Journal of Industrial and Management Optimization

https://www.aimspress.com/journal/jimo

JIMO, 22(6): 2942–2962.
DOI:10.3934/jimo.2026108
Received: 04 February 2026
Revised: 06 May 2026
Accepted: 15 May 2026
Published: 26 May 2026

Research article

A non-convex tensor RPCA model with TL1 penalty for image restoration

Xiao Guo[1], **Chuanpei Xu**[1,*], **Zhibin Zhu**[2] **and Benxin Zhang**[1]

[1] School of Electronic Engineering and Automation, Guilin University of Electronic Technology, Guilin, 541004 China

[2] School of Mathematics and Computing Science, Guilin University of Electronic Technology, Guilin, 541004, China

* **Correspondence:** Email: xcp@guet.edu.cn.

Abstract: For high-dimensional image restoration, tensor robust principal component analysis (TRPCA) is a highly effective method. However, since it uniformly penalizes all singular values, it tends to over-penalize larger singular values, thereby failing to adequately preserve the primary structural information of the image. In contrast, the non-convex transformed L_1 penalty (TL1) function demonstrates a stronger ability to protect larger singular values and reduce estimation bias, offering a more accurate approximation of the rank function. Inspired by this, we propose a TRPCA with TL1 penalty model (TL1-TRPCA), which non-convexly substitutes the low-rank function with the TL1, and apply the alternating direction method of multipliers (ADMM) to solve the model. Extensive experiments on color image, hyperspectral image, and gray video datasets show that the proposed method achieves superior performance compared to several state-of-the-art approaches. Our code is available at https://github.com/zbx913/Tensor-RPCA-with-TL1-Penalty.

Keywords: TRPCA; image restoration; TL1 penalty; ADMM
Mathematics Subject Classification: 65K05, 90C30

1. Introduction

In fields such as computer vision, medical imaging, and hyperspectral remote sensing, high-dimensional images are often susceptible to noise contamination or data loss during acquisition and transmission. The observed data typically contain underlying essential information that can be characterized by low-dimensional structures, known as the low-rank property of the data. Leveraging this property allows for the effective recovery of clean original data from corrupted observations. Tensor robust principal component analysis (TRPCA) [1] is a natural extension of the classical robust principal component analysis (RPCA) [2] to high-dimensional tensor space. Its objective is to

decompose an observed tensor $\mathcal{Y}$ into the sum of a low-rank tensor $\mathcal{X}$ (representing the background or clean information) and a sparse tensor $\mathcal{E}$ (representing noise or outliers) via an optimization model. The fundamental model can be formulated as

$$\min_{\mathcal{X},\mathcal{E}} \ rank(\mathcal{X}) + \lambda\|\mathcal{E}\|_0 \quad \text{s.t. } \mathcal{Y} = \mathcal{X} + \mathcal{E}, \tag{1.1}$$

where $rank(\cdot)$ is the tensor rank function, $\|\cdot\|_0$ is the l_0 norm, and $\lambda > 0$ is a balancing parameter.

However, the definition of tensor rank is not unique and depends on the specific tensor decomposition method [3]. To address this, researchers have proposed different tensor decomposition frameworks and corresponding convex relaxation strategies to solve model (1.1). For instance: Canonical polyadic (CP) decomposition [4] expresses a tensor as a weighted sum of several rank-one tensors. The corresponding CP rank is defined as the minimum number of rank-one tensors required for such a decomposition. Nevertheless, minimizing the CP rank is a nondeterministic polynomial (NP) hard problem. Tucker decomposition [5] decomposes a tensor into a core tensor multiplied by several factor matrices. The corresponding Tucker rank is defined as the vector composed of the ranks of the mode-unfolding matrices, and directly minimizing its components is also an NP-hard problem. Tensor singular value decomposition (t-SVD) [6] expresses a tensor as the product of two orthogonal tensors and an f-diagonal tensor. The tubal rank derived from this framework is defined as the number of non-zero tubal fibers in the f-diagonal tensor and the average rank is defined as the average of the ranks of the frontal slices in the transformed domain. Directly minimizing these rank functions is likewise NP-hard.

Lu et al. [1], based on the t-SVD framework, proposed using the tensor nuclear norm (TNN) as a convex relaxation of the tubal rank, establishing the following TRPCA model:

$$\min_{\mathcal{X},\mathcal{E}} \ \|\mathcal{X}\|_* + \lambda\|\mathcal{E}\|_1 \quad \text{s.t. } \mathcal{Y} = \mathcal{X} + \mathcal{E}, \tag{1.2}$$

where $\|\cdot\|_*$ is TNN and $\|\cdot\|_1$ is the L_1 norm. This model successfully extends RPCA to the tensor form and has achieved favorable results in practical applications. Subsequent studies have proposed various improvement strategies, such as the tensor partial sum of singular values (TPSSV) method introduced by Zhang et al. [7]. However, such convex relaxation-based methods still suffer from inherent limitations: Equal treatment of singular values excessively penalizes principal signal components while inadequately suppressing noise components, which directly leads to suboptimal recovery quality in convex relaxation models, manifesting as blurred results or residual noise. As the convex envelope of the rank function, TNN cannot accurately approximate the true non-convex rank function. To overcome the aforementioned bias, researchers have further proposed various improved schemes, but their performance is often limited by parameter settings or the accuracy of prior information. The partial sum of tensor nuclear norm (PSTNN) proposed by Jiang et al. [8] penalizes only the smallest few singular values, but its effectiveness heavily depends on a preset truncation position. Mu et al. [9] designed a weighted tensor nuclear norm (WTNN) for the tensor completion (TC) problem, whose weight assignment depends on the unknown true singular values. Zhang et al. [10] proposed an improved tensor principal component pursuit method based on observation priors, but its performance is sensitive to the accuracy of the prior information.

In recent years, researchers have increasingly turned to using non-convex functions to approximate the rank function. Non-convex penalties can better distinguish significant signals from noise and apply differential treatment to them, thereby alleviating the estimation bias caused by uniform shrinkage in

convex relaxation methods. For example, Wang et al. [11] designed a generalized non-convex surrogate function to approximate the tensor tubal rank and proposed a unified non-convex framework for solving the low-rank tensor recovery (LRTR) problem. Gao et al. [12] replaced TNN with a weighted tensor Schatten-p norm and employed the ADMM for solution. Yang et al. [13] proposed a weighted tensor Schatten-p quasi-norm ($0 < p < 1$) regularization method based on t-SVD. Yang et al. [14] proposed a non-convex robust tensor recovery model based on regularized redescending M-estimators and enhanced sparsity measures. To tackle the resulting non-convex optimization problem, they developed an efficient solution framework that combines linearization and proximal block coordinate techniques. Li et al. [15] established a non-convex L_p norm relaxed model for low Tucker rank recovery. The studies by Jiang [16] and Xu [17] replaced TNN with integrated non-convex penalty functions, but did not involve classical non-convex forms like minimax concave penalty (MCP) or smoothly clipped absolute deviation (SCAD) penalty. Li et al. [18] constructed a unified tensor recovery model with reduced bias under the t-SVD algebraic framework by replacing the traditional convex nuclear norm and L_1 norm with a set of carefully designed non-convex penalty functions (SCAD and MCP). They utilized the majorization-minimization (MM) algorithm framework to transform the complex non-convex problem into a series of convex sub-problems that can be efficiently solved. Zhou [19] proposed a robust low-rank tensor constrained orthogonal symmetric non-negative matrix factorization (RTOSNMF) method for community detection in multi-layer networks, aiming to eliminate noise and exploit inter-layer relationships. By stacking the adjacency matrices of all layers into a third-order tensor and imposing a nuclear norm constraint to enforce low-rankness, the method effectively captures higher-order correlations across layers, enabling more accurate identification of intrinsic community structures from noisy data. Pu [20] presented a novel multiview clustering method, which leverages non-convex low-rank tensor approximation and a weighted tensor nuclear norm to capture high-order correlations and complementary information across views. By employing three distinct norms to linearly separate noise, the method significantly enhances clustering robustness.

The alternative forms and L_0 approximation properties of the TL1 function [21] have been extensively studied in [22, 23]. As a non-convex penalty function with a closed-form proximal operator, it has recently been applied by Zhang et al. [24, 25] to compressed sensing, where it not only theoretically guarantees exact recovery but also enables efficient reconstruction via difference-of-convex algorithms and closed-form iterative thresholding. Furthermore, they developed a fast iterative thresholding algorithm based on TL1 for low-rank matrix recovery problems [26]. Inspired by the above research, this paper introduces the TL1 non-convex penalty function [25, 27] into the TRPCA framework, proposing a new non-convex tensor robust principal component analysis model (TL1-TRPCA) and applies ADMM for its solution. For each subproblem, we derive their closed-form solutions. Finally, the method is applied to the restoration of high-dimensional images with sparse noise. The main contributions of this paper lay in the following three aspects:

First, we introduce the TL1 non-convex penalty function into the TRPCA and derive a closed-form solution for its proximal operator. This closed-form solution has a concise thresholding form, which ensures efficient solution of the subproblem.

Second, based on this closed-form solution, the iterative algorithm we designed does not require any inner loop. Each step involves only a simple singular value thresholding operation, resulting in low computational complexity and ease of implementation.

Third, extensive restoration experiments on real-world data demonstrate that the proposed method achieves higher peak signal-to-noise ration (PSNR) and structural similarity (SSIM) than those of the compared methods under various noise levels.

The remainders of this paper are structured as follows. Section 2 provides an overview of the relevant foundational concepts. Section 3 presents the proposed non-convex TRPCA model with the TL1 penalty. Section 4 introduces the proximal operator of the TL1-TNN. Section 5 details the ADMM for solving the optimization problem. Section 6 presents extensive experiments on color image denoising, HSI denoising, and gray video denoising. Finally, Section 7 summarizes the paper and offers concluding remarks.

2. Notations and preliminaries

Based on [1, 28–30], this section presents the foundational tensor algebra and introduces the core notation used throughout this work. We use lowercase letters x for scalars, and vectors are denoted by boldface letters $\boldsymbol{x}$. Matrices are denoted by uppercase letters X. We employ Euler script letters $\mathcal{X}$ for tensors. The fields of complex numbers and real numbers are denoted as $\mathbb{C}$ and $\mathbb{R}$, respectively. For a 3-way tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, the i-th frontal slice $\mathcal{X}(:,:,i)$ is represented by $\mathcal{X}^{(i)}$, $i = 1, \cdots, n_3$. The tubes are vectors in the form of $\mathcal{X}(i, j, :)$. $\|\mathcal{X}\|_1 = \sum_{i,j,k} |x_{i,j,k}|$ and $\|\mathcal{X}\|_F = \sqrt{\sum_{i,j,k} x_{i,j,k}^2}$ are the L_1-norm and the Frobenius norm of $\mathcal{X}$, respectively. For any square matrix M, $tr(M)$ denotes the trace of M. For a matrix $X \in \mathbb{C}^{n_1 \times n_2}$, denote by $\sigma_j(X)$ its j-th largest singular value. For a set A, $\#A$ represents the cardinality of A. The inner product of $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ and $\mathcal{Y} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is $\langle \mathcal{X}, \mathcal{Y} \rangle = \sum_{i=1}^{n_3} \langle \mathcal{X}^{(i)}, \mathcal{Y}^{(i)} \rangle$. $\mathcal{X}^* \in \mathbb{C}^{n_2 \times n_1 \times n_3}$ is the conjugate transpose of $\mathcal{X}$. $\overline{\mathcal{X}}$ is the discrete Fourier transform (DFT) of $\mathcal{X}$ along the third dimension, i.e., $\overline{\mathcal{X}} = \texttt{fft}(\mathcal{X}, [], 3)$, and its inverse is $\mathcal{X} = \texttt{ifft}(\overline{\mathcal{X}}, [], 3)$. The block circulant matrix and the block diagonal matrix are defined, as

$$\overline{X} = \text{bdiag}(\overline{\mathcal{X}}) = \begin{bmatrix} \overline{\mathcal{X}}^{(1)} & & & \\ & \overline{\mathcal{X}}^{(2)} & & \\ & & \ddots & \\ & & & \overline{\mathcal{X}}^{(n_3)} \end{bmatrix}, \qquad \text{bcirc}(\mathcal{X}) = \begin{bmatrix} \mathcal{X}^{(1)} & \mathcal{X}^{(n_3)} & \dots & \mathcal{X}^{(2)} \\ \mathcal{X}^{(2)} & \mathcal{X}^{(1)} & \dots & \mathcal{X}^{(3)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{X}^{(n_3)} & \mathcal{X}^{(n_3-1)} & \dots & \mathcal{X}^{(1)} \end{bmatrix},$$

where $\overline{X} \in \mathbb{C}^{n_1 n_3 \times n_2 n_3}$ is a block diagonal matrix whose ith diagonal block is the ith frontal slice $\overline{\mathcal{X}}^{(i)}$ of $\overline{\mathcal{X}}$ and the block circulant matrix $\text{bcirc}(\mathcal{X}) \in \mathbb{R}^{n_1 n_3 \times n_2 n_3}$ of $\mathcal{X}$.

Let $\mathcal{Y} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ and $Z \in \mathbb{R}^{n_2 \times l \times n_3}$. Then, the t-product for $\mathcal{Y}$ and $\mathcal{Z}$ is $\mathcal{C} \in \mathbb{R}^{n_1 \times l \times n_3}$, i.e.,

$$\mathcal{C} = \mathcal{Y} * \mathcal{Z} = \text{fold}(\text{bcirc}(\mathcal{Y}) \cdot \text{unfold}(\mathcal{Z})).$$

It is equivalent to $\overline{\mathcal{C}} = \overline{\mathcal{Y}} \cdot \overline{\mathcal{Z}}$, where the operators that unfold and fold are specified as follows:

$$\text{unfold}(\mathcal{X}) = \begin{bmatrix} \mathcal{X}^{(1)} \\ \mathcal{X}^{(2)} \\ \vdots \\ \mathcal{X}^{(n_3)} \end{bmatrix} \quad \text{and} \quad \text{fold}(\text{unfold}(\mathcal{X})) = \mathcal{X}.$$

The t-SVD of a tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is defined as

$$\mathcal{X} = \mathcal{U} * \mathcal{S} * \mathcal{V}^*,$$

where $\mathcal{U} \in \mathbb{R}^{n_1 \times n_1 \times n_3}$ and $\mathcal{V} \in \mathbb{R}^{n_2 \times n_2 \times n_3}$ are orthogonal tensors, i.e., $\mathcal{U}^* * \mathcal{U} = \mathcal{V}^* * \mathcal{V} = \mathcal{I}$ and $\mathcal{I} \in \mathbb{R}^{n \times n \times n_3}$ is an identity tensor. $\mathcal{S} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is an f-diagonal tensor whose frontal slices are all diagonal matrices. In the Fourier domain, $\overline{\mathcal{X}}^{(i)} = \overline{\mathcal{U}}^{(i)} * \overline{\mathcal{S}}^{(i)} * (\overline{\mathcal{V}}^{(i)})^*$, $i = 1, ..., n_3$, where $\overline{\mathcal{U}}^{(i)} \in \mathbb{C}^{n_1 \times n_1}$ and $\overline{\mathcal{V}}^{(i)} \in \mathbb{C}^{n_2 \times n_2}$ are unitary matrices, and $\overline{\mathcal{S}}^{(i)}$ is a rectangular diagonal matrix with non-negative diagonal entries: $\overline{\mathcal{S}}^{(i)}(j, j) = \sigma_j(\overline{\mathcal{X}}^{(i)}), 1 \leq j \leq min\{n_1, n_2\}$. Hence, the tensor tubal rank of $\mathcal{X}$ is defined as the number of nonzero singular tubes of $\mathcal{S}$. Using the $'\#'$ to denote the number of elements, it can be written as

$$\text{rank}_t(\mathcal{X}) = \#\{j : \mathcal{S}(j, j, :) \neq \mathbf{0}\}.$$

The $\text{rank}_t(\mathcal{X})$ can also be expressed as the number of nonzero diagonal entries of the first frontal slice $\mathcal{S}(:, :, 1)$, i.e., $\text{rank}_t(\mathcal{X}) = \#\{j : \mathcal{S}(j, j, 1) \neq 0\}$. This equivalence follows from the property induced by the inverse DFT:

$$\mathcal{S}(j, j, 1) = \frac{1}{n_3} \sum_{i=1}^{n_3} \overline{\mathcal{S}}(j, j, i).$$

The singular values are expressed as $\sigma_j(\overline{\mathcal{X}}^{(i)}) = \overline{\mathcal{S}}(j, j, i)$ of $\overline{\mathcal{X}} = \overline{\mathcal{U}} * \overline{\mathcal{S}} * \overline{\mathcal{V}}^*$ in the Fourier domain, $j \in \{1, ..., m\}$, $m = min\{n_1, n_2\}$. Therefore, the definition of the tensor nuclear norm (TNN) of $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is given as

$$\|\mathcal{X}\|_* = \frac{1}{n_3} \sum_{i=1}^{n_3} \|\overline{\mathcal{X}}^{(i)}\|_* = \frac{1}{n_3} \sum_{i=1}^{n_3} \sum_{j=1}^{m} \sigma_j(\overline{\mathcal{X}}^{(i)}). \tag{2.1}$$

3. New model for TRPCA

First, we introduce the definition of the TL1 penalty function [23–25] for a vector $\boldsymbol{x} \in \mathbb{R}^n$. It can be decomposed into a sum of functions over its components, that is,

$$\text{TL1}_a(\boldsymbol{x}) = \sum_{i=1}^{n} \psi_a(x_i),$$

where $a \in (0, +\infty)$ works as a scale parameter (also called a regularization parameter in optimization models). The function $\psi_a(x_i) = \frac{(a+1)|x_i|}{a+|x_i|}$, which includes unbiasedness, sparsity, and continuity [21], has the following asymptotic forms:

$$\lim_{a \to 0^+} \psi_a(x_i) = I_{\{x_i \neq 0\}}, \quad \lim_{a \to +\infty} \psi_a(x_i) = |x_i|,$$

where $I_{\{x_i \neq 0\}}$ is the indicator function. With the change of parameter a, the TL1 interpolates L_0 and L_1 norms:

$$\lim_{a \to 0^+} \text{TL1}_a(\boldsymbol{x}) = \sum_{i=1}^{n} I_{\{x_i \neq 0\}} = \|\boldsymbol{x}\|_0, \quad \lim_{a \to +\infty} \text{TL1}_a(\boldsymbol{x}) = \sum_{i=1}^{n} |x_i| = \|\boldsymbol{x}\|_1.$$

Combining TNN and TL1, we propose the following definition.

Definition 3.1. *For a tensor $\mathcal{X} \in \mathbb{R}^{n_1\times n_2\times n_3}$, its TL1-TNN is defined as the sum of all non-linearly varying singular values of all frontal slices in the Fourier domain, that is*

$$\|\mathcal{X}\|_{TL1,*} = \frac{1}{n_3}\sum_{i=1}^{n_3}\|\overline{\mathcal{X}}^{(i)}\|_{TL1,*} = \frac{1}{n_3}\sum_{i=1}^{n_3}\sum_{j=1}^{m}\psi_a\Big(\sigma_j(\overline{\mathcal{X}}^{(i)})\Big),\ \ \psi_a(\sigma_j(\overline{\mathcal{X}}^{(i)}) = \frac{(a+1)\sigma_j(\overline{\mathcal{X}}^{(i)})}{a+\sigma_j(\overline{\mathcal{X}}^{(i)})}, \tag{3.1}$$

where $\sigma_j(\overline{\mathcal{X}}^{(i)}) = \overline{\mathcal{S}}(j,j,i)$, $\overline{\mathcal{X}}^{(i)}$ is the i-th frontal slice.

The TNN has a critical limitation in model (1.2): It enforces uniform shrinkage on all singular values. Thus, dominant singular values encoding the essential low-rank structure are penalized as strongly as minor ones typically associated with noise. This results in the over-shrinkage of significant signal components, thereby causing distorted estimation and hindering the accurate recovery of the original low-rank structure. Building on the above description, we employ a non-convex surrogate for the tensor rank function by generalizing the TL1 penalty to operate on the singular values of a tensor. As a result, a non-convex TRPCA model with TL1 (TL1-TRPCA) is proposed:

$$\min_{\mathcal{X},\mathcal{E}}\ \|\mathcal{X}\|_{TL1,*} + \lambda\|\mathcal{E}\|_1 \quad \text{s.t. } \mathcal{Y} = \mathcal{X} + \mathcal{E}, \tag{3.2}$$

where $\|\cdot\|_{TL1,*}$ is a TL1 tensor nuclear norm (TL1-TNN), which serves as a non-convex surrogate for the tensor low-rank function (1.1), and will lead to a superior approximation of the low-rank function compared to the conventional TNN.

4. Proximal operator of TL1-TNN

Building upon the closed-form solution for the unconstrained TL1-regularized vector problem presented in [24], we now extend this framework to the matrix case:

$$\min_{X\in\mathbb{C}^{n_1\times n_2}} \mathrm{TL1}_a(X) + \frac{1}{2\gamma}\|Q-X\|_F^2. \tag{4.1}$$

Here, $\mathrm{TL1}_a(X) = \|X\|_{TL1,*} = \sum_{i=1}^{m}\psi_a(\sigma_i(X))$, $m = min\{n_1,n_2\}$, $\sigma_i(X)$ is a singular value decomposition of matrix $X = USV^*$, $X \in \mathbb{C}^{n_1\times n_2}$, $i \in [1,n_1]$, and $j \in [1,n_2]$, with the closed-form solution given elementwise by

$$\begin{aligned}\mathrm{prox}_{\gamma TL1}(Q) &= \arg\min_{X\in\mathbb{C}^{n_1\times n_2}} \mathrm{TL1}_a(X) + \frac{1}{2\gamma}\|Q-X\|_F^2 \\ &= U_Q S_{\gamma,TL1}(Q) V_Q^*\end{aligned} \tag{4.2}$$

where $Q = U_Q S_Q V_Q^*$ is the singular value decomposition of matrix Q, and

$$S_{\gamma,TL1}(Q) = D_{\gamma,TL1}(diag(\sigma_1,...,\sigma_m)),\ m = min\{n_1,n_2\}.$$

Here, $D_{\gamma,TL1}(\cdot)$ is a singular value thresholding operator, that is

$$\mathrm{D}_{\gamma TL1}(\sigma_i) = \begin{cases}\frac{2}{3}(a+\sigma_i)\cos\frac{\phi(\sigma_i)}{3} - \frac{2}{3}a + \frac{\sigma_i}{3}, & \text{if } \sigma_i > \theta,\\ 0, & \text{if } \sigma_i \le \theta,\end{cases} \tag{4.3}$$

where

$$\phi(\sigma_i) = \arccos\left(1 - \frac{27\gamma a(a+1)}{2(a+\sigma_i)^3}\right),$$

and

$$\theta = \begin{cases} \gamma\dfrac{a+1}{a}, & \text{if } \gamma \le \dfrac{a^2}{2(a+1)}, \\ \sqrt{2\gamma(a+1)} - \dfrac{a}{2}, & \text{if } \gamma > \dfrac{a^2}{2(a+1)}. \end{cases}$$

Next, we extend the formulation (4.1) to a tensor setting. For a tensor $\mathcal{Q} \in \mathbb{R}^{n_1\times n_2\times n_3}$, the proximal operator associated with TL1 penalty is defined as

$$\text{prox}_{\gamma TL1}(\mathcal{Q}) = \arg\min_{\mathcal{X}} \|\mathcal{X}\|_{TL1,*} + \frac{1}{2\gamma}\|\mathcal{X} - \mathcal{Q}\|_F^2. \tag{4.4}$$

Theorem 4.1. *The proximal operator of (4.4) is solved by*

$$\text{prox}_{\gamma TL1}(\mathcal{Q}) = \mathcal{U}_{\mathcal{Q}} * \text{ifft}(\mathcal{S}_{\gamma TL1}(\overline{\mathcal{Q}})) * \mathcal{V}_{\mathcal{Q}}^*, \tag{4.5}$$

where $\mathcal{Q} = \mathcal{U}_{\mathcal{Q}} * \mathcal{S}_{\mathcal{Q}} * \mathcal{V}_{\mathcal{Q}}^*$ *is the t-SVD of* $\mathcal{Q}$, $\overline{\mathcal{S}}_{\gamma TL1} = \mathcal{S}_{\gamma TL1}(\overline{\mathcal{Q}}) = \text{fold}(\{\overline{\mathcal{S}}_{\gamma TL1}^{(i)}\}_{i=1}^{n_3})$, *and* $\overline{\mathcal{S}}_{\gamma TL1}^{(i)} = \text{prox}_{\gamma TL1}(\overline{\mathcal{S}}_Q^{(i)})$.

Proof. For any tensor $\mathcal{X} \in \mathbb{R}^{n_1\times n_2\times n_3}$, the relationship between the Frobenius norm of $\mathcal{X}$ and its DFT $\overline{\mathcal{X}}$ is $\|\mathcal{X}\|_F = \frac{1}{\sqrt{n_3}}\|\overline{\mathcal{X}}\|_F$ [1]. Moreover, the TL1-TNN is defined as

$$\|\mathcal{X}\|_{TL1,*} = \frac{1}{n_3}\sum_{i=1}^{n_3} \|\overline{\mathcal{X}}^{(i)}\|_{TL1,*},$$

where $\overline{\mathcal{X}}^{(i)}$ is the i-th frontal slice of $\overline{\mathcal{X}}$, and the matrix TL1-TNN $\|\cdot\|_{TL1,*}$ is the sum of TL1 penalties on the singular values (see (3.1)). Using the DFT relation for $\|\mathcal{X} - \mathcal{Q}\|_F$, then

$$\|\mathcal{X} - \mathcal{Q}\|_F^2 = \frac{1}{n_3}\|\overline{\mathcal{X}} - \overline{\mathcal{Q}}\|_F^2 = \frac{1}{n_3}\sum_{i=1}^{n_3} \|\overline{\mathcal{X}}^{(i)} - \overline{\mathcal{Q}}^{(i)}\|_F^2.$$

Thus, for (4.4), we have

$$\begin{aligned} \text{prox}_{TL1}(\mathcal{Q}) &= \arg\min_{\mathcal{X}} \|\mathcal{X}\|_{TL1,*} + \frac{1}{2\gamma}\|\mathcal{X} - \mathcal{Q}\|_F^2 \\ &= \arg\min_{\mathcal{X}} \frac{1}{n_3}\sum_{i=1}^{n_3}\left(\|\overline{\mathcal{X}}^{(i)}\|_{TL1,*} + \frac{1}{2\gamma}\|\overline{\mathcal{X}}^{(i)} - \overline{\mathcal{Q}}^{(i)}\|_F^2\right). \end{aligned}$$

Since multiplication by the positive constant n_3 does not change the minimizer, the original problem is equivalent to solving, for each $i = 1, \ldots, n_3$ independently,

$$\overline{\mathcal{X}}^{(i)} = \text{prox}_{\gamma TL1}(\overline{\mathcal{Q}}^{(i)}) = \arg\min_{\mathcal{X}^{(i)}} \|\overline{\mathcal{X}}^{(i)}\|_{TL1,*} + \frac{1}{2\gamma}\|\overline{\mathcal{X}}^{(i)} - \overline{\mathcal{Q}}^{(i)}\|_F^2.$$

Each subproblem is a matrix TL1 proximal operator. Let $\overline{Q}^{(i)} = \overline{\mathcal{U}}_Q^{(i)}\overline{\mathcal{S}}_Q^{(i)}(\overline{\mathcal{V}}_Q^{(i)})^*$ be its singular value decomposition. From (4.2), we can obtain

$$\overline{\mathcal{X}}^{(i)} = \overline{\mathcal{U}}_Q^{(i)}\overline{\mathcal{S}}_{\gamma TL1}^{(i)}(\overline{\mathcal{V}}_Q^{(i)})^*, \tag{4.6}$$

where $\overline{\mathcal{S}}_{\gamma TL1}^{(i)} = \text{prox}_{\gamma TL1}(\overline{\mathcal{S}}_Q^{(i)})$, and $\text{prox}_{\gamma TL1}$ applies the scalar TL1 proximal operator elementwise to the diagonal entries. Upon stacking these frontal slices into a tensor $\overline{\mathcal{X}}$ and recognizing that $\overline{\mathcal{S}}_{\gamma TL1} = \text{fold}(\{\overline{\mathcal{S}}_{\gamma TL1}^{(i)}\}_{i=1}^{n_3})$, the inverse DFT then leads to

$$\mathcal{X}^* = \text{prox}_{TL1}(\mathcal{Q}) = \mathcal{U}_Q * \text{ifft}(\overline{\mathcal{S}}_{\gamma TL1}) * \mathcal{V}_Q^*.$$

From Theorem 4.1, the TL1-TNN minimization problem in (4.4) is separable across the frontal slices of the Fourier domain, as given by (4.6). Since each slice is a matrix, its proximal operator follows directly from the result in (4.2). By applying this result to all slices and transforming back, we obtain the closed-form global proximal operator given in (4.5).

5. ADMM for solving (3.2)

This section presents the ADMM for solving the non-convex optimization problem TL1-TRPCA (3.2). We give the augmented Lagrangian function of the proposed model:

$$\mathcal{L}(\mathcal{X}, \mathcal{E}, \mathcal{M}) = \|\mathcal{X}\|_{TL1,*} + \lambda\|\mathcal{E}\|_1 + \langle \mathcal{M}, \mathcal{X} + \mathcal{E} - \mathcal{Y}\rangle + \frac{\mu}{2}\|\mathcal{X} + \mathcal{E} - \mathcal{Y}\|_F^2,$$

where $\mathcal{M}$ is the Lagrange multiplier and $\mu > 0$ is the penalty parameter. In the ADMM framework, the optimization variables are updated as

$$\begin{cases} \mathcal{X}^{k+1} = \arg\min\limits_{\mathcal{X}} \mathcal{L}(\mathcal{X}, \mathcal{E}^k, \mathcal{M}^k), \\ \mathcal{E}^{k+1} = \arg\min\limits_{\mathcal{E}} \mathcal{L}(\mathcal{X}^{k+1}, \mathcal{E}, \mathcal{M}^k), \\ \mathcal{M}^{k+1} = \mathcal{M}^k + \mu(\mathcal{X}^{k+1} + \mathcal{E}^{k+1} - \mathcal{Y}), \end{cases}$$

where k is the iteration number. Next, we provide the detailed solutions for the $\mathcal{X}$ and $\mathcal{E}$ subproblems.

(1) The $\mathcal{X}$ subproblem:

$$\mathcal{X}^{k+1} = \arg\min_{\mathcal{X}} \|\mathcal{X}\|_{TL1,*} + \langle \mathcal{M}, \mathcal{X} + \mathcal{E}^k - \mathcal{Y}\rangle + \frac{\mu^k}{2}\|\mathcal{X} + \mathcal{E}^k - \mathcal{Y}\|_F^2.$$

After rearranging terms, we obtain

$$\min_{\mathcal{X}} \frac{1}{\mu^k}\|\mathcal{X}\|_{TL1,*} + \frac{1}{2}\|\mathcal{X} - \mathcal{N}^k\|_F^2,$$

where $\mathcal{N}^k = \mathcal{Y} - \frac{1}{\mu^k}\mathcal{M}^k - \mathcal{E}^k$. By (4.5), the closed-form solution is given by

$$\mathcal{X}^{k+1} = \text{prox}_{\frac{1}{\mu^k}TL1}(\mathcal{N}^k). \tag{5.1}$$

Algorithm 1 outlines the concrete procedure for solving this subproblem.

Algorithm 1 Solving to (5.1)

1. **Input:** $\mu > 0$, $\mathcal{N} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$.
2. Compute $\overline{\mathcal{N}} = fft(\mathcal{N}, [\,], 3)$.
3. Compute each frontal slice of $\overline{\mathcal{N}}$ by
   for $i = 1, ..., \lceil \frac{n_3+1}{2} \rceil$ **do**
   $[\overline{\mathcal{U}}^{(i)}, \overline{\mathcal{S}}^{(i)}, \overline{\mathcal{V}}^{(i)}] \leftarrow \text{SVD}(\overline{\mathcal{N}}^{(i)})$
   Singular value threshold shrinkage for $\overline{\mathcal{S}}^{(i)}$ by (4.3)
   $\overline{\mathcal{N}}^{(i)} \leftarrow \overline{\mathcal{U}}^{(i)} \cdot (\overline{\mathcal{S}}^{(i)}) \cdot (\overline{\mathcal{V}}^{(i)})^*$
   end for
   for $i = \lceil \frac{n_3+1}{2} \rceil + 1, ..., n_3$ **do**
   $\overline{\mathcal{N}}^{(i)} \leftarrow \text{conj}(\overline{\mathcal{N}}^{(n_3+2-i)})$
   end for
4. Compute $\text{prox}_{\frac{1}{\mu}TL1}(\mathcal{N}) = ifft(\overline{\mathcal{N}}, [\,], 3)$.
5. **Output:** $\text{prox}_{\frac{1}{\mu}TL1}(\mathcal{N})$.

(2) The $\mathcal{E}$ subproblem:

$$\mathcal{E}^{k+1} = \arg\min_{\mathcal{E}} \lambda \|\mathcal{E}\|_1 + \langle \mathcal{M}^k, \mathcal{X}^{k+1} + \mathcal{E} - \mathcal{Y} \rangle + \frac{\mu^k}{2} \|\mathcal{X}^{k+1} + \mathcal{E} - \mathcal{Y}\|_F^2.$$

After rearranging, we have

$$\min_{\mathcal{E}} \frac{\lambda}{\mu^k} \|\mathcal{E}\|_1 + \frac{1}{2} \|\mathcal{E} - \mathcal{J}^k\|_F^2,$$

where $\mathcal{J}^k = \mathcal{Y} - \frac{1}{\mu^k}\mathcal{M}^k - \mathcal{X}^{k+1}$. The optimal solution is given by the soft thresholding operator:

$$\mathcal{E}^{k+1} = \mathcal{T}_{\frac{\lambda}{\mu^k}}(\mathcal{J}^k), \tag{5.2}$$

where $\mathcal{T}_{\frac{\lambda}{\mu^k}}(\cdot)$ is the soft thresholding operator, defined as

$$\mathcal{T}_\tau(x) = sgn(x) \max\{|x| - \tau, 0\}, x \in \mathbb{R},$$

and $sgn(\cdot)$ is the sign function.

(3) Updating Lagrange multiplier $\mathcal{M}$:

$$\mathcal{M}^{k+1} = \mathcal{M}^k + \mu^k(\mathcal{X}^{k+1} + \mathcal{E}^{k+1} - \mathcal{Y}). \tag{5.3}$$

Finally, Algorithm 2 summarizes the main steps for solving each subproblem of the proposed non-convex model (3.2).

Since (3.2) introduces non-convex functions as approximate relaxations of low-rank, the objective function becomes a non-convex problem. The convergence of the ADMM for minimizing a non-convex and possibly non-smooth objective function has been discussed in related references. The developed

convergence guarantee covered a variety of non-convex and non-smooth objective functions such as piecewise linear functions, MCP, SCAD, and TL1 penalty [31, 32]. Our model belongs to the frame of convergence theory, and the proposed ADMM will lead to convergence guarantee by using similar proof techniques [33, 34]. In fact, the convergence of ADMM for solving model (3.2) has been given by [35] as follows.

Algorithm 2 ADMM for solving (3.2).

1. **Input:** The observation data $\mathcal{Y}$.
2. **Initialize:** $\mu_0 > 0, \rho > 1, k = 1000, \mu_{\max} = 10^{10}, \epsilon = 10^{-10}, \mathcal{X}^k = \mathcal{E}^k = \mathcal{M}^k = 0$.
3. **while** not convergent **do**
   (a) Update $\mathcal{X}^{k+1}$ by (5.1).
   (b) Update $\mathcal{E}^{k+1}$ by (5.2).
   (c) Update $\mathcal{M}^{k+1}$ by (5.3).
   (d) $\mu^{k+1} = \min\{\rho\mu^k, \mu_{\max}\}$.
   (e) Verify the stop conditions

$$\max\{\|\mathcal{X}^{k+1} - \mathcal{X}^k\|_\infty, \|\mathcal{E}^{k+1} - \mathcal{E}^k\|_\infty, \|\mathcal{X}^{k+1} + \mathcal{E}^{k+1} - \mathcal{Y}\|_\infty\} \leq \epsilon.$$

4. **end while**
5. **Output:** $\mathcal{X}$ and $\mathcal{E}$.

Theorem 5.1. *Suppose that $\mathcal{P}_k = \{\mathcal{X}_k, \mathcal{E}_k, \mathcal{M}_k\}$ is generated from the proposed Algorithm 2, then $\{\mathcal{P}_k\}$ is bounded. In addition, there exists a subsequence of $\mathcal{P}_k$ that converges to a stationary point $\{\mathcal{X}^*, \mathcal{E}^*, \mathcal{M}^*\}$, which satisfies the Karush-Kuhn-Tucker (KKT) conditions.*

Remark 5.1. *The detailed proof of Theorem 5.1 follows that of the Theorem 2 in [35].*

Remark 5.2. *The key difference of our Algorithm 2 and the ADMM in [35] is solving the $\mathcal{X}$ subproblem: [35] adopts the first-order Taylor expansion and the difference of convex functions algorithm to obtain the local minimum after several iterations, while we use the proximal operator of the TL1 function to derive a closed-form solution for the $\mathcal{X}$-subproblem.*

Moreover, as discussed in [36], the TL1 function satisfies the Kurdyka-Lojasiewicz (KL) property. However, the convergence of the algorithm does not fully rely on the KL property. In this paper, we adopt the ADMM framework for solving the problem, where each subproblem admits a closed-form solution. By leveraging the classic analysis framework for non-convex ADMM, we verify that: the iterate sequence $\mathcal{P}_k$ is bounded; from the Bolzano-Weierstrass theorem, there must be at least one accumulation point of the sequence, and any cluster point of the subsequence $\mathcal{P}_k$ is a stationary point satisfying the KKT conditions. These conclusions only require the following regularity assumptions: The TL1 function is differentiable at non-zero points with bounded derivative and the penalty parameter μ increases. In the future, we will further investigate the KL property of the TL1-TNN model and refine the global theoretical analysis. Although a complete global convergence proof to the original non-convex problem remains an open challenge, the sequence generated by Algorithm 2 admits at least one limit point and every limit point is a stationary point of the original non-convex energy function.

6. Experiments

In this section, we evaluate the performance of the proposed method. Color image, hyperspectral image (HSI), and gray video data are tested. All simulations are implemented on a PC with an i9-13900H CPU 2.60GHz and 32.0GB memory. The quality of the restored result is evaluated using two widely adopted metrics: PSNR and SSIM. Higher PSNR and SSIM values indicate that the recovered image is closer to the ground truth. For a restored image, the PSNR is defined as follows:

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{\|\mathcal{X}\|_\infty^2}{\frac{1}{n_1n_2n_3}\|\overline{\mathcal{X}}-\mathcal{X}\|_F^2}\right),$$

where $\|\mathcal{X}\|_\infty$ denotes the infinity norm. The SSIM between two images $\mathcal{X}$ and $\mathcal{Y}$ is given by the following formula:

$$\mathrm{SSIM}(\mathcal{X},\mathcal{Y}) = \frac{(2\mu_x\mu_y + c_1)(2\delta_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\delta_x^2 + \delta_y^2 + c_2)},$$

where μ_x and μ_y are the mean of $\mathcal{X}$ and $\mathcal{Y}$, respectively. δ_x^2 and δ_y^2 are their variances, and δ_{xy} is their covariance. The terms $c_1 = (k_1L)^2$ and $c_2 = (k_2L)^2$ are stabilizing constants, with L denoting the dynamic range of the pixel values, $k_1 = 0.01$, and $k_2 = 0.03$.

6.1. Color image denoising

We set the same parameters $\lambda = \frac{1}{\sqrt{\max\{n_1,n_2\}n_3}}$ [1], $\rho = 1.02$, $\mu_0 = 10^{-4}$, $\mu_{max} = 10^{10}$, and $\varepsilon = 10^{-10}$ in TL1-TRPCA method. The stop criterion is that the relative error is less than 10^{-10} and the maximum number of iterations is 1000. For TRPCA [1], ETRPCA [12], and BTRTF [37], the parameters are set to be the same as those in related references.

In this experiment, TL1-TRPCA is applied to natural image restoration from partial pixel observations, based on the representation of color images as third-order low-rank tensors comprising R, G, and B channels. We select the 100 testcolor images from the Berkeley Segmentation Dataset (BSD) [38], each of size $481 \times 321 \times 3$ or $321 \times 481 \times 3$. For each image, we corrupt 20% and 30% of its pixels by replacing them with values randomly sampled (SR) from the range [0, 255], and the corruption positions are unknown. In our model, $a = 10$ for 20% noise level and $a = 20$ for 30% noise level.

Since image restoration performance is positively correlated with PSNR and SSIM values, we evaluate the effectiveness of different algorithms in restoring images based on these two metrics. We use the BSD100 image dataset for testing, and the results of PSNR and SSIM are presented in Table 1 and Table 2. Table 1 provides the PSNR and SSIM values for the restoration performance of different algorithms under 20% and 30% noise corruption, while Table 2 lists the average PSNR and SSIM values of the restoration results for 100 images using four algorithms. The data in both tables clearly demonstrate that the proposed TL1-TRPCA method outperforms the other three compared methods, yielding better restoration results.

In addition to the numerical evaluation provided by PSNR and SSIM, Figures 1 and 2 provide the visual results corresponding to the numerical data in Table 1. Each figure intuitively displays the restoration results for the selected four images, including the original images, the noisy images, and the restored images produced by TRPCA, ETRPCA, and BTRTF, respectively. The visual results

of different algorithms further demonstrate the superior performance of the TL1-TRPCA in image restoration.

Table 1. Performance comparison of PSNR and SSIM at 20% and 30% noise levels.

SR	Metrics	TRPCA	ETRPCA	BTRTF	TL1-TRPCA
20%	PSNR	27.61	28.65	29.01	**30.22**
	SSIM	0.817	0.828	0.855	**0.886**
	PSNR	29.45	30.14	30.25	**31.99**
	SSIM	0.865	0.840	0.841	**0.903**
	PSNR	29.51	31.12	31.80	**32.77**
	SSIM	0.904	0.887	0.905	**0.936**
	PSNR	30.49	31.46	32.79	**33.30**
	SSIM	0.889	0.871	0.915	**0.928**
30%	PSNR	24.57	23.66	24.46	**25.33**
	SSIM	0.793	0.760	0.730	**0.812**
	PSNR	24.10	24.77	24.92	**26.08**
	SSIM	0.792	0.750	0.688	**0.794**
	PSNR	28.34	29.15	29.52	**30.14**
	SSIM	0.834	0.823	0.799	**0.863**
	PSNR	29.34	31.09	26.63	**31.16**
	SSIM	0.875	0.884	0.667	**0.899**

Table 2. Average results of PSNR and SSIM for 100 images.

SR	Metrics	TRPCA	ETRPCA	BTRTF	TL1-TRPCA
20%	PSNR	27.15	29.45	29.97	**30.82**
	SSIM	0.888	0.898	0.882	**0.926**
30%	PSNR	25.06	25.62	25.73	**26.33**
	SSIM	0.804	0.783	0.734	**0.809**

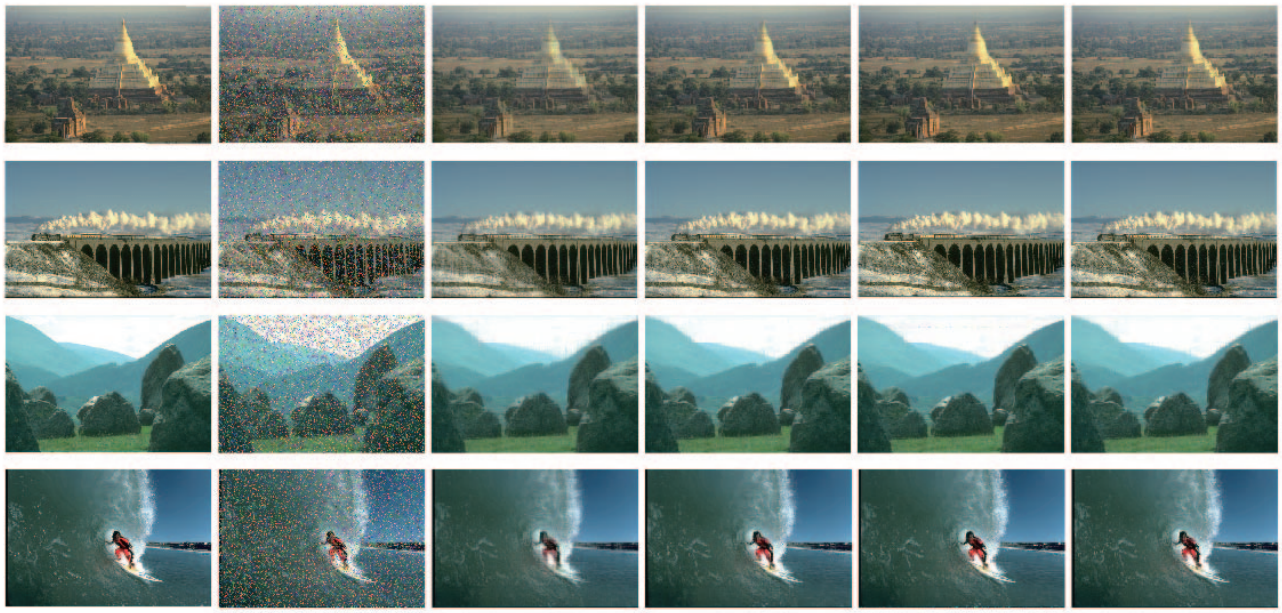


Figure 1. Restoration results displayed for the selected four images with 20% noise.

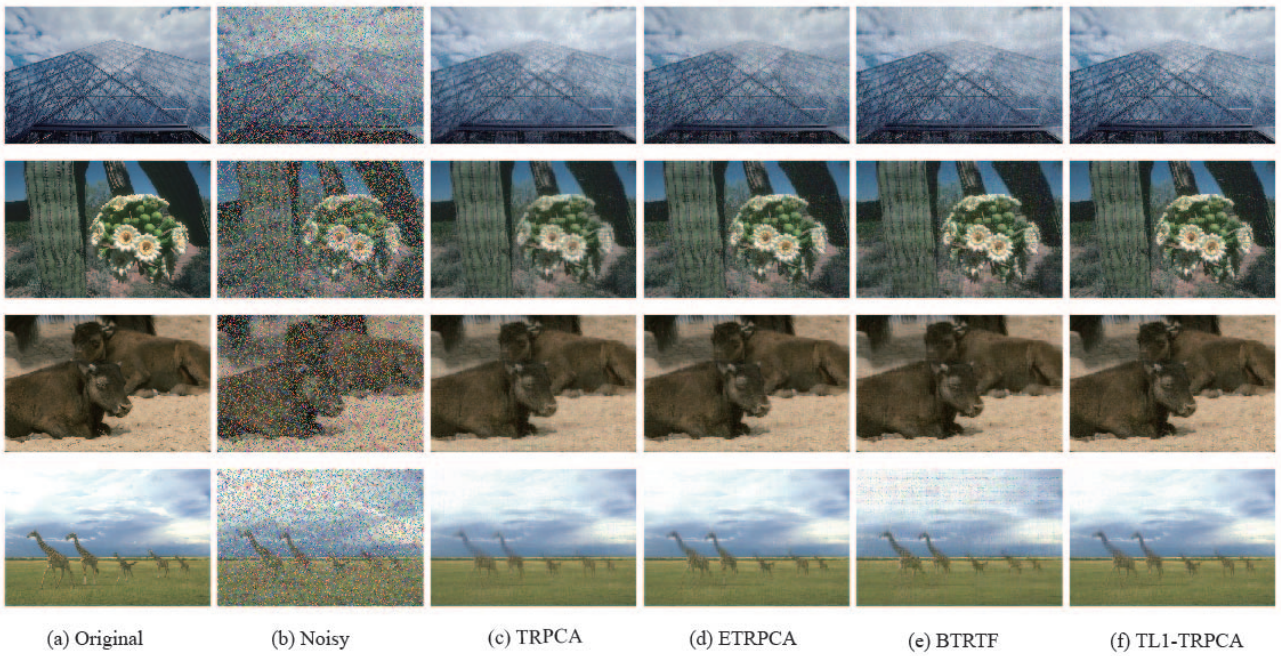
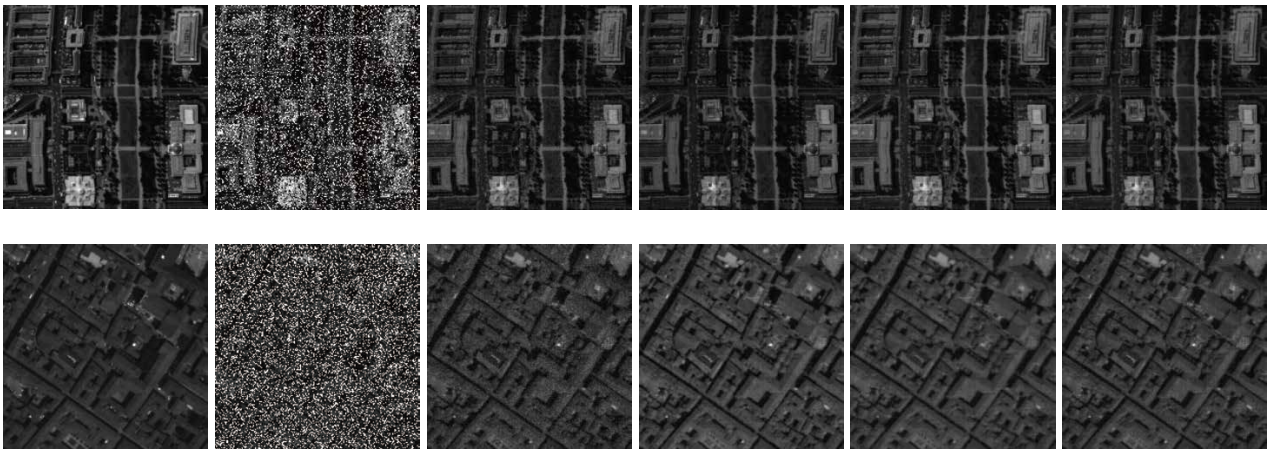


Figure 2. Restoration results displayed for the selected four images with 30% noise.

6.2. HSI denoising

In this subsection, we test the DC Mall and Pavia HSI data-sets. The random salt-and-pepper noise level is set to 30% and 40%. We set the same parameters $\lambda = \frac{1}{\sqrt{\max\{n_1,n_2\}n_3}}$ [1] and $\rho = 1.2$, with $a = 165$ for 30% noise-level and $a = 200$ for 40% noise level in the TL1-TRPCA method. The compared

methods are as follows: TRPCA, MCP-TRPCA, and SP_(0.9)-TRPCA (Schatten-p quasi-norm with $p = 0.9$). Table 3 lists the PSNR and SSIM values by different methods. From these results, we observe that our method evidently performs better than the competing ones in terms of all the evaluation measures. In Figure 3, we show one band from these two HSIs. From the figure and table, our method almost removes sparse noise and yields the better results.



(a) Original (b) Noisy (c) TRPCA (d) MCP-TRPCA (e) SP_(0.9)-TRPCA (f) TL1-TRPCA

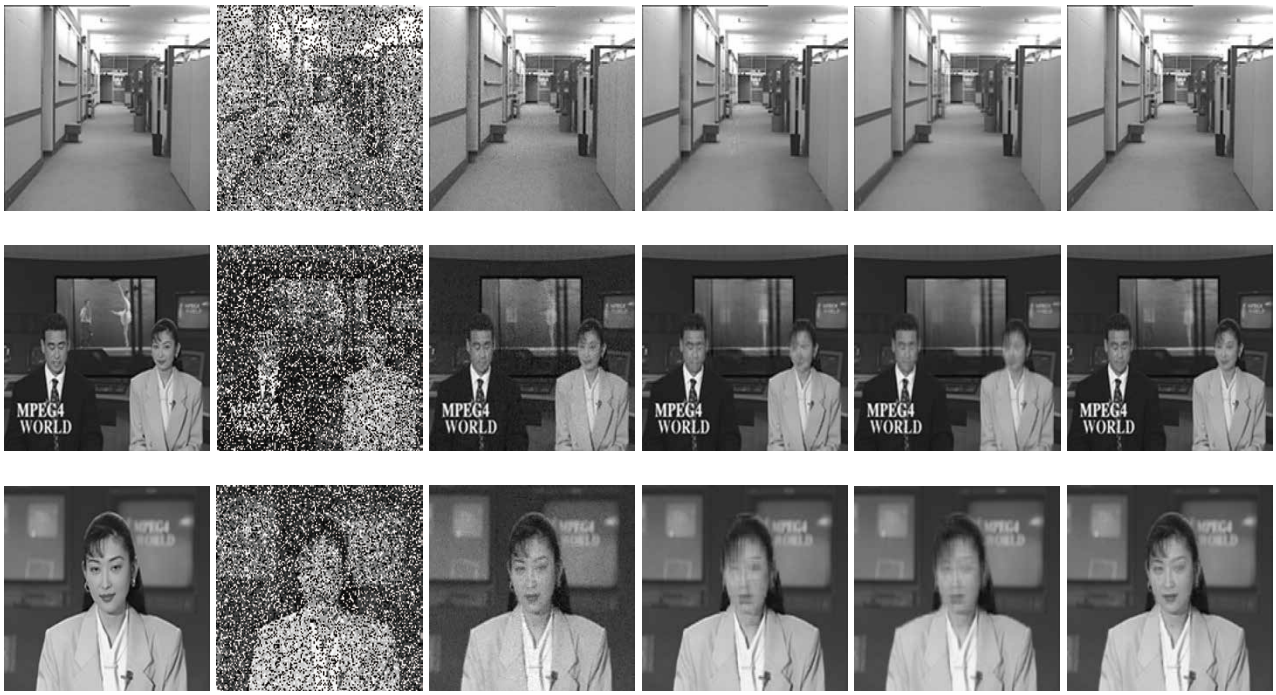
Figure 3. The denoising results of the DC Mall with 30% noise and Pavia with 40% noise.

Table 3. The PSNR and SSIM values of the four TRPCA methods.

Data	DC				PA			
Noise level	30%		40%		30%		40%	
Metrics	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
TRPCA	33.304	0.946	32.554	0.939	33.287	0.924	30.293	0.861
MCP-TRPCA	33.857	0.959	32.097	0.944	34.650	0.946	32.202	0.921
SP_(0.9)-TRPCA	37.102	0.977	34.345	0.959	33.888	0.953	32.623	0.939
TL1-TRPCA	**40.135**	**0.988**	**35.389**	**0.969**	**43.940**	**0.987**	**38.302**	**0.973**

6.3. Gray video denoising

In this subsection, we conduct the denoising task of all competing methods on the Hall, News, and Akiyo video sequences. All video sequences are processed with size $176 \times 144 \times 100$. We set the parameters $\lambda = \frac{1}{\sqrt{\max\{n_1,n_2\}n_3}}$ [1], $\rho = 1.2$, and $a = 1000$ for the TL1-TRPCA method. The denoising results are given in Table 4 and Figure 4. In Table 4, the values of quantitative metrics are listed. In Figure 4, we display the denoising results for the 1st frame at a 40% noise level. When the percentage of sparse noise is 40%, our methods still have better denoising performance, but others lose effectiveness. As observed, our TL1-TRPCA method achieves the best visual results in terms of both noise removal and detail preservation.



(a) Original (b) Noisy (c) TRPCA (d) MCP-TRPCA (e) SP_(0.9)-TRPCA (f) TL1-TRPCA

Figure 4. Video denoising by different methods with 40% noise level.

Table 4. Performance of video denoising by competing methods under 40% noise level.

data	Metrics	TRPCA	MCP-TRPCA	SP_(0.9)-TRPCA	TL1-TRPCA
Hall	PSNR	30.656	29.333	28.567	**31.647**
	SSIM	0.878	0.941	0.944	**0.959**
News	PSNR	29.973	28.617	28.129	**31.456**
	SSIM	0.848	0.898	0.908	**0.945**
Akiyo	PSNR	30.766	29.658	29.431	**33.523**
	SSIM	0.823	0.925	0.930	**0.970**

6.4. *Parameter analysis*

In fact, the TNN is the sum of nuclear norms pertaining to each frontal slice within the Fourier domