mathbf{W}_2$ via (26);
  end While
  Update $\mathbf{Y}_{(k)}^l, k = 2, 3, \cdots, N$ via (22);
  Update $\mathcal{Y}$ via (31);
end While

3.3. Convergence analysis

In this subsection, we prove that the sequence $\{\mathbf{A}^l, \mathbf{X}^l, \mathcal{Y}^l\}$ generated by the proposed PAM-based algorithm converges to a critical point of $f(\mathbf{A}, \mathbf{X}, \mathcal{Y})$. We first recall the preparation knowledge.

Lemma 1. *[45] Assume $f : \mathbb{R}^n \to \mathbb{R} \cup +\infty$ is a proper lower semi-continuous function. Let $\{x^k\}_{k \in \mathbb{N}}$ be a sequence which satisfies the following conditions:*

A1. *For $\forall k \in \mathbb{N}$, there exist positive constant a so that $f(x^{k+1}) + a\|x^{k+1} - x^k\|^2 \leq f(x^k)$;*
A2. *For $\forall k \in \mathbb{N}$, $\exists v^{k+1} \in \partial f(x^{k+1})$, and it satisfies $\|v^{k+1}\|_2 \leq b\|x^{k+1} - x^k\|$ for a constant $b \in (0, +\infty)$;*
A3. *$\exists \{x^{k_j}\}_{j \in \mathbb{N}}$ and $\tilde{x} \in \mathbb{R}^n$ so that $x^{k_j} \to \tilde{x}$ and $f(x^{k_j}) \to f(\tilde{x})$, when $j \to \infty$.*

If f satisfies the Kurdyka-Łojasiewicz (KŁ) property [46] at the cluster point $\tilde{x}$, then $\{x^k\}_{k \in \mathbb{N}}$ converges to $\bar{x} = \tilde{x}$ when $k \to \infty$, where $\bar{x}$ is a critical point of f.

Now, we study the convergence of the proposed algorithm. For simplicity's sake, the short version of objective function (16) can be expressed as:

$$f(\mathbf{A}, \mathbf{X}, \mathcal{Y}) = f_1(\mathbf{A}, \mathbf{X}, \mathcal{Y}) + f_2(\mathcal{Y}), \tag{32}$$

where $f_1(\mathbf{A}, \mathbf{X}, \mathcal{Y}) = \sum_{k=1}^{N} \frac{\alpha_k}{2} \|\mathbf{Y}_{(k)} - \mathbf{A}_k \mathbf{X}_k\|_F^2$ and $f_2(\mathcal{Y}) = \lambda \langle \phi(\kappa), |\nabla \mathbf{Y}_{(1)}| \rangle$. As we regard $\phi(\kappa(\mathbf{Y}_{(1)}))$ as a weighted matrix in solving $\mathbf{Y}_{(1)}$-subproblem, so f_2 is the Frobenius norm with coefficient matrix $\lambda\phi(\kappa)$. Next, we give the boundedness of $\{\mathbf{A}^l, \mathbf{X}^l, \mathcal{Y}^l\}$ generated by Algorithm 1. And then, we illustrate the function (32) and the sequence $\{\mathbf{A}^l, \mathbf{X}^l, \mathcal{Y}^l\}$ meets the conditions in Lemma 1 and its convergence theorem.

Lemma 2. *The sequence $\{\mathbf{A}^l, \mathbf{X}^l, \mathcal{Y}^l\}$ obtained by Algorithm 1 is bounded.*

Proof. We first prove the objective functions in (18) satisfy sufficient decrease property. Suppose that $\mathbf{A}^{l+1}$, $\mathbf{X}^{l+1}$, and $\mathcal{Y}^{l+1}$ represent the optimal solutions for each subproblem of (18), then it follows that

$$\begin{cases} f(\mathbf{A}^{l+1}, \mathbf{X}^l, \mathcal{Y}^l) + \frac{\gamma}{2}\|\mathbf{A}^{l+1} - \mathbf{A}^l\|_F^2 & \leq f(\mathbf{A}^l, \mathbf{X}^l, \mathcal{Y}^l), \\ f(\mathbf{A}^{l+1}, \mathbf{X}^{l+1}, \mathcal{Y}^l) + \frac{\gamma}{2}\|\mathbf{X}^{l+1} - \mathbf{X}^l\|_F^2 & \leq f(\mathbf{A}^{l+1}, \mathbf{X}^l, \mathcal{Y}^l), \\ f(\mathbf{A}^{l+1}, \mathbf{X}^{l+1}, \mathcal{Y}^{l+1}) + \frac{\gamma}{2}\|\mathcal{Y}^{l+1} - \mathcal{Y}^l\|_F^2 & \leq f(\mathbf{A}^{l+1}, \mathbf{X}^{l+1}, \mathcal{Y}^l). \end{cases} \tag{33}$$

Next, we prove the sequence $\{\mathbf{A}^l, \mathbf{X}^l, \mathcal{Y}^l\}$ is bounded. Since for $k = 1, 2, \cdots, N$,

$$\lim_{\|\mathbf{A}_k\|_F \to +\infty} \frac{\alpha_k}{2} \|\mathbf{Y}_{(k)} - \mathbf{A}_k \mathbf{X}_k\|_F^2 = +\infty,$$
$$\lim_{\|\mathbf{X}_k\|_F \to +\infty} \frac{\alpha_k}{2} \|\mathbf{Y}_{(k)} - \mathbf{A}_k \mathbf{X}_k\|_F^2 = +\infty,$$
$$\lim_{\|\mathcal{Y}\|_F \to +\infty} \frac{\alpha_k}{2} \|\mathbf{Y}_{(k)} - \mathbf{A}_k \mathbf{X}_k\|_F^2 = +\infty.$$

We can conclude that

$$\lim_{\|\mathbf{A}_k\|_F \to +\infty} f(\mathbf{A}, \mathbf{X}, \mathcal{Y}) = +\infty,$$
$$\lim_{\|\mathbf{X}_k\|_F \to +\infty} f(\mathbf{A}, \mathbf{X}, \mathcal{Y}) = +\infty,$$
$$\lim_{\|\mathcal{Y}\|_F \to +\infty} f(\mathbf{A}, \mathbf{X}, \mathcal{Y}) = +\infty.$$

Therefore, for unbounded $\{\mathbf{A}^l, \mathbf{X}^l, \mathcal{Y}^l\}$, $f(\mathbf{A}^{l+1}, \mathbf{X}^{l+1}, \mathcal{Y}^{l+1})$ tends to infinity. That is, the bounded nature of $f(\mathbf{A}^{l+1}, \mathbf{X}^{l+1}, \mathcal{Y}^{l+1})$ ensures a bounded solution for $\{\mathbf{A}^l, \mathbf{X}^l, \mathcal{Y}^l\}$.

In the following, we proceed to show that $f(\mathbf{A}^{l+1}, \mathbf{X}^{l+1}, \mathcal{Y}^{l+1})$ exhibits bounded behavior. From (33), we derive

$$\begin{aligned} f(\mathbf{A}^{l+1}, \mathbf{X}^{l+1}, \mathcal{Y}^{l+1}) &\le f(\mathbf{A}^{l+1}, \mathbf{X}^{l+1}, \mathcal{Y}^{l+1}) + \frac{\gamma}{2}\|\mathbf{A}^{l+1} - \mathbf{A}^l\|_F^2 + \frac{\gamma}{2}\|\mathbf{X}^{l+1} - \mathbf{X}^l\|_F^2 + \frac{\gamma}{2}\|\mathcal{Y}^{l+1} - \mathcal{Y}^l\|_F^2 \\ &\le f(\mathbf{A}^l, \mathbf{X}^l, \mathcal{Y}^l) \\ &\le f(\mathbf{A}^l, \mathbf{X}^l, \mathcal{Y}^l) + \frac{\gamma}{2}\|\mathbf{A}^l - \mathbf{A}^{l-1}\|_F^2 + \frac{\gamma}{2}\|\mathbf{X}^l - \mathbf{X}^{l-1}\|_F^2 + \frac{\gamma}{2}\|\mathcal{Y}^l - \mathcal{Y}^{l-1}\|_F^2 \\ &\le \\ &\cdots \\ &\le f(\mathbf{A}^0, \mathbf{X}^0, \mathcal{Y}^0). \end{aligned}$$

Therefore, $f(\mathbf{A}^{l+1}, \mathbf{X}^{l+1}, \mathcal{Y}^{l+1})$ is bounded, and it follows that $\{\mathbf{A}^l, \mathbf{X}^l, \mathcal{Y}^l\}$ is bounded.

Theorem 1. *The bounded sequence $\{\mathbf{A}^l, \mathbf{X}^l, \mathcal{Y}^l\}$ produced by Algorithm 1 can converge to a critical point of f.*

Proof. First, f_1 is proper, and also is a C^1 Lipschitz continuous gradient function about variables $\mathbf{A}$, $\mathbf{X}$, and $\mathcal{Y}$. f_2 is proper and continuous. So f is a proper lower semi-continuous function.

Second, the alternating procedure (18) is a special case of PAM framework in Remark 6.1 of [45] when $\mathbf{B}_i = \gamma \mathbf{I}$. Thus, the sequence $\{\mathbf{A}^l, \mathbf{X}^l, \mathcal{Y}^l\}$ generated via PAM framework meets the conditions A1, A2, and A3 in Lemma 1.

Third, we will illustrate that f satisfies the KŁ property at each iteration point $\{\mathbf{A}^l, \mathbf{X}^l, \mathcal{Y}^l\}$. The first fact is that the semi-algebraic function satisfies the KŁ property [47]. The union of finite semi-algebraic functions is still semi-algebraic [45]. $f_1(\mathbf{A}, \mathbf{X}, \mathcal{Y})$ is a polynomial function about variables $\mathbf{A}$, $\mathbf{X}$ and $\mathcal{Y}$, and it is formed by a linear mapping of some finite-dimensional spaces. In addition, polynomial is semi-algebraic function. So f_1 is semi-algebraic. Since the Frobenius norm is semi-algebraic function [41], $f_2(\mathcal{Y})$ is semi-algebraic. Ultimately, as the sum of f_1 and f_2, f is a semi-algebraic function and satisfies the KŁ property.

By Lemma 1, the bounded sequence $\{\mathbf{A}^l, \mathbf{X}^l, \mathcal{Y}^l\}$ converges to a critical point of f. □

4. Numerical results and discussion

In this section, we carry out numerical experiments on three datasets to show the performance of the LRTC method incorporating TCV (TCV-LRTC). We compare the proposed method with five different LRTC techniques: TMac [22], MF-TV [28], HaLRTC [19], LRTC-TV [29], and KLRTC [20].

The peak signal to noise rate (PSNR) and the structural similarity index (SSIM) [48] are employed to evaluate the restoration quality. All experiments are implemented using MATLAB R2021a on Windows 11, with Intel Core i5 CPU @2.50 GHz and 16 GB Memory.

We stop algorithm once the relative change (RelCha) is less than the tolerance ϵ (i.e., RelCha= $\frac{\|\mathcal{Y}^l - \mathcal{Y}^{l-1}\|_F}{\|\mathcal{Y}^{l-1}\|_F} < \epsilon$), or the maximum number of iterations is satisfied. The tolerance ϵ is generally set to be 10^{-4}.

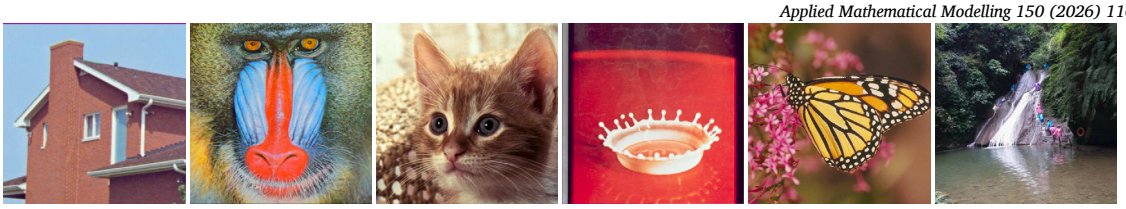

Fig. 3. The tested color images used in experiments.

Table 1
Quantitative evaluation of the reconstructed results by all compared methods for color images with different SRs.

