Robust High-Order Tensor Compressive Sensing Based on M-estimators

Xiaowei Wang[1,2], **Jie Yu**[1,2], **Yulong Wang**[1,2,3*]

[1] Engineering Research Center of Intelligent Technology for Agriculture, Ministry of Education, China
[2] College of Informatics, Huazhong Agricultural University, China
[3] Key Laboratory of Smart Farming Technology for Agricultural Animals, Ministry of Agriculture and Rural Affairs, China
{xiaoweiw742, jieyu.gm, wangyulong6251}@gmail.com

Abstract

Tensor Compressive Sensing (TCS) has gained significant attention recently due to its strong ability to preserve the multidimensional structure of data. However, existing TCS methods face three critical challenges: 1) Biased approximation of tensor rank imposed by the convex surrogate Tensor Nuclear Norm (TNN) may interfere with the original low-rank structure of tensor data. 2) Vulnerability to non-Gaussian noise and outliers makes TCS methods highly susceptible to complex noise environments ubiquitous in real-world applications. 3) Most of them are confined to third-order tensors and cannot handle high-order tensor data effectively. Being aware of these, we propose Robust Tensor Compressive Sensing (RTCS) based on M-estimators with three key innovations: 1) We design a novel M-estimator-based low-rank regularizer for high-order tensors, which provides a superior approximation of tensor rank and better preserves the original data structure. 2) RTCS incorporates a robust Welsch estimator that adaptively mitigates the influence of complex noises and outliers in tensor recovery. 3) RTCS is developed to handle high-order tensors, thereby allowing for broader applicability beyond conventional third-order tensors. We further design an efficient algorithm based on the Alternating Direction Method of Multipliers (ADMM) to handle the complex optimization problem. Experiments show that RTCS consistently outperforms existing approaches across various noises.

Introduction

Compressive sensing (CS) recovers the original data from a small number of measurements, offering high efficiency and reduced computational costs (Donoho 2006). The goal of CS is to recover $\mathbf{X} \in \mathbb{R}^{n_1 \times n_2}$ from $\mathbf{y} = \mathcal{A}_M(\mathbf{X}) + \mathbf{e}$, where $\mathcal{A}_M(\cdot)$ is a sampling operator, $\mathbf{y} \in \mathbb{R}^m$ is the measurement vector and $\mathbf{e} \in \mathbb{R}^m$ represents noise. However, most existing CS methods are limited to 2D (matrix) or 1D (vector) data, while real-world applications often involve high-order tensors, such as in hyperspectral imaging, medical imaging and video surveillance. To handle tensor data, one needs to reshape the tensor into matrices in advance, which results in loss of the original tensor structure and potential performance degradation.

To address this, tensor compressive sensing (TCS) extends CS to tensors, preserving their structure and making it suitable for broader applications (Hsu et al. 2024; Sun et al. 2023). The TCS model is given by

$$\mathbf{y} = \mathcal{A}(\mathcal{X}) + \mathbf{e}, \tag{1}$$

where $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}$ and $\mathcal{A}$: $\mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d} \to \mathbb{R}^m$ denotes a linear sampling operator. For the ill-posed inverse problem, existing methods typically need to incorporate priors such as global low-rankness and local smoothness.

The low-rank prior suggests that tensor data has an intrinsic low-dimensional structure. Various methods have been proposed to define different types of tensor ranks, including the Tucker rank (Tucker 1966), CANDECOMP/PARAFAC (CP) rank (Kolda and Bader 2009), and tensor tubal rank based on tensor Singular Value Decomposition (t-SVD) (Misha et al. 2011). Within the t-SVD framework, the tensor nuclear norm (TNN) (Lu et al. 2016) serves as the tightest convex relaxation of the tensor average rank. Given its theoretical advantages, TNN is widely used in tensor learning (Wang et al. 2023b), leading to the low-rank TCS model

$$\min_{\mathcal{X}} \|\mathbf{y} - \mathcal{A}(\mathcal{X})\|_2^2 + \lambda \|\mathcal{X}\|_{\circledast}, \tag{2}$$

where $\|\mathcal{X}\|_{\circledast}$ denotes the TNN of $\mathcal{X}$ and λ is a parameter. This bias arises from TNN's uniform shrinkage of all singular values without considering their relative importance to the underlying tensor structure. Since larger singular values generally encode crucial information like color components in images (Gao et al. 2021), TNN's loose and biased approximation of tensor rank may fail to effectively exploit the low-rank prior (Zhang and Peng 2019). Consequently, various improvements have been proposed to tackle this limitation (Sun et al. 2020; Liu et al. 2024). However, these methods rely on a single low-rank prior and are designed for third-order tensors, limiting the model's generalizability and practical applicability.

Besides the low-rank prior, the local smoothness prior assumes smooth variations between adjacent pixels in the image. While most methods treat low-rank and local smoothness priors as independent regularization terms (Qiu et al. 2021; Xu et al. 2024), (Wang et al. 2023a) propose the TCTV norm to unify both priors in a single term. They theoretically demonstrated that the TCTV norm can leverage both types of prior information, achieving favorable results

*Corresponding author.

for TCS. Subsequently, (Huang et al. 2024) introduced a reweighted TCTV model (RTCTV) to further exploit the intrinsic structural properties of tensors.

Despite the decent performance of these TCS methods, they encounter several limitations: 1) **Biased Tensor Rank Approximation**. Despite numerous enhancements to the convex relaxation-based TNN for TCS problem, these methods often introduce biased approximations that may not fully preserve the underlying low-rank structure of the data. Developing a non-convex tensor rank approximation function that more effectively preserves the intrinsic structure of the original data remains a critical open challenge. 2) **Vulnerability to noise**. Current TCS methods primarily utilize Mean Squared Error (MSE) as the loss function, but MSE depends on the Gaussian noise assumption (Erdogmus and Principe 2002), making it sensitive to non-Gaussian noise and outliers in real-world scenarios. 3) **Limited to third-order tensors**. Existing TCS methods are primarily designed for processing third-order tensors, making them inapplicable to high-order tensors.

Despite extensive research efforts, no existing tensor compressive sensing approach has been developed to holistically solve these problems. The main contributions of this paper can be summarized as follows:

1. We propose a novel M-estimator-based Adaptive Tensor Rank Approximation (MATRA) function, which introduces an innovative **significance-aware non-convex tensor rank approximation mechanism**. This mechanism dynamically adapts penalization based on the importance of tensor components, effectively reducing the approximation bias inherent in convex surrogates and preserving the original data structure. Furthermore, MATRA leverages gradient tensors to capture local smoothness information. This enables the method to exploit both low-rank and local smoothness priors for superior recovery performance.
2. Building upon the MATRA, we propose **Robust Tensor Compressive Sensing (RTCS)**, a novel methodology for high-order tensor recovery. RTCS is designed to integrate MATRA's powerful low-rank approximation with a robust Welsch estimator, which effectively handles non-Gaussian noise encountered in real-world scenarios. Furthermore, RTCS is further extended to **process order-d ($d \geq 3$) tensors**, offering broad applicability.
3. The non-convex and coupled nature of RTCS poses significant challenges for traditional optimization approaches. We devise an **efficient optimization algorithm** by integrating the half-quadratic theory and the ADMM (Boyd et al. 2011) framework to overcome these challenges. Extensive experiments demonstrate that RTCS consistently outperforms competitors, particularly in the presence of 0.1% impulse noise (Fig. 1).

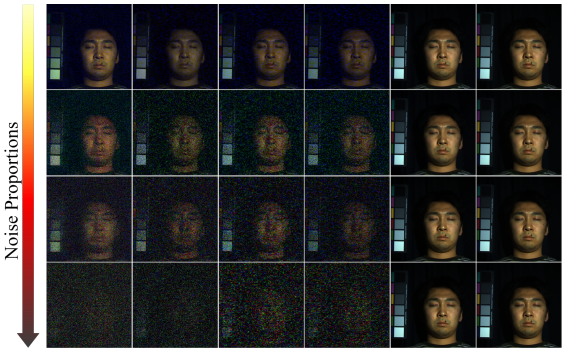


Figure 1: The recovery performance of all competing methods on multispectral images with 20% sampling rate. From top to bottom: impulse noise with noise proportions at 0.002%, 0.005%, 0.01%, and 0.1%. Recovery results of competing methods: (a) TNN (Lu et al. 2020), (b) E3DTV (Peng et al. 2020), (c) TCTV (Wang et al. 2023a), (d) RTCTV (Huang et al. 2024) and (e) **RTCS (ours)**. In the presence of impulse noise, our method consistently demonstrates superior performance in data recovery.

Notations and Preliminaries

To enhance readability, we introduce the key notations used in this paper and the definitions related to tensor operations. We denote scalars, vectors, matrices and tensors in light (e.g., x), bold lower case letters(e.g., $\mathbf{x}$), bold upper case letters (e.g., $\mathbf{X}$) and upper cursive letters (e.g., $\mathcal{X}$), respectively. For a high-order tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}$, we denote its $(j_1, j_2, \cdots, j_n)$-th entry by $\mathcal{X}(j_1, j_2, \cdots, j_n)$, where each index j_i ranges over $1, \ldots, n_i$ for $i = 1, \ldots, d$. $\mathcal{X}(:, :, j_3, \cdots, j_n)$ represents the frontal slice at indices j_3 to j_n. To simplify notation, this slice is denoted as $\mathcal{X}(:, :, \mathbf{j})$, where the index vector $\mathbf{j} = (j_3, \ldots, j_d)$ ranges over all indices in the set $\mathcal{J} = \{1, \ldots, n_3\} \times \cdots \times \{1, \ldots, n_d\}$. Let $\bar{\mathcal{X}}$ denote the Discrete Fourier Transform (DFT) of $\mathcal{X}$ along mode-3 to mode-d, i.e., $\bar{\mathcal{X}} = fft(\mathcal{X}, [], p)$ for $p = 3, \cdots, d$.

Definition 1 (T-product (Qin et al. 2022))**.** *Given tensors $\mathcal{X} \in \mathbb{R}^{n_1 \times l \times n_3 \cdots \times n_d}$ and $\mathcal{Y} \in \mathbb{R}^{l \times n_2 \times n_3 \cdots \times n_d}$, their T-product can be calculated by $\mathcal{X} * \mathcal{Y} = fold(circ(\mathcal{X}) * unfold(\mathcal{Y}))$.*

Using DFT, the T-product $\mathcal{Z} = \mathcal{X} * \mathcal{Y}$ can be efficiently computed by transforming high-order tensor multiplication into a matrix multiplication problem in the Fourier domain, expressed as $bdiag(\bar{\mathcal{Z}}) = bdiag(\bar{\mathcal{X}}) * bdiag(\bar{\mathcal{Y}})$. Here, $bdiag(\bar{\mathcal{X}})$ denotes the block diagonal matrix composed of $\mathcal{X}$.

Definition 2 (Order-d t-SVD (Qin et al. 2022))**.** *The tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \cdots \times n_d}$ can be factorized as $\mathcal{X} = \mathcal{U} * \mathcal{S} * \mathcal{V}^T$, where $\mathcal{U} \in \mathbb{R}^{n_1 \times n_1 \times n_3 \times \cdots \times n_d}$, $\mathcal{V} \in \mathbb{R}^{n_2 \times n_2 \times n_3 \times \cdots \times n_d}$ are orthogonal tensors, and $\mathcal{S} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}$ is a f-diagonal tensor.*

Definition 3 (TNN (Qin et al. 2022))**.** *For a tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}$, its t-SVD is $\mathcal{X} = \mathcal{U} * \mathcal{S} * \mathcal{V}^T$, the TNN of $\mathcal{X}$ is defined as*

$$\|\mathcal{X}\|_{\circledast} = \frac{1}{\nu} \sum_{\mathbf{j} \in \mathcal{J}} \|\bar{\mathcal{X}}(:, :, \mathbf{j})\|_*, \tag{3}$$

where $\|\cdot\|_*$ is the matrix nuclear norm, $\nu = \prod_{p=3}^{d} n_p$ is the scale factor of the DFT transform and $\sum_{\mathbf{j} \in \mathcal{J}}$ denotes $\sum_{j_3=1}^{n_3} \cdots \sum_{j_d=1}^{n_d}$.

Definition 4 (Gradient Tensor (Wang et al. 2023a))**.** *For $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \cdots \times n_d}$, its gradient tensor along the γ-th mode is defined*

as

$$\nabla_\gamma \mathcal{X} = \mathcal{X} \times_\gamma D_{n_\gamma}, \quad \gamma = 1, 2, \cdots, d, \tag{4}$$

where $\times_\gamma$ denotes the mode-γ product (Kolda and Bader 2009) and D_{n_γ} is a row circulant matrix of $(-1, 1, 0, \ldots, 0)$.

Proposed Method

Motivation

Although many studies based on TCS have shown strong development, three challenges remain:

(1) Biased Tensor Rank Approximation. As a widely adopted convex surrogate for tensor rank, TNN provides a loose and biased approximation. This approximation may not effectively preserve the underlying low-rank structure of the data, which limits the model's performance. Fig. 2 shows TNN's approximation curve significantly deviating from the true rank, demonstrating its limitations in capturing low-rank structure information. Despite various proposed improvements (Liu et al. 2024, 2023; Huang et al. 2024), they still do not adequately preserve original tenor structures.

(2) Vulnerability to Non-Gaussian Noise. Existing TCS methods often use Mean Squared Error (MSE) as the loss function, which relies on the Gaussian noise assumption and may fail with non-Gaussian noise (e.g., heavy-tailed distributions) (Erdogmus and Principe 2002). This makes these TCS methods highly vulnerable to complex noise in real-world applications.

(3) Limited to Third-Order Tensors. Most existing TCS methods are specifically designed for third-order tensors, which significantly limits their applicability to higher-dimensional real-world data.

Based on the above analysis, we propose a Robust Tensor Compressive Sensing (RTCS) method based on M-estimator for high-order tensors. RTCS incorporates three key contributions: **1)** We develop an M-estimator-based Adaptive Tensor Rank Approximation (MATRA) function that effectively exploits inherent structural information of tensors. The MATRA employs a significance-aware mechanism that dynamically adjusts penalization according to the importance of tensor components, thereby preserving critical structural information while suppressing less significant elements. **2)** We employ a robust Welsch estimator that effectively mitigates the impact of non-Gaussian noise and outliers. **3)** RTCS extends beyond conventional third-order limitations to process arbitrary-order tensors with enhanced robustness.

M-estimator-based Adaptive Tensor Rank Approximation

Since TNN may over-penalize the larger singular values, leading to biased tensor rank approximation, discarding critical information from dominant singular values (Gao et al. 2021). Effective tensor rank approximation requires appropriate treatment of the tensor: applying smaller penalties to larger ones to preserve salient information, and larger penalties to smaller ones. However, achieving such differential penalization within a convex framework is challenging, motivating our non-convex surrogate approach.

To achieve this, we consider the Welsch function from M-estimators. M-estimators are robust parametric estimation methods in statistics that effectively reduce biases from data contamination in real-world applications (Huber 2011). The Welsch function (De Menezes et al. 2021) is defined as

$$\rho(x) = \frac{\tau^2}{2}\left(1 - \exp\left(-\frac{x^2}{\tau^2}\right)\right), \tag{5}$$

where τ is a scale parameter. However, such replacement is inappropriate for tensor rank approximation, as its exponential decay may be too aggressive for preserving important singular values. This motivates us to propose an improved Welsch function for tensor rank approximation, defined as

$$f_W(x) = b\left(1 - \exp\left(-\frac{|x|}{b}\right)\right), \tag{6}$$

where b is a positive constant. As depicted in Fig. 2, our proposed $f_W(x)$ demonstrates superior performance in approximating the true rank and preserves low-rank structures more effectively than both TNN and existing non-convex surrogate functions (MCP (Jiang, Xu, and Liu 2024), Geman (Tian and Zhang 2022), MLCP (Zhang et al. 2023)).

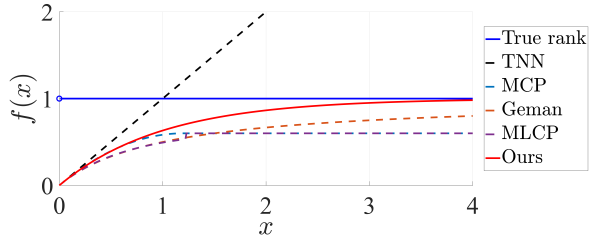


Figure 2: Comparison of different approximation functions with the same parameters.

To effectively exploit both low-rank and local smoothness priors, we propose the M-estimator-based Adaptive Tensor Rank Approximation (MATRA) function for high-order TCS problems. MATRA simultaneously leverages gradient tensors (see Def. 4) for capturing local smoothness information and incorporates the improved Welsch function for adaptive tensor rank approximation. **MATRA** is defined as

$$\mathcal{R}_{MATRA}(\mathcal{X}) = \sum_{\gamma \in \Gamma} \alpha_\gamma F_W(\mathcal{Z}_\gamma), \tag{7}$$

where $\mathcal{Z}_\gamma = \nabla_\gamma \mathcal{X}$ is the gradient tensor along the γ-th mode and Γ is a prior set constituted by the local smooth prior directions of $\mathcal{X}$. The parameters $\{\alpha_\gamma\}_{\gamma \in \Gamma}$ control the penalty intensity for each mode, reflecting the relative importance of smoothness in different directions (Huang et al. 2024). In our experiments, all α_γ are set to 0.5 for simplicity. Let $n = \min\{n_1, n_2\}$, the $F_W(\mathcal{Z}_\gamma)$ is formulated as

$$F_W(\mathcal{Z}_\gamma) = \frac{1}{\nu} \sum_{\mathbf{j} \in \mathcal{J}} \sum_{k=1}^{n} f_W\left(\sigma_k\left(\bar{\mathcal{Z}}_\gamma(:, :, \mathbf{j})\right)\right), \tag{8}$$

where f_W is defined in Eq. (6), b is adaptively set as $b = c_r * \text{mean}(\sigma_1(\bar{\mathcal{Z}}_\gamma(:, :, \mathbf{j}), \ldots, \sigma_n(\bar{\mathcal{Z}}_\gamma(:, :, \mathbf{j}))$, and c_r is a parameter detailed in the experimental section. Here, $\nu = \prod_{p=3}^{d} n_p$

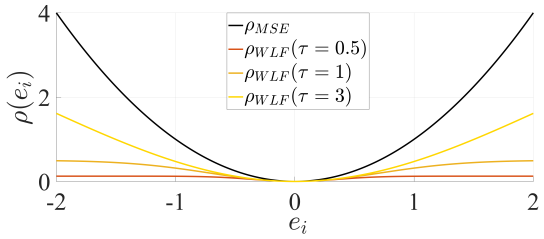


Figure 3: Comparison of robustness between $\rho_{MSE}(e_i) = e_i^2$ and $\rho_{WLF}(e_i) = \frac{\tau^2}{2}\left(1 - \exp(-\frac{e_i^2}{\tau^2})\right)$ with different τ.

is the DFT scale factor and $\sigma_k\left(\bar{\mathcal{Z}}_\gamma(:,:,\mathbf{j})\right)$ denotes the k-th singular value of the matrix slice of $\bar{\mathcal{Z}}_\gamma$ indexed by $\mathbf{j}$.

According to Eq. (8), each singular value $\sigma_k\left(\bar{\mathcal{Z}}_\gamma(:,:,\mathbf{j})\right)$ corresponds to a penalty of varying magnitude. This adaptive penalty mechanism allows MATRA to apply stronger penalties to smaller singular values while preserving larger ones, thereby achieving a more accurate low-rank approximation compared to uniform penalty-based approaches.

Robust Tensor Compressive Sensing (RTCS)

While MATRA function effectively mitigates tensor rank approximation bias, real-world data contains non-Gaussian noise and outliers. However, conventional TCS methods typically rely on MSE-based loss functions, which assume a Gaussian noise distribution (Erdogmus and Principe 2002) and may not perform well against non-Gaussian conditions.

To enhance the robustness of our method against such data corruptions, we extend the Welsch estimator introduced in Eq. (5) to formulate the Welsch Loss Function (WLF)

$$\mathcal{L}_{WLF}(\mathbf{e}) = \sum_{i=1}^{m} \rho(e_i), \tag{9}$$

where $\mathbf{e} = \mathbf{y} - \mathcal{A}(\mathcal{X}) \in \mathbb{R}^m$ is the residual vector, with e_i denoting its i-th element. Here, τ is adaptively set as $\tau = c_l \cdot \text{mean}(|\mathbf{e}|)$, where c_l is a parameter detailed in the experimental section. Fig. 3 shows that MSE quadratically amplifies large residuals, making it sensitive to non-Gaussian and outliers. Conversely, WLF mitigates the impact of large residuals on the loss.

By integrating WLF and the MATRA function into a joint framework, we propose the Robust TCS (RTCS) model

$$\min_{\mathcal{X}\in\mathbb{R}^{n_1\times n_2\times\cdots\times n_d}} \mathcal{L}_{WLF}(\mathbf{y} - \mathcal{A}(\mathcal{X})) + \lambda\, \mathcal{R}_{MATRA}(\mathcal{X}). \tag{10}$$

Remark 1. *The RTCS model integrates three key innovations to address the fundamental challenges in TCS. First, the $\mathcal{R}_{MATRA}(\cdot)$ incorporates a significance-aware penalty mechanism that adaptively penalizes singular values, thereby preserving dominant components while exploiting low-rank and local smoothness priors. Second, the $\mathcal{L}_{WLF}(\cdot)$ enhances the model's robustness against non-Gaussian noise and outliers, overcoming the limitations of MSE-based methods. Third, RTCS establishes a comprehensive, robust tensor compressive sensing approach that can efficiently handle high-order tensors in complex noise.*

Optimization

It is pointing out that the RTCS model for high-order tensors is complex to solve using traditional optimization methods due to the non-convex and coupled nature of $\rho(e)$ and $f_W(x)$. To overcome this challenge, we have devised an efficient optimization algorithm that integrates the ADMM framework with half-quadratic theory (He et al. 2014). We first introduce a useful lemma in light of the half-quadratic theory.

Lemma 1 (He et al. 2014)**.** *For the Welsch loss $\rho(e) = \frac{\tau^2}{2}\left(1 - \exp\left(-\frac{e^2}{\tau^2}\right)\right)$, there exists a dual function $\phi(w)$, $w \in \mathbb{R}$ such that*

$$\rho(e) = \min_{w}\left(\frac{1}{2}we^2 + \phi(w)\right), \tag{11}$$

where the minimum is reached at $w = \exp\left(-\frac{e^2}{\tau^2}\right)$.

According to Lemma 1, we have

$$\mathcal{L}_{WLF}(\mathbf{y} - \mathcal{A}(\mathcal{X})) = \min_{\mathbf{w}} \frac{1}{2}\left\|\mathbf{w}^{\frac{1}{2}} \otimes (\mathbf{y} - \mathcal{A}(\mathcal{X}))\right\|_2^2 + \Phi(\mathbf{w}), \tag{12}$$

where $\Phi(\mathbf{w}) = \sum_{i=1}^{m}\phi(w_i)$, $\otimes$ denotes the Hadamard (element-wise) product and the minimum is reached at

$$w_i = \exp\left(-\frac{(\mathbf{y}_i - (\mathcal{A}(\mathcal{X}))_i)^2}{\tau^2}\right). \tag{13}$$

Lemma 1 enables efficient dual optimization of $\rho(e)$. Similarly, we derive the dual form of $f_W(x)$ as Lemma 2.

Proof. The proof is provided in supplementary material. □

Lemma 2. *For the function $f_W(x) = b\left(1 - \exp(-\frac{|x|}{b})\right)$, there exists a dual function $\psi(W)$, $W \in \mathbb{R}$ such that*

$$f_W(x) = \min_{W}\left(Wx + \psi(W)\right), \tag{14}$$

where the minimum is reached at $W = \exp\left(-\frac{|x|}{b}\right)$.

According to Lemma 2, Eq. (8) can be transformed into

$$F_W(\mathcal{Z}_\gamma) = \min_{\mathcal{W}} \frac{1}{\nu}\sum_{\mathbf{j}\in\mathcal{J}}\sum_{k=1}^{n} \mathcal{W}_{\mathbf{j}k,\gamma}\sigma_k\left(\bar{\mathcal{Z}}_\gamma(:,:,\mathbf{j})\right) + \Psi(\mathcal{W}_\gamma), \tag{15}$$

where $\Psi(\mathcal{W}_\gamma) = \sum_{\mathbf{j}\in\mathcal{J}}\sum_{k=1}^{n}\psi(\mathcal{W}_{\mathbf{j}k,\gamma})$ and the minimum is reached at

$$\mathcal{W}_{\mathbf{j}k,\gamma} = \exp\left(-\frac{\sigma_k(\bar{\mathcal{Z}}_\gamma(:,:,\mathbf{j}))}{b}\right). \tag{16}$$

Here, $\mathcal{W}_{\mathbf{j}k,\gamma}$ represents the weight of $\sigma_k(\bar{\mathcal{Z}}_\gamma(:,:,\mathbf{j}))$. For convenience, we denote

$$\mathcal{G}_{\mathcal{W}}(\mathcal{Z}_\gamma) = \frac{1}{\nu}\sum_{\mathbf{j}\in\mathcal{J}}\sum_{k=1}^{n}\mathcal{W}_{\mathbf{j}k,\gamma}\sigma_k\left(\bar{\mathcal{Z}}_\gamma(:,:,\mathbf{j})\right). \tag{17}$$

Notably, $\mathcal{G}_w(\mathcal{Z}_\gamma)$ represents a generalized weighted formulation for high-order tensor singular values of tensor $\mathcal{Z}_\gamma$.

Therefore, based on Eq. (12) and Eq. (15), problem (10) can be reformulated into an equivalent weighted problem by introducing auxiliary variables $\mathbf{w}$ and $\mathcal{W}$, as follows:

$$\min_{\mathcal{X},\mathbf{w},\{\mathcal{W}_\gamma\}_{\gamma\in\Gamma}} \frac{1}{2}\left\|\mathbf{w}^{\frac{1}{2}}\otimes(\mathbf{y}-\mathcal{A}(\mathcal{X}))\right\|_2^2+\Phi(\mathbf{w}) + \lambda\sum_{\gamma\in\Gamma}\alpha_\gamma\left(\mathcal{G}_{\mathcal{W}}(\mathcal{Z}_\gamma)+\Psi(\mathcal{W}_\gamma)\right),\ \text{s.t. } \nabla_\gamma\mathcal{X}=\mathcal{Z}_\gamma,\gamma\in\Gamma. \tag{18}$$

Then, model (18) can be efficiently solved using the ADMM framework (Boyd et al. 2011). The augmented Lagrangian function is

$$\mathcal{L}(\mathcal{X},\left\{\mathcal{Z}_\gamma,\mathcal{W}_\gamma,\mathcal{D}_\gamma,\gamma\in\Gamma\right\},\mathbf{w},\mu) = \frac{1}{2}\left\|\mathbf{w}^{\frac{1}{2}}\otimes(\mathbf{y}-\mathcal{A}(\mathcal{X}))\right\|_2^2+\Phi(\mathbf{w})+\lambda\sum_{\gamma\in\Gamma}\alpha_\gamma\big(\mathcal{G}_{\mathcal{W}}(\mathcal{Z}_\gamma) + \Psi(\mathcal{W}_\gamma)\big)+\frac{\mu}{2}\sum_{\gamma\in\Gamma}\left\|\mathcal{Z}_\gamma-\nabla_\gamma\mathcal{X}+\frac{\mathcal{D}_\gamma}{\mu}\right\|_F^2+C, \tag{19}$$

where $\mathcal{D}_\gamma$ denotes the Lagrange multipliers and C represents the squared term that is dependent only on the multipliers. We minimize Eq. (19) iteratively by fixing each variable and adjusting the others. Based on the ADMM framework, the subproblems are updated as follows:

Step 1. Update $\mathcal{X}$ by solving the optimization problem:

$$\min_{\mathcal{X}}\frac{1}{2}\|(\mathbf{w}^t)^{\frac{1}{2}}\otimes(\mathbf{y}-\mathcal{A}(\mathcal{X}))\|_2^2+\frac{\mu^t}{2}\sum_{\gamma\in\Gamma}\left\|\mathcal{Z}_\gamma^t-\nabla_\gamma\mathcal{X}+\frac{\mathcal{D}_\gamma}{\mu^t}\right\|_F^2.$$

By setting the derivative with respect to $\mathcal{X}$ to zero and $\mathbf{W}^t = \text{diag}((\mathbf{w}^t)^{\frac{1}{2}})$, we have

$$\left(\mathcal{A}^T\left((\mathbf{W}^t)^T\mathbf{W}^t\right)\mathcal{A}+\mu^t\sum_{\gamma\in\Gamma}\nabla_\gamma^T\nabla_\gamma\right)(\mathcal{X}) = \mathcal{A}^T\left((\mathbf{W}^t)^T\mathbf{W}^t\right)\mathbf{y}+\sum_{\gamma\in\Gamma}\nabla_\gamma^T\left(\mu^t\mathcal{Z}_\gamma^t+\mathcal{D}_\gamma^t\right), \tag{20}$$

where $\nabla_\gamma^T(\cdot)$ represents the transposed resulting operator of $\nabla_\gamma(\cdot)$. Eq. (20) can be solved by off-the-shelf techniques such as the preconditioned conjugate gradient method (Axelsson and Lindskog 1986).

Step 2. Update $\mathcal{Z}_\gamma$ by

$$\mathcal{Z}_\gamma^{t+1}=\arg\min_{\mathcal{Z}_\gamma}\lambda\,\alpha_\gamma\mathcal{G}_{\mathcal{W}}(\mathcal{Z}_\gamma)+\frac{\mu^t}{2}\left\|\mathcal{Z}_\gamma-\nabla_\gamma\mathcal{X}^{t+1}+\frac{\mathcal{D}_\gamma^t}{\mu^t}\right\|_F^2. \tag{21}$$

Inspired by (Wang et al. 2023c) for order-3 tensors, we propose the following theorem for solving Eq. (21).

Proof. The proof is provided in supplementary material. □

Theorem 1. *Given a tensor $\mathcal{Y}\in\mathbb{R}^{n_1\times n_2\times\cdots\times n_d}$ and a weight tensor $\mathcal{W}\in\mathbb{R}^{n_3\times n_4\times\cdots\times n_d\times n}$, where $n=\min\{n_1,n_2\}$. Let $\mathcal{Y}=\mathcal{U}*\mathcal{S}*\mathcal{V}^T$ denotes the order-d t-SVD of $\mathcal{Y}$. Consider the following optimization problem*

$$\mathbf{Prox}_{\mathcal{G}_{\mathcal{W}}}(\mathcal{Y})=\arg\min_{\mathcal{X}}\left\{\mathcal{G}_{\mathcal{W}}(\mathcal{X})+\frac{1}{2}\|\mathcal{X}-\mathcal{Y}\|_F^2\right\}, \tag{22}$$

where $\mathbf{Prox}_{\mathcal{G}_{\mathcal{W}}}(\mathcal{Y})$ is the proximal operator and $\mathcal{G}_{\mathcal{W}}(\mathcal{X})=\frac{1}{\nu}\sum_{\mathbf{j}\in\mathcal{J}}\sum_{k=1}^{n}\mathcal{W}_{\mathbf{j}k}\sigma_k\left(\bar{\mathcal{X}}(:,:,\mathbf{j})\right)$. Assuming the weights are ordered as $0\le\mathcal{W}_{\mathbf{j}1}\le\mathcal{W}_{\mathbf{j}2}\le\ldots\le\mathcal{W}_{\mathbf{j}n}$ for index vector $\mathbf{j}=(j_3,\ldots,j_d)$ ranges over all indices in the set $\mathcal{J}=\{1,\ldots,n_3\}\times\cdots\times\{1,\ldots,n_d\}$. The global solution to the problem (22) is given by

$$\mathcal{X}^*=\mathbf{Prox}_{\mathcal{G}_{\mathcal{W}}}(\mathcal{Y})=\mathcal{U}*\mathit{ifft}(\mathcal{P}_{\mathcal{W}}(\bar{\mathcal{S}}),[],p)*\mathcal{V}^T, \tag{23}$$

where $\mathcal{P}_{\mathcal{W}}(\bar{\mathcal{S}})$ is a tensor whose $\mathbf{j}$-th frontal slice is $P_{\mathcal{W}_\mathbf{j}}(\bar{S}(:,:,\mathbf{j}))$. Here, $P_{\mathcal{W}_\mathbf{j}}(\bar{S}(:,:,\mathbf{j}))$ is a diagonal matrix computed as $(P_{\mathcal{W}_\mathbf{j}}(\bar{S}(:,:,\mathbf{j})))_{kk}=((\bar{S}(:,:,\mathbf{j}))_{kk}-\mathcal{W}_{\mathbf{j}k})_+$, where $(x)_+=\max(x,0)$. The element $\mathcal{W}_{\mathbf{j}k}$ denotes the weight for the k-th singular value of the matrix slice at index $\mathbf{j}$.

Referring back to Eq. (17), we have $\frac{\lambda\alpha_\gamma}{\mu_t}\mathcal{G}_{\mathcal{W}}(\mathcal{Z}_\gamma)=\mathcal{G}_{\frac{\lambda\alpha_\gamma}{\mu_t}\mathcal{W}}(\mathcal{Z}_\gamma)$, and the global solution for the Eq. (21) can be expressed as

$$\mathcal{Z}_\gamma^{t+1}=\mathbf{Prox}_{\mathcal{G}_{\frac{\lambda\alpha_\gamma}{\mu_t}\mathcal{W}}}\left(\nabla_\gamma\mathcal{X}^{t+1}-\frac{\mathcal{D}_\gamma^t}{\mu^t}\right),\gamma\in\Gamma. \tag{24}$$

Step 3. Update $\mathbf{w}$ and $\mathcal{W}$ according to Eq. (13) and Eq. (16) using the formulas

$$w_i^{t+1}=\exp\left(-\frac{(e_i^{t+1})^2}{\tau^2}\right),\quad \mathcal{W}_{\mathbf{j}k,\gamma}^{t+1}=\exp\left(-\frac{s_{\mathbf{j}k,\gamma}^{t+1}}{b}\right), \tag{25}$$

where $e_i^{t+1}=\mathbf{y}_i-(\mathcal{A}(\mathcal{X}^{t+1}))_i$ and $s_{\mathbf{j}k,\gamma}^{t+1}=\sigma_k(\bar{\mathcal{Z}}_\gamma^{t+1}(:,:,\mathbf{j}))$.

Algorithm 1 summarizes the optimization algorithm for the RTCS model.

Algorithm 1: Algorithm for the RTCS Model

Input: linear operator $\mathcal{A}$, noisy measurements $\mathbf{y}$.
Initialization: $\mathcal{X}^0,\mathcal{Z}_\gamma^0$ and $\mathcal{D}_\gamma^0$ for $\gamma\in\Gamma$, $\mathbf{w}^0=\mathbf{1}$, $\mathcal{W}^0=\mathbf{1}_{n_3\times n_4\times\cdots\times n_d\times n}$, regularization parameter λ, $\mu^0>0$, tolerate error $\epsilon=10^{-8}$, $\rho=1.1$, $\mu_{\max}=10^6$, maxIter = 200.
Output: $\hat{\mathcal{X}}=\mathcal{X}^{t+1}$.

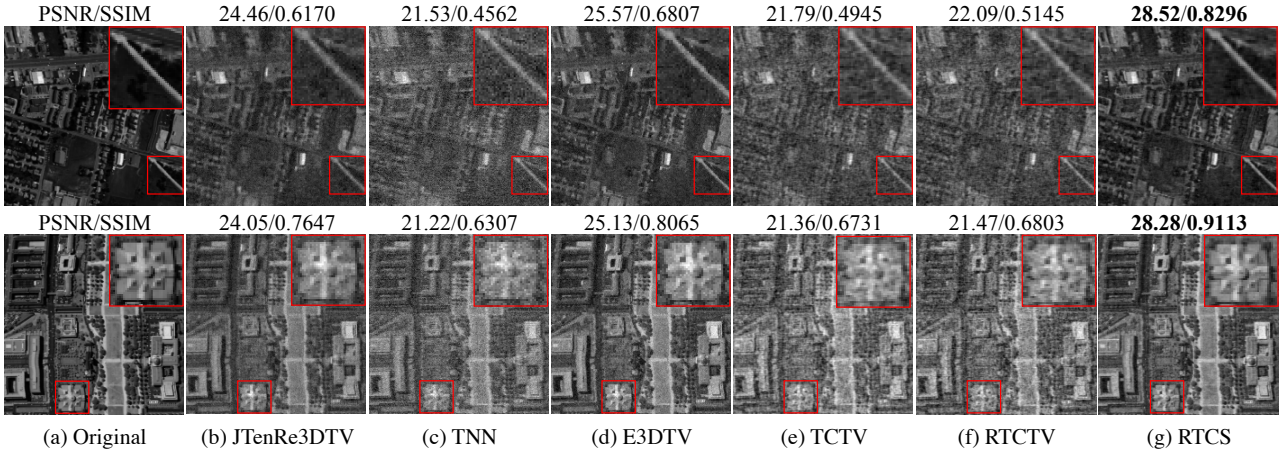
```
while not convergent and t < maxIter do
    Update X^{t+1} by (20);
    Update Z_γ^{t+1} by (24);
    Update w^{t+1} and W^{t+1} by (25);
    D_γ^{t+1} = D_γ^t + μ^t(Z_γ^{t+1} − ∇_γ X^{t+1});
    μ^{t+1} = min{ρμ^t, μ_max};
    Check the convergence criteria
        ||X^{t+1} − X^t||_F / max{||X^t||_F, 1} < ϵ;
end while
```

Complexity Analysis and Convergence Curves

The detailed complexity analysis and empirical convergence curves of RTCS are presented in the supplementary material due to space limitations.

Experiments

In this section, we analyze the performance on different types of data, including hyperspectral images (HSI), multispectral images (MSI), grayscale videos, and other high-order tensors ($d>3$). Due to space limitations, additional experimental results and details for different noise conditions can be found in the supplementary material.



(a) Original (b) JTenRe3DTV (c) TNN (d) E3DTV (e) TCTV (f) RTCTV (g) RTCS

Figure 4: HSI recovery results for the 47-th band of all competing methods with 10% sampling rates, which are corrupted by student's t-distribution noise (Top: Urban; Bottom: DCMall).

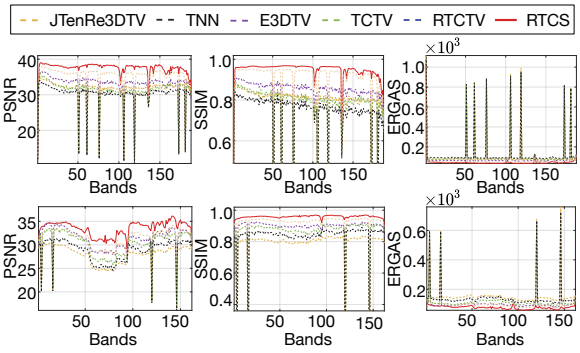


Figure 5: Comparison of PSNR, SSIM and ERGAS across all spectral bands on two HSI datasets at 15% sampling rate with random missing entries (Top: Cuprite; Bottom: Urban).

Setting. We adopt the Walsh-Hadamard sampling strategy $\mathcal{A}(\mathcal{X}) = \mathbf{A}\text{vec}(\mathcal{X})$ with $\mathbf{A} = \mathbf{DFR}$ (Liu et al. 2023), where $\mathbf{R}$, $\mathbf{F}$, and $\mathbf{D}$ represent random permutation, Walsh-Hadamard transform, and random downsampling operators, respectively. Following the parameter setting in (He et al. 2014), both c_r and c_l are set to 0.5 to maintain consistency with previous studies. To simulate data corruption, we consider three types of noise: student's t-distribution noise with 2 degrees of freedom, impulse noise, and random missing entries. We use the noise level to control the intensity of student's t-distribution noise and noise proportion to control the contamination ratio of impulse noise and missing entries.

Comparison methods. We compare the RTCS method with JTenRe3DTV (Wang et al. 2017), TNN, E3DTV, TCTV and RTCTV using the parameters recommended by the authors. For the RTCS method, λ is selected from $\{10^{-3}, 10^{-2}, 10^{-1}, 10^{1}, 10^{2}\}$. This paper evaluates recovery performance using three standard indices: PSNR, SSIM (Wang et al. 2004) and ERGAS (Wald 2002), where ERGAS measures the spectral quality of HSI.

SR	Methods	0.01		0.03	
		PSNR ↑	ERGAS ↓	PSNR ↑	ERGAS ↓
10%	**RTCS**	**36.25**	**90.35**	**34.91**	**115.83**
	RTCTV	33.31	108.52	30.13	171.67
	TCTV	34.19	102.40	29.90	196.97
	E3DTV	31.74	128.18	30.41	180.18
	TNN	31.72	160.38	28.73	228.58
	JTenRe3DTV	29.83	149.53	28.54	191.5
20%	**RTCS**	**43.86**	**60.23**	**40.96**	**75.73**
	RTCTV	34.64	128.32	32.58	170.95
	TCTV	35.50	90.05	30.80	174.13
	E3DTV	34.17	135.62	31.61	189.41
	TNN	33.52	145.51	29.44	181.82
	JTenRe3DTV	30.95	126.14	29.40	200.72

Table 1: Average PSNR and ERGAS values across five HSI datasets at 10% and 20% sampling rates with random missing entries. The best results are in bold.

Application to HSI Compressive Sensing

In this subsection, we evaluate the performance of RTCS on HSI datasets. Specifically, five publicly available hyperspectral datasets are used: DCMall[1], Cuprite[1], Indian Pines[1], Pavia University (PaviaU)[1] and Urban[1]. Figs. 4 and 5 display visual recovery results and quality metrics for each spectral band. Table 1 presents average PSNR and ERGAS metrics across five datasets. The results show that RTCS outperforms competing methods in most spectral bands, with superior noise robustness and detail preservation.

Application to MSI Compressive Sensing

In this subsection, we evaluate the performance of our method on MSI datasets by selecting four samples from the CAVE dataset (Yasuma et al. 2010): Balloons, Beads, Clay, and Face. Columns 3 and 4 of Table 2 show the average

[1]http://lesun.weebly.com/hyperspectral-data-set.html

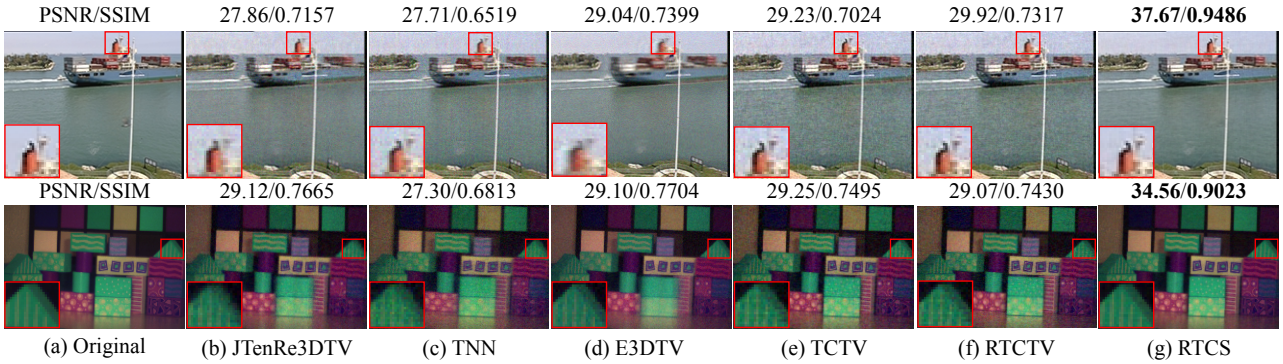


Figure 6: Order-4 tensor recovery results with 20% sampling rate, which are corrupted by student's t-distribution noise. (Top: container 4D dataset of 258-th frame. Bottom: pseudo-color image of the hyperspectral video utilizing bands 7-21-28 as R-G-B)

SR	Methods	MSI		YUV	
		0.01	0.03	0.01	0.03
10%	**RTCS**	**38.29**	**36.36**	**32.23**	**31.82**
	RTCTV	34.44	32.97	29.53	26.74
	TCTV	33.49	31.13	29.43	26.12
	E3DTV	30.08	29.01	29.20	28.04
	TNN	29.15	27.61	29.42	27.08
	JTenRe3DTV	35.37	33.12	27.11	24.39
20%	**RTCS**	**41.85**	**39.62**	**36.75**	**32.66**
	RTCTV	37.84	35.73	31.73	27.18
	TCTV	37.31	33.06	30.53	26.92
	E3DTV	35.91	33.09	30.71	27.75
	TNN	34.77	31.20	29.06	26.72
	JTenRe3DTV	37.09	33.84	28.56	25.01

Table 2: The average PSNR comparison for MSI datasets and YUV datasets at 10% and 20% sampling rates, with random missing entries at 0.01 and 0.03 proportions. Best results are marked bold.

PSNR results. RTCS outperforms other methods across various sampling rates and noise levels, particularly demonstrating strong robustness at higher noise levels.

Application to Video Compressive Sensing

We use four video sequences from the YUV dataset[1] to evaluate the performance of our method: Akiyo, Container, News and Carphone. For efficiency, we selected the first 100 frames from each sequence. Columns 5 and 6 of Table 2 show that RTCS surpasses competing methods in PSNR for the YUV dataset at both 10% and 20% sampling rates, demonstrating superior recovery performance and noise robustness.

Application to High-order Tensor Compressive Sensing

We conducted further experiments on high-order tensor data to assess the generalizability of the RTCS method in compressive sensing. These included an order-4 color video dataset (Mohaoui, Hakim, and Raghay 2022) and an order-4 hyperspectral video dataset (Mian and Hartley 2012). We adapt 3D baselines to 4D data via a standard slice-wise strategy (Qin et al. 2022), processing 3D slices independently and averaging the results. Fig. 6 shows that RTCS achieves higher PSNR and significantly reduced visual noise compared to competing methods in high-order TCS.

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

Methods	$\mathcal{L}_{WLF}(\cdot)$	$\mathcal{R}_{MATRA}(\cdot)$	HSI	MSI	Video
TCTV			30.94	32.75	29.22
RTCS-L	✓		33.48	34.40	30.77
RTCS-R		✓	34.66	35.69	30.31
RTCS	✓	✓	**36.45**	**36.94**	**31.12**

Table 3: Ablation Study. PSNR values of RTCS on different datasets. Best results are marked in bold.

Ablation Study

To evaluate the impact of $\mathcal{L}_{WLF}(\cdot)$ or $\mathcal{R}_{MATRA}(\cdot)$ on recovery performance, we conducted an ablation study with two model variants: RTCS-L ($\mathcal{L}_{WLF}$ + $\mathcal{R}_{TCTV}$) and RTCS-R ($\mathcal{L}_{MSE}$ + $\mathcal{R}_{MATRA}$). Experiments on Cuprite (HSI), Balloons (MSI), and Container (video) with 10% sampling rate and student's t-distribution noise show both $\mathcal{L}_{WLF}(\cdot)$ and $\mathcal{R}_{MATRA}(\cdot)$ improve the performance of TCTV. RTCS integrates both components and consistently outperforms the variants, demonstrating the benefit of the unified framework.

Conclusion

This paper proposes RTCS, which simultaneously addresses three fundamental limitations of existing TCS methods. 1) An M-estimator-based adaptive tensor rank approximation method that applies lighter penalties to large singular values, leading to better preservation of original tensor structures. 2) A robust Welsch estimator that mitigates complex noise and outlier influences in practical scenarios. 3) Extended applicability to high-order tensors beyond conventional third-order limitations. Comprehensive experiments show RTCS's superior performance under various noise conditions.

Acknowledgments

We are grateful to the anonymous AAAI reviewers for their constructive comments. This work was supported in part by National Natural Science Foundation of China under Grant Nos. 62276111, 61806027, 62076041, and 62172458, and in part by the Fundamental Research Funds for the Central Universities under Project 2662023XXPY002, 2662023LXPY005, and SZYJY2023010.

References

Axelsson, O.; and Lindskog, G. 1986. On the rate of convergence of the preconditioned conjugate gradient method. *Numerische Mathematik*, 48: 499–523.

Boyd, S.; Parikh, N.; Chu, E.; Peleato, B.; and Eckstein, J. 2011. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3: 1–122.

De Menezes, D.; Prata, D.; Secchi, A.; and Pinto, J. 2021. A review on robust M-estimators for regression analysis. *Computers & Chemical Engineering*, 147: 107254.

Donoho, D. 2006. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4): 1289–1306.

Erdogmus, D.; and Principe, J. C. 2002. An error-entropy minimization algorithm for supervised training of nonlinear adaptive systems. *IEEE Transactions on Signal Processing*, 50(7): 1780–1786.

Gao, Q.; Zhang, P.; Xia, W.; Xie, D.; Gao, X.; and Tao, D. 2021. Enhanced Tensor RPCA and its Application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(6): 2133–2140.

He, R.; Zheng, W.-S.; Tan, T.; and Sun, Z. 2014. Half-Quadratic-Based Iterative Minimization for Robust Sparse Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(2): 261–275.

Hsu, C.-C.; Jian, C.-Y.; Tu, E.-S.; Lee, C.-M.; and Chen, G.-L. 2024. Real-Time Compressed Sensing for Joint Hyperspectral Image Transmission and Restoration for CubeSat. *IEEE Transactions on Geoscience and Remote Sensing*, 62: 1–16.

Huang, K.; Kong, W.; Zhou, M.; Qin, W.; Zhang, F.; and Wang, J. 2024. Enhanced Low-Rank Tensor Recovery Fusing Reweighted Tensor Correlated Total Variation Regularization for Image Denoising. *Journal of scientific computing*, (3): 99.

Huber, P. J. 2011. *Robust Statistics*, 1248–1251. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN 978-3-642-04898-2.

Jiang, H.; Xu, C.; and Liu, L. 2024. High-Fidelity compressive spectral image reconstruction through a novel Non-Convex Non-Local Low-Rank tensor approximation model. *Optics & Laser Technology*, 171: 110424.

Kolda, T. G.; and Bader, B. W. 2009. Tensor decompositions and applications. *SIAM review*, 51(3): 455–500.

Liu, X.; Hou, J.; Peng, J.; Wang, H.; Meng, D.; and Wang, J. 2023. Tensor Compressive Sensing Fused Low-Rankness and Local-Smoothness. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(7): 8879–8887.

Liu, Y.; Du, B.; Chen, Y.; Zhang, L.; Gong, M.; and Tao, D. 2024. Convex–Concave Tensor Robust Principal Component Analysis. *International Journal of Computer Vision*, 132(5): 1721–1747.

Lu, C.; Feng, J.; Chen, Y.; Liu, W.; Lin, Z.; and Yan, S. 2016. 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*, 5249–5257.

Lu, C.; Feng, J.; Chen, Y.; Liu, W.; Lin, Z.; and Yan, S. 2020. Tensor Robust Principal Component Analysis with a New Tensor Nuclear Norm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(4): 925–938.

Mian, A.; and Hartley, R. 2012. Hyperspectral Video Restoration Using Optical Flow and Sparse Coding. *Optics Express*, 20(10): 10658–10673.

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

Mohaoui, S.; Hakim, A.; and Raghay, S. 2022. Smooth tensor robust principal component analysis with application to color image recovery. *Digital Signal Processing*, 123: 103390.

Peng, J.; Xie, Q.; Zhao, Q.; Wang, Y.; Yee, L.; and Meng, D. 2020. Enhanced 3DTV regularization and its applications on HSI denoising and compressed sensing. *IEEE Transactions on Image Processing*, 29: 7889–7903.

Qin, W.; Wang, H.; Zhang, F.; Wang, J.; Luo, X.; and Huang, T. 2022. Low-Rank High-Order Tensor Completion With Applications in Visual Data. *IEEE Transactions on Image Processing*, 31: 2433–2448.

Qiu, D.; Bai, M.; Ng, M. K.; and Zhang, X. 2021. Robust low-rank tensor completion via transformed tensor nuclear norm with total variation regularization. *Neurocomputing*, 435: 197–215.

Sun, L.; He, C.; Zheng, Y.; Wu, Z.; and Jeon, B. 2023. Tensor Cascaded-Rank Minimization in Subspace: A Unified Regime for Hyperspectral Image Low-Level Vision. *IEEE Transactions on Image Processing*, 32: 100–115.

Sun, L.; Wu, F.; Zhan, T.; Liu, W.; Wang, J.; and Jeon, B. 2020. Weighted Nonlocal Low-Rank Tensor Decomposition Method for Sparse Unmixing of Hyperspectral Images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13: 1174–1188.

Tian, Y.; and Zhang, Y. 2022. A comprehensive survey on regularization strategies in machine learning. *Information Fusion*, 80: 146–166.

Tucker, L. R. 1966. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31(3): 279–311.

Wald, L. 2002. *Data fusion: definitions and architectures: fusion of images of different spatial resolutions*. Presses des MINES.

Wang, H.; Peng, J.; Qin, W.; Wang, J.; and Meng, D. 2023a. Guaranteed Tensor Recovery Fused Low-rankness

and Smoothness. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9): 10990–11007.

Wang, M.; Hong, D.; Han, Z.; Li, J.; Yao, J.; Gao, L.; Zhang, B.; and Chanussot, J. 2023b. Tensor Decompositions for Hyperspectral Data Processing in Remote Sensing: A comprehensive review. *IEEE Geoscience and Remote Sensing Magazine*, 11(1): 26–72.

Wang, Y.; Kou, K. I.; Chen, H.; Tang, Y. Y.; and Li, L. 2023c. Double Auto-Weighted Tensor Robust Principal Component Analysis. *IEEE Transactions on Image Processing*, 32: 5114–5125.

Wang, Y.; Lin, L.; Zhao, Q.; Yue, T.; Meng, D.; and Leung, Y. 2017. Compressive Sensing of Hyperspectral Images via Joint Tensor Tucker Decomposition and Weighted Total Variation Regularization. *IEEE Geoscience and Remote Sensing Letters*, 14(12): 2457–2461.

Wang, Z.; Bovik, A.; Sheikh, H.; and Simoncelli, E. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612.

Xu, S.; Ke, Q.; Peng, J.; Cao, X.; and Zhao, Z. 2024. Pan-Denoising: Guided Hyperspectral Image Denoising via Weighted Represent Coefficient Total Variation. *IEEE Transactions on Geoscience and Remote Sensing*, 62: 1–14.

Yasuma, F.; Mitsunaga, T.; Iso, D.; and Nayar, S. K. 2010. Generalized Assorted Pixel Camera: Postcapture Control of Resolution, Dynamic Range, and Spectrum. *IEEE Transactions on Image Processing*, 19(9): 2241–2253.

Zhang, H.; Fan, H.; Li, Y.; Liu, X.; Liu, C.; and Zhu, X. 2023. Tensor Recovery Based on a Novel Non-Convex Function Minimax Logarithmic Concave Penalty Function. *IEEE Transactions on Image Processing*, 32: 3413–3428.

Zhang, L.; and Peng, Z. 2019. Infrared Small Target Detection Based on Partial Sum of the Tensor Nuclear Norm. *Remote Sensing*, 11(4).