Image	SR	0.1		0.2		0.3		0.4	
	Method	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
house	TMac	7.69	0.0283	12.50	0.1434	20.78	0.5358	28.08	0.7739
	MF-TV	12.71	0.0530	23.74	0.3532	27.38	0.7610	29.51	0.8590
	HaLRTC	20.26	0.4776	23.63	0.6456	26.52	0.7616	28.96	0.8409
	LRTC-TV	22.86	0.7208	26.47	0.8125	28.58	0.8592	30.18	0.8704
	KLRTC	22.45	0.7147	25.75	0.8060	27.73	0.8549	29.14	0.8894
	TCV-LRTC	**25.75**	**0.7211**	**28.31**	**0.8137**	**29.74**	**0.8595**	**32.59**	**0.8942**
baboon	TMac	6.28	0.0364	7.32	0.0648	8.67	0.0978	10.22	0.1387
	MF-TV	9.39	0.0702	15.16	0.1661	18.66	0.4479	21.79	0.5466
	HaLRTC	18.18	0.3110	20.05	0.4576	21.49	0.5741	22.93	0.6776
	LRTC-TV	20.29	0.4254	21.66	0.5636	22.75	0.6655	23.78	0.7462
	KLRTC	20.26	0.4246	21.63	0.5668	22.65	0.6642	23.62	0.7419
	TCV-LRTC	**20.63**	**0.4708**	**22.12**	**0.6010**	**23.02**	**0.6895**	**24.17**	**0.7606**
cat	TMac	7.20	0.0451	9.19	0.0762	12.04	0.1954	16.85	0.4797
	MF-TV	11.11	0.0736	16.99	0.2176	25.20	0.5504	29.28	0.8035
	HaLRTC	20.10	0.4223	23.25	0.6001	25.83	0.7274	28.11	0.8206
	LRTC-TV	22.49	0.5709	25.60	0.7134	27.39	0.7982	28.99	0.8555
	KLRTC	22.63	0.5716	25.48	0.7121	27.26	0.7928	28.94	0.8499
	TCV-LRTC	**24.62**	**0.6868**	**27.14**	**0.7808**	**28.73**	**0.8455**	**30.32**	**0.8887**
splash	TMac	14.91	0.4397	23.80	0.8068	32.65	0.9289	34.65	0.9544
	MF-TV	24.57	0.5551	31.31	0.9070	33.42	0.9399	34.49	0.9593
	HaLRTC	24.69	0.8131	28.96	0.8971	32.02	0.9384	34.38	0.9612
	LRTC-TV	27.21	0.9069	30.39	0.9270	31.49	0.9500	34.50	0.9619
	KLRTC	25.73	0.8784	28.71	0.9245	30.20	0.9445	31.31	0.9568
	TCV-LRTC	**30.29**	**0.9084**	**32.56**	**0.9334**	**33.71**	**0.9643**	**34.74**	**0.9754**
monarch	TMac	9.72	0.1452	12.28	0.2768	15.23	0.6443	18.54	0.8117
	MF-TV	13.23	0.1561	21.70	0.6556	26.07	0.8534	30.95	0.9558
	HaLRTC	18.13	0.6405	21.04	0.8096	23.69	0.8959	26.23	0.9411
	LRTC-TV	20.15	0.8001	24.32	0.9267	27.01	0.9627	29.04	0.9770
	KLRTC	19.87	0.8061	23.73	0.9217	26.27	0.9589	28.37	0.9759
	TCV-LRTC	**24.45**	**0.8915**	**27.76**	**0.9417**	**29.11**	**0.9649**	**31.95**	**0.9782**
waterfall	TMac	9.79	0.1743	11.39	0.2609	13.34	0.3840	16.65	0.6352
	MF-TV	14.47	0.1639	16.71	0.3863	23.39	0.7009	28.17	0.8974
	HaLRTC	21.30	0.5896	23.52	0.7545	25.43	0.8509	27.33	0.9125
	LRTC-TV	22.52	0.6484	24.69	0.8028	26.40	0.8832	27.87	0.9270
	KLRTC	22.76	0.6597	24.65	0.8009	26.10	0.8747	27.49	0.9207
	TCV-LRTC	**24.28**	**0.7547**	**26.55**	**0.8636**	**27.99**	**0.9174**	**29.39**	**0.9478**

4.1. Color image completion

We test the capability of all methods on six color images,[1] called *house*, *baboon*, *cat*, *splash*, *monarch*, and *waterfall*. The first three color images are of size $256 \times 256 \times 3$, and the last three images are of size $512 \times 512 \times 3$. Fig. 3 exhibits all the tested color images. The observed data is obtained by random sampling method and the sampling rates (SRs) are set to 0.1, 0.2, 0.3, and 0.4, respectively.

Table 1 exhibits the PSNR and SSIM values of the recovered results performed by TCV-LRTC and other compared methods. The highest values are marked with black bold font. We can see from Table 1 that the proposed method TCV-LRTC obtains the highest PSNR and SSIM values for all color images. For visualization, we symbolically show some experimental results in Fig. 4. As observed in Fig. 4, when the SR is lower, TMac and MF-TV have terrible reconstructed results, the recovery result of HaLRTC has a lot of

[1] https://sipi.usc.edu/database/database.php.

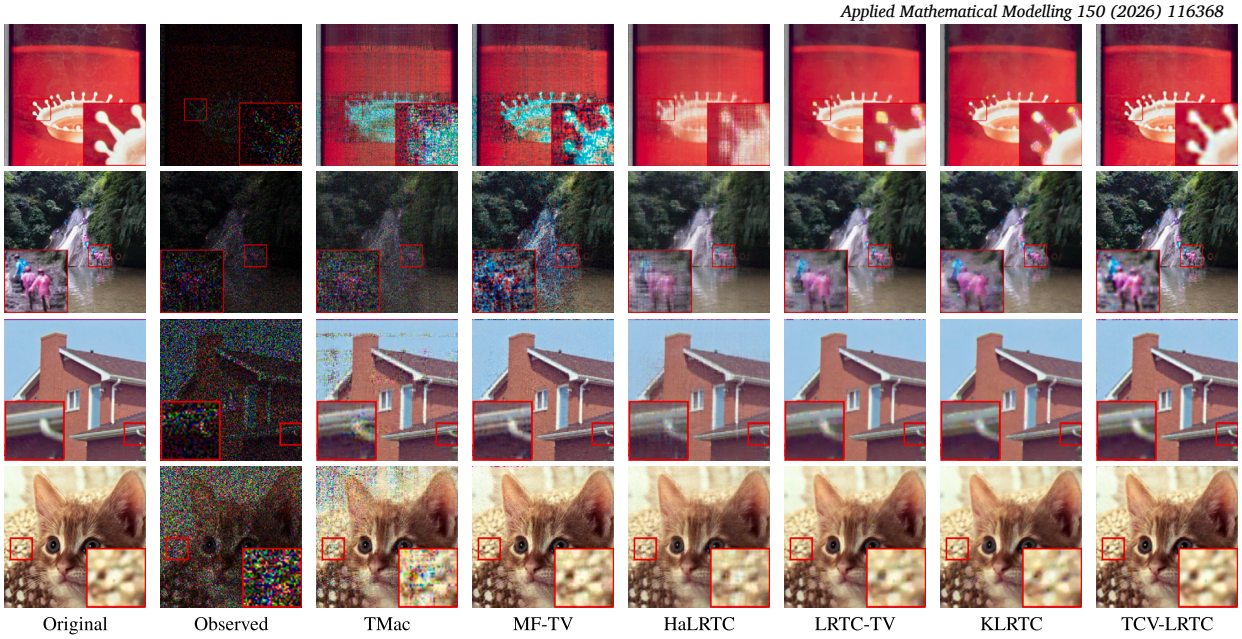

Fig. 4. The recovered color images by all compared methods. From top to bottom: *splash* with SR= 0.1, *waterfall* with SR= 0.2, *house* with SR= 0.3, and *cat* with SR= 0.4. From left to right: original data, observed data, the recovered results by TMac, MF-TV, HaLRTC, LRTC-TV, KLRTC, and TCV-LRTC.

Fig. 5. The recovered results of *monarch* by all compared methods. From top to bottom: SR= 0.1, 0.2, 0.3, and 0.4. From left to right: original data, observed data, the recovered results by TMac, MF-TV, HaLRTC, LRTC-TV, KLRTC, and TCV-LRTC.

vertical thorns, and LRTC-TV and KLRTC have too smooth effects. For all experiments on color images, our method has achieved the most excellent performance. While maintaining the edges of image, TCV-LRTC can also keep good contrast and corners. The detail features of the recovered results by TCV-LRTC are most similar to original images. Moreover, TCV-LRTC can obtain more accurate effects when the image has some objects with elongated structures, see the recovered results of tentacle of *monarch* with different SRs in Fig. 5.

Table 2
Quantitative evaluation of the reconstructed results by all compared methods for videos with different SRs.

Video	SR	0.1		0.2		0.3	
	Method	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
street $158\times238\times24$	TMac	7.98	0.0572	9.41	0.0989	11.04	0.1687
	MF-TV	8.32	0.0686	11.74	0.1226	15.58	0.2394
	HaLRTC	19.27	0.4781	22.45	0.6578	24.68	0.7680
	LRTC-TV	21.27	0.5888	23.99	0.7186	25.55	0.7943
	KLRTC	21.14	0.5866	23.70	0.7106	25.39	0.7895
	TCV-LRTC	**21.98**	**0.6294**	**24.04**	**0.7458**	**25.58**	**0.8190**
bus $256\times256\times30$	TMac	10.23	0.0841	11.31	0.1304	12.50	0.1860
	MF-TV	10.72	0.0983	13.05	0.1741	15.57	0.2671
	HaLRTC	18.04	0.4341	20.00	0.5750	21.66	0.6819
	LRTC-TV	19.23	0.4926	21.04	0.6406	22.69	0.7484
	KLRTC	18.62	0.4824	20.77	0.6302	22.25	0.7348
	TCV-LRTC	**19.77**	**0.5105**	**22.29**	**0.6677**	**23.60**	**0.7641**
salesman $144\times176\times75$	TMac	14.11	0.1494	19.01	0.5538	25.30	0.8154
	MF-TV	15.06	0.1614	22.77	0.7089	26.23	0.8272
	HaLRTC	20.06	0.5964	23.43	0.7679	26.22	0.8632
	LRTC-TV	22.45	0.6664	25.89	0.8264	28.25	0.8981
	KLRTC	20.52	0.5690	23.02	0.7155	24.76	0.8012
	TCV-LRTC	**23.35**	**0.7141**	**26.48**	**0.8578**	**28.58**	**0.9111**
suzie $144\times176\times75$	TMac	15.32	0.2248	23.05	0.7143	32.11	0.8878
	MF-TV	21.03	0.3722	29.65	0.8082	34.00	0.9127
	HaLRTC	23.33	0.6967	26.90	0.8001	29.49	0.8644
	LRTC-TV	26.59	0.7955	30.35	0.8833	32.81	0.9228
	KLRTC	26.50	0.7813	29.96	0.8692	32.21	0.9113
	TCV-LRTC	**29.88**	**0.8682**	**32.87**	**0.9159**	**34.53**	**0.9385**

Fig. 6. The recovered results by all compared methods with SR= 0.3. The first (last) two rows are two frames of the recovered video *salesman* (*suzie*). From left to right: original data, observed data, the recovered results by TMac, MF-TV, HaLRTC, LRTC-TV, KLRTC, and TCV-LRTC.

4.2. Video completion

We test four gray videos[2] with different sizes via TMac, MF-TV, HaLRTC, LRTC-TV, KLRTC and the proposed method TCV-LRTC in this subsection. Table 2 presents the average PSNR and SSIM values of all tested videos by different methods. Obviously, our curvature-based method consistently achieves highest quality evaluations for all videos with different sampling rates. Fig. 6 exhibits the recovered visual effects by different methods for videos *salesman* and *suzie* with SR= 0.3. We can see that the results reconstructed by TCV-LRTC outperform other comparative methods. MF-TV and LRTC-TV, which are TV-based method, cause undesirable staircase artifacts. The proposed method can avoid staircase artifacts caused by TV, while preserving sharp edges and fine details. Fig. 7 shows

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

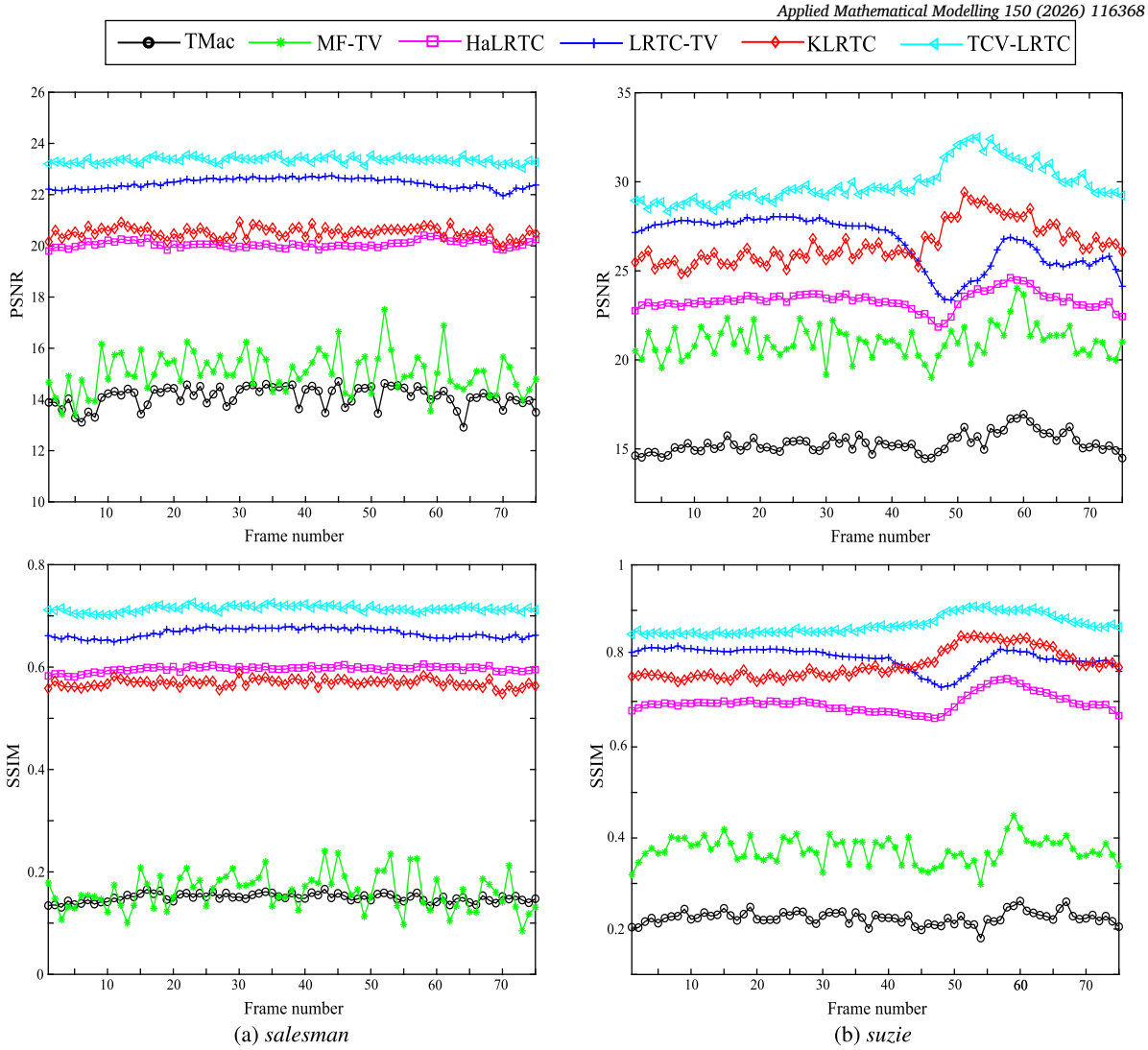

Fig. 7. The PSNR and SSIM values of all frames of videos reconstructed by all compared methods with SR= 0.1.

the PSNR and SSIM values of all frames of videos *salesman* and *suzie* reconstructed by different methods with SR= 0.1. It can be seen that all frames recovered by TCV-LRTC obtain the highest PSNR and SSIM values. In Fig. 8, we present the pixel values of one randomly selected tube recovered by different methods for *suzie* with SR= 0.1, it shows that the result recovered by TCV-LRTC is closer to the original data, which validates its ability to preserve image contrast.

4.3. MRI completion

We study the completion results of all methods on two MRIs[3] of size $181 \times 217 \times 40$. The SRs are taken as 0.1, 0.2, and 0.3. Table 3 reports the average PSNR and SSIM values of the recovered MRIs by different methods. For two MRIs with different SRs, the proposed method shows better performance than the comparative methods quantitatively. In particular, the PSNR values of MRIs reconstructed by TCV-LRTC are at least 3 dB higher than those recovered by other methods when SR= 0.1 and 0.2. Fig. 9 displays the recovered results of *MRI-I* by different methods. It is clear that, no matter how much SR is set, TCV-LRTC has obtained the best visual effect. The main reason is that these images are made up of surface patches which are piecewise smooth and curvatures used in our method are computed through small patches around the center point. So the proposed curvature-based regularization is a better choice to promote piecewise smoothness for local small patches and preserve significant geometric features of image, therefore gaining a competitive advantage. Fig. 10 shows the PSNR and SSIM values of all frames of *MRI-II* recovered by different methods with SR= 0.1 and 0.2.

[3] https://brainweb.bic.mni.mcgill.ca/brainweb/.

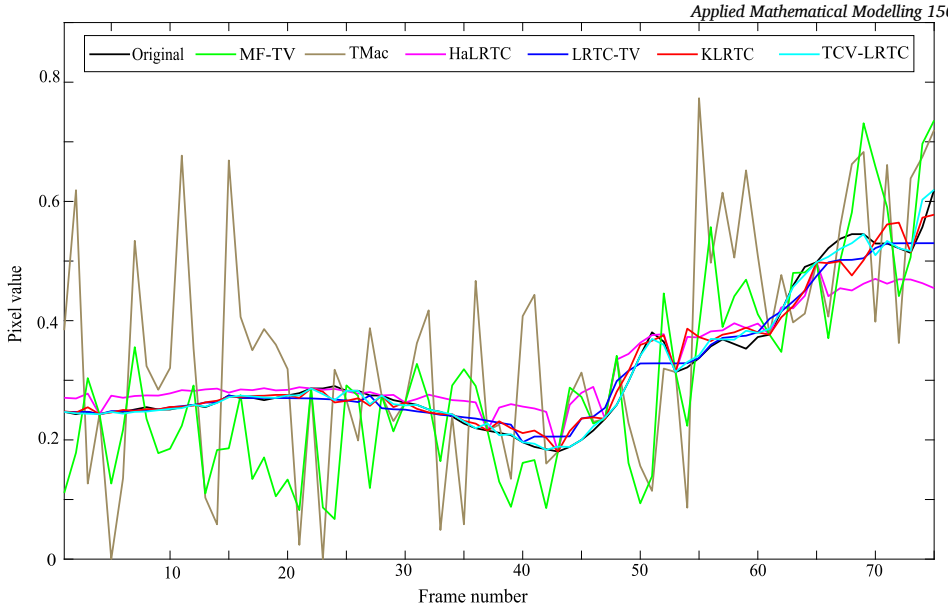

Fig. 8. The pixel values of one selected tube reconstructed by all compared methods for *suzie* with SR= 0.1.

Table 3
Quantitative evaluation of the reconstructed results by all compared methods for MRIs with different SRs.

MRI	SR	0.1		0.2		0.3	
	Method	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
MRI-I	TMac	12.70	0.1295	16.26	0.3192	21.48	0.7064
	MF-TV	14.77	0.1736	22.38	0.5128	30.39	0.8484
	HaLRTC	18.64	0.4822	22.79	0.6871	26.23	0.8206
	LRTC-TV	22.59	0.7380	27.22	0.8748	30.31	0.9305
	KLRTC	21.04	0.7019	25.34	0.8433	28.18	0.9069
	TCV-LRTC	**26.73**	**0.8234**	**31.19**	**0.9356**	**33.56**	**0.9622**
MRI-II	TMac	12.85	0.1818	17.44	0.4793	34.00	0.9382
	MF-TV	14.96	0.2550	26.74	0.7171	34.98	0.9456
	HaLRTC	19.15	0.4511	22.84	0.6526	26.34	0.7926
	LRTC-TV	21.03	0.7228	26.04	0.8827	29.60	0.9394
	KLRTC	19.47	0.5655	23.81	0.7741	27.20	0.8673
	TCV-LRTC	**26.18**	**0.8401**	**31.35**	**0.9483**	**35.37**	**0.9693**

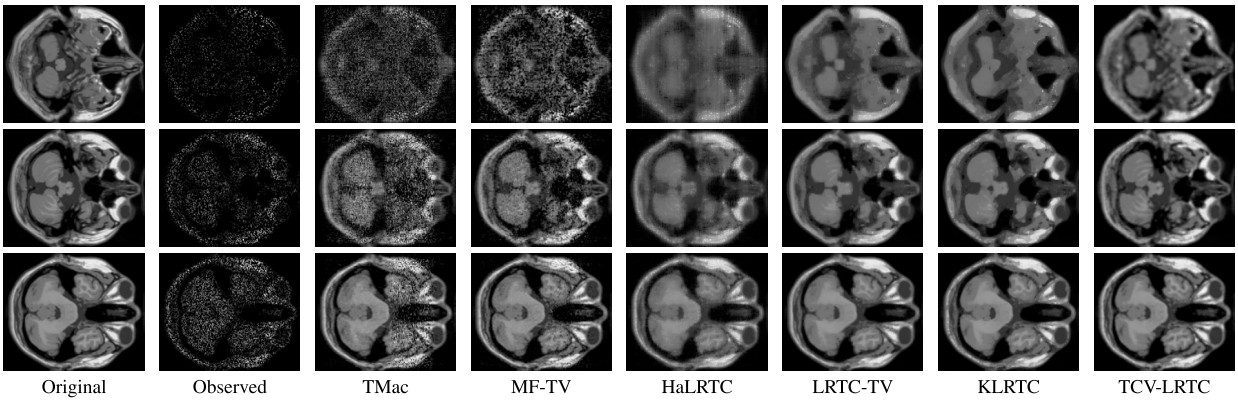

Fig. 9. Reconstructed results of *MRI-I* by all compared methods. From top to bottom: one frame of the recovered *MRI-I* with SR= 0.1, 0.2, and 0.3, respectively. From left to right: original data, observed data, the recovered results by TMac, MF-TV, HaLRTC, LRTC-TV, KLRTC, and TCV-LRTC.

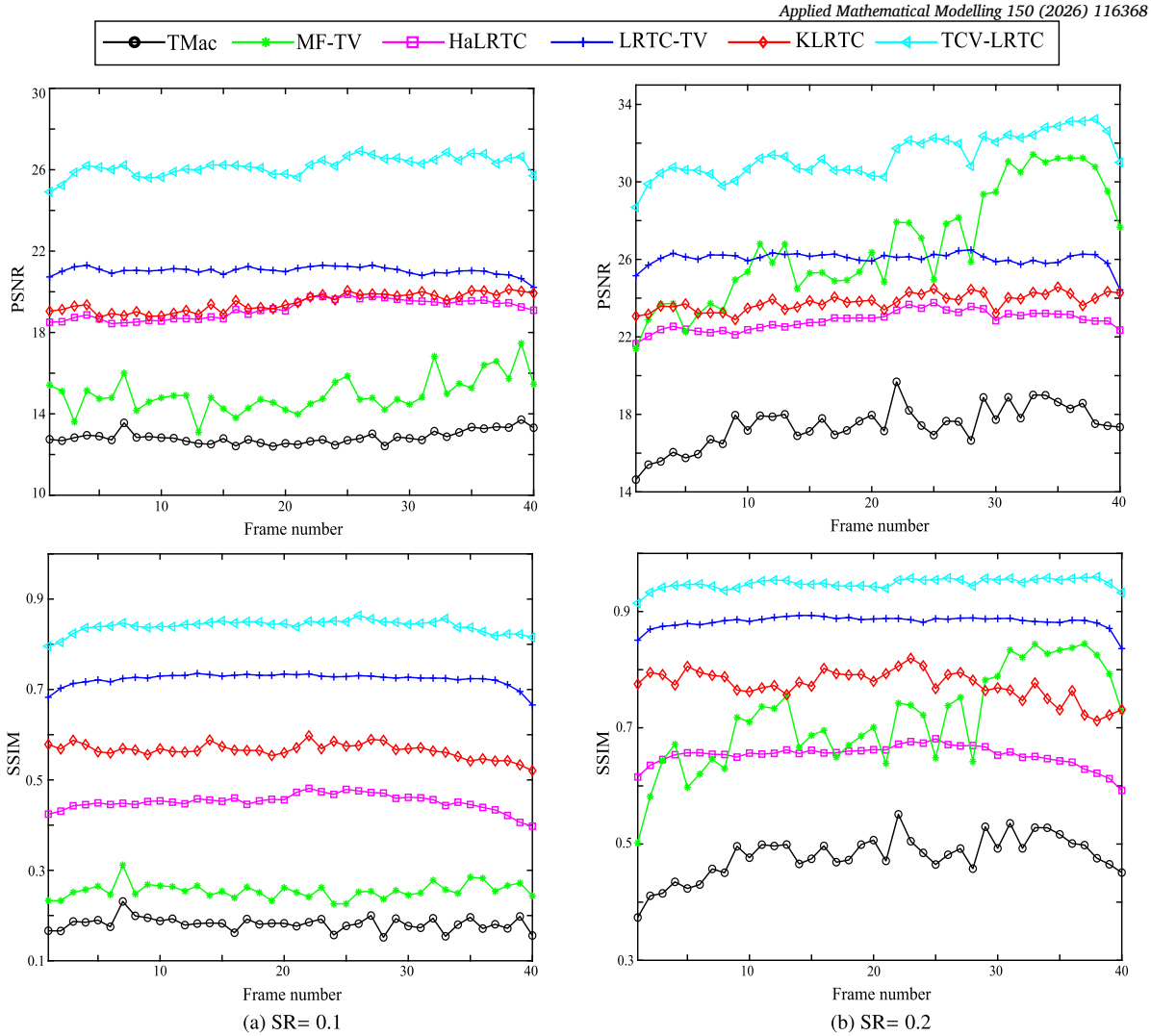

Fig. 10. The PSNR and SSIM values of all frames of *MRI-II* reconstructed by all compared methods with SR= 0.1 and 0.2.

4.4. Discussions

In this subsection, we analyze the algorithmic parameters and their influence on model performance, as presented in this work. In all the experiments, the weights $\alpha_k = 1/N$, the maximum numbers of outer and inner iterations, denoted as l_{max} and c_{max} respectively, were set to 500 and 300. We mainly analyze the influence of parameters λ, γ, β and ω in $\phi(\kappa)$. The tested data is *suzie* with SR= 0.3. Fig. 11 displays the Relcha, PSNR and SSIM values in terms of the iteration number for different regularization parameter λ. It can be seen that when λ is respectively selected from $\{0.001, 0.01, 0.1\}$, the corresponding PSNR and SSIM values of recovered results are almost the same. So we select a fixed $\lambda = 0.1$ in all our experiments to improve the experimental efficiency. As shown in Fig. 12, the selection of parameter γ follows a similar rationale to that of parameter λ. When $\gamma = 0.001$ or 0.01, the algorithm requires nearly the same number of iterations to converge. Therefore, we opt for $\gamma = 0.01$ to achieve marginally higher PSNR and SSIM values. In Fig. 13, we display the change curves of the Relcha, PSNR and SSIM values in regard to the initial penalty parameter β^0. We can see this conclusion, a larger β^0 could produce poorer result, and a smaller $\beta^0 = 10$ results almost the same PSNR and SSIM values as $\beta^0 = 50$. Moreover, a smaller β^0 can lead to increasing the number of iteration. Therefore, we empirically select β^0 from $\{50, 100\}$. The parameter ω in function $\phi(\kappa)$ is selected from the candidate set $\{5, 15, 20, 25, 30\}$. Fig. 14 illustrates the curves of RelCha, PSNR and SSIM values of the proposed method in regard to parameter ω during the iteration process. Evidently, different ω yields nearly identical final PSNR and SSIM values, with the biggest difference being the number of iterations. Consequently, in all our experiments, ω is empirically set to 25 to minimize the required number of iterations. According to the curve charts of RelCha in Fig. 11-14, we can see that RelCha rapidly decrease to the tolerance ε no matter which λ, γ, β, and ω we choose, and this can also prove the numerical convergence of the proposed method.

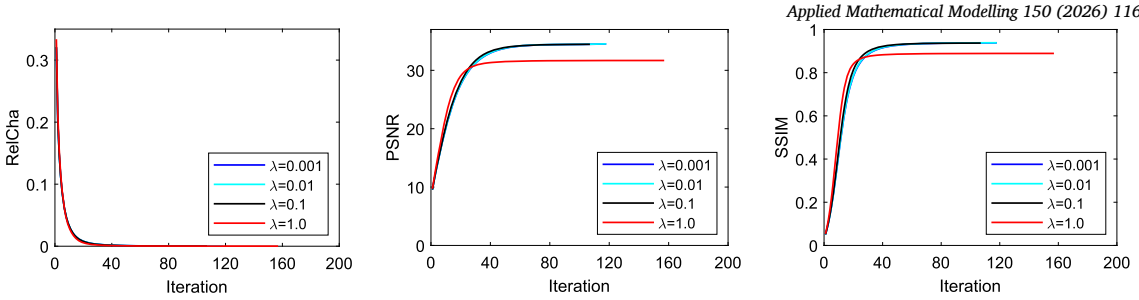

Fig. 11. The RelCha, PSNR and SSIM values of the proposed method in regard to parameter λ.

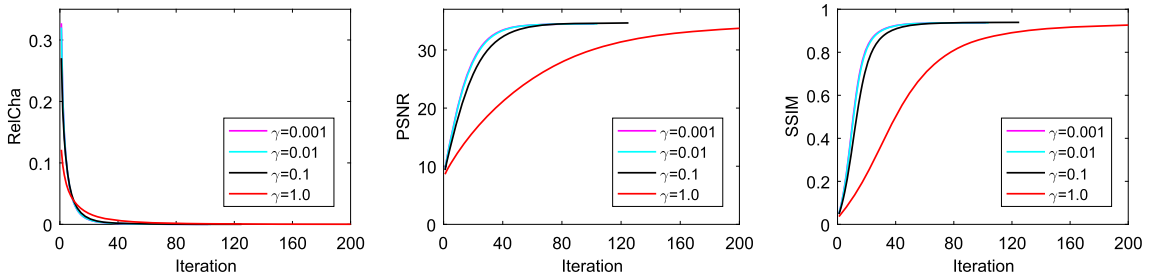

Fig. 12. The RelCha, PSNR and SSIM values of the proposed method in regard to parameter γ.

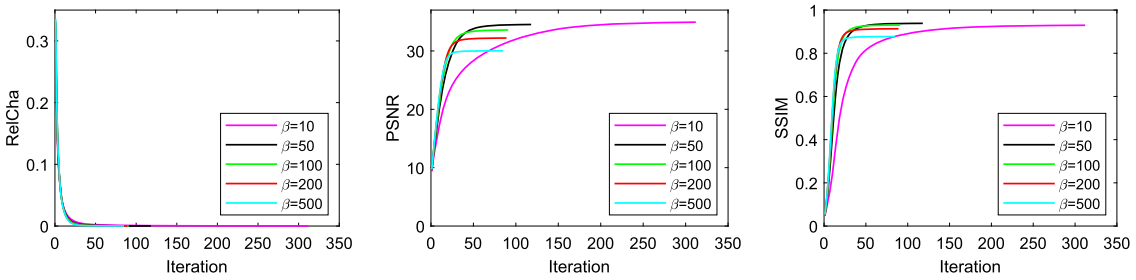

Fig. 13. The RelCha, PSNR and SSIM values of the proposed method in regard to parameter β.

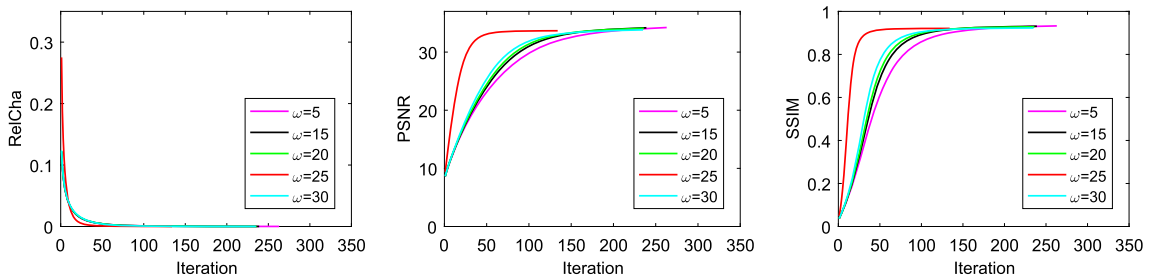

Fig. 14. The RelCha, PSNR and SSIM values of the proposed method in regard to parameter ω.

5. Conclusions

In this work, we first design a total curvature variation regularization which is based on curvature and total variation. By integrating this regularization term with low-rank matrix factorization, we propose a model for LRTC problem which can capture global and local prior information of high-dimensional image. An efficient PAM-based algorithm is used to solve the proposed model. It is demonstrated that the sequence formed by the proposed algorithm can converge to a critical point. Moreover, numerical experiments reveal that: 1) quantitatively, the proposed method obtains higher PSNR and SSIM values than the compared methods. 2) qualitatively, our method can not only preserve sharp edges, but keep contrast and details of images. In the future work, one remaining challenge for the proposed method is to optimize the parameter adjustment strategy and develop efficient algorithms for large-scale high-dimensional data processing.

CRediT authorship contribution statement

Zhi Xu: Writing – original draft, Visualization, Software, Project administration, Methodology. **Jing-Hua Yang:** Writing – review & editing, Validation, Resources, Data curation. **Xi-Le Zhao:** Writing – review & editing, Supervision, Conceptualization. **Xi-hong Yan:** Writing – review & editing, Validation, Methodology. **Chuan-long Wang:** Writing – review & editing, Formal analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The research is supported by NSFC (12371381, 12401605, 12371456), Scientific and Technologial Innovation Programs of Higher Education Institutions in Shanxi (2023L241), 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).

Data availability

Data will be made available on request.

References

[1] J.A. Bengua, H.N. Phien, H.D. Tuan, M.N. Do, Efficient tensor completion for color image and video recovery: low-rank tensor train, IEEE Trans. Image Process. 26 (5) (2017) 2466–2479.
[2] 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.
[3] J.-M. Wu, S.-B. Yin, T.-X. Jiang, G.-S. Liu, X.-L. Zhao, PALADIN: a novel plug-and-play 3D CS-MRI reconstruction method, Inverse Probl. 41 (3) (2025) 035014.
[4] T.-Y. Ji, T.-Z. Huang, X.-L. Zhao, T.-H. Ma, L.-J. Deng, A non-convex tensor rank approximation for tensor completion, Appl. Math. Model. 48 (2017) 410–422.
[5] 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.
[6] T.-X. Jiang, M.K. Ng, J. Pan, G.-J. Song, Nonnegative low rank tensor approximations with multidimensional image applications, Numer. Math. 153 (1) (2023) 141–170.
[7] X. Chen, T.-X. Jiang, Y. Hu, J. Yu, M.K. Ng, Adaptive sampling with tensor leverage scores for exact low-rank third-order tensor completion, Appl. Math. Model. 138 (2025) 115744.
[8] 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.
[9] M. Ding, X.-L. Zhao, J.-H. Yang, Z. Zhou, M.K. Ng, Bilateral tensor low-rank representation for insufficient observed samples in multidimensional image clustering and recovery, SIAM J. Imaging Sci. 18 (1) (2025) 20–59.
[10] A. Singer, M. Cucuringu, Uniqueness of low-rank matrix completion by rigidity theory, SIAM J. Matrix Anal. Appl. 31 (4) (2010) 1621–1641.
[11] Y. Yang, The epsilon-alternating least squares for orthogonal low-rank tensor approximation and its global convergence, SIAM J. Matrix Anal. Appl. 41 (4) (2020) 1797–1825.
[12] R. Yamamoto, H. Hontani, A. Imakura, T. Yokota, Fast algorithm for low-rank tensor completion in delay-embedded space, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 2058–2066.
[13] Y.-B. Zheng, T.-Z. Huang, X.-L. Zhao, T.-X. Jiang, T.-Y. Ji, T.-H. Ma, Tensor n-tubal rank and its convex relaxation for low-rank tensor recovery, Inf. Sci. 532 (2020) 170–189.
[14] G. Yu, C. Chen, S. Wan, L. Qi, Y. Xu, Multi-mode tensor train factorization with spatial-spectral regularization for third-order tensor completion, Appl. Math. Model. 141 (2025) 115921.
[15] M. Ding, T.-Z. Huang, X.-L. Zhao, T.-H. Ma, Tensor completion via nonconvex tensor ring rank minimization with guaranteed convergence, Signal Process. 194 (2022) 108425.
[16] 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.
[17] 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.
[18] C.J. Hillar, L.-H. Lim, Most tensor problems are NP-hard, J. ACM 60 (6) (2013) 1–39.
[19] 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.
[20] T.-Y. Ji, X.-L. Zhao, D.-L. Sun, Low-rank tensor completion method for implicitly low-rank visual data, IEEE Signal Process. Lett. 29 (2022) 1162–1166.
[21] M. Malek-Mohammadi, M. Babaie-Zadeh, M. Skoglund, Performance guarantees for Schatten-p quasi-norm minimization in recovery of low-rank matrices, Signal Process. 114 (2015) 225–230.
[22] Y. Xu, R. Hao, W. Yin, Z. Su, Parallel matrix factorization for low-rank tensor completion, Inverse Probl. Imaging 9 (2) (2015) 601–624.
[23] L.I. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, Phys. D, Nonlinear Phenom. 60 (1–4) (1992) 259–268.
[24] N. Kumar, M. Sonkar, G. Bhatnagar, Efficient image restoration via non-convex total variation regularization and ADMM optimization, Appl. Math. Model. 132 (2024) 428–453.
[25] Z. Xu, J.-H. Yang, C.-l. Wang, F. Wang, X.-h. Yan, Tensor robust principal component analysis with total generalized variation for high-dimensional data recovery, Appl. Math. Comput. 483 (2024) 128980.
[26] Y. Wang, J. Yang, W. Yin, Y. Zhang, A new alternating minimization algorithm for total variation image reconstruction, SIAM J. Imaging Sci. 1 (3) (2008) 248–272.

[27] X.-L. Zhao, F. Wang, T.-Z. Huang, M.K. Ng, R.J. Plemmons, Deblurring and sparse unmixing for hyperspectral images, IEEE Trans. Geosci. Remote Sens. 51 (7) (2013) 4045–4058.
[28] T.-Y. Ji, T.-Z. Huang, X.-L. Zhao, T.-H. Ma, G. Liu, Tensor completion using total variation and low-rank matrix factorization, Inf. Sci. 326 (2016) 243–257.
[29] X. Li, Y. Ye, X. Xu, Low-rank tensor completion with total variation for visual data inpainting, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 31, 2017.
[30] T. Yokota, H. Hontani, Simultaneous visual data completion and denoising based on tensor rank and total variation minimization and its primal-dual splitting algorithm, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 3732–3740.
[31] Y. Chen, T. Xu, X. Zhao, H. Zeng, Y. Xu, J. Chen, Asymmetry total variation and framelet regularized nonconvex low-rank tensor completion, Signal Process. 206 (2023) 108901.
[32] Z. Zhang, K. Chen, K. Tang, Y. Duan, Fast multi-grid methods for minimizing curvature energies, IEEE Trans. Image Process. 32 (2023) 1716–1731.
[33] L. Tan, W. Liu, Z. Pan, Color image restoration and inpainting via multi-channel total curvature, Appl. Math. Model. 61 (2018) 280–299.
[34] Z. Liu, B. Sun, X.-C. Tai, Q. Wang, H. Chang, A fast minimization algorithm for the Euler elastica model based on a bilinear decomposition, SIAM J. Sci. Comput. 46 (1) (2024) A290–A312.
[35] C. Wang, Z. Zhang, Z. Guo, T. Zeng, Y. Duan, Efficient SAV algorithms for curvature minimization problems, IEEE Trans. Circuits Syst. Video Technol. 33 (4) (2022) 1624–1642.
[36] W. Zhu, T. Chan, Image denoising using mean curvature of image surface, SIAM J. Imaging Sci. 5 (1) (2012) 1–32.
[37] Y. Gong, O. Goksel, Weighted mean curvature, Signal Process. 164 (2019) 329–339.
[38] C. Brito-Loeza, K. Chen, V. Uc-Cetina, Image denoising using the Gaussian curvature of the image surface, Numer. Methods Partial Differ. Equ. 32 (3) (2016) 1066–1089.
[39] B. Dong, H. Ju, Y. Lu, Z. Shi, Cure: curvature regularization for missing data recovery, SIAM J. Imaging Sci. 13 (4) (2020) 2169–2188.
[40] Q. Zhong, Y. Li, Y. Yang, Y. Duan, Minimizing discrete total curvature for image processing, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 9474–9482.
[41] J. Bolte, S. Sabach, M. Teboulle, Proximal alternating linearized minimization for nonconvex and nonsmooth problems, Math. Program. 146 (1) (2014) 459–494.
[42] M. Hong, Z.-Q. Luo, M. Razaviyayn, Convergence analysis of alternating direction method of multipliers for a family of nonconvex problems, SIAM J. Optim. 26 (1) (2016) 337–364.
[43] J.-H. Yang, Y. Zhou, L. Zhang, H.-C. Li, Mixed-noise robust tensor multi-view clustering via adaptive dictionary learning, Inf. Fusion (2025) 103322.
[44] M. Ding, T.-Z. Huang, T.-Y. Ji, X.-L. Zhao, J.-H. Yang, Low-rank tensor completion using matrix factorization based on tensor train rank and total variation, J. Sci. Comput. 81 (2019) 941–964.
[45] H. Attouch, J. Bolte, B.F. Svaiter, Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized Gauss–Seidel methods, Math. Program. 137 (1) (2013) 91–129.
[46] J. Bolte, A. Daniilidis, A. Lewis, M. Shiota, Clarke subgradients of stratifiable functions, SIAM J. Optim. 18 (2) (2007) 556–572.
[47] H. Attouch, J. Bolte, P. Redont, A. Soubeyran, Proximal alternating minimization and projection methods for nonconvex problems: an approach based on the Kurdyka-Łojasiewicz inequality, Math. Oper. Res. 35 (2) (2010) 438–457.
[48] 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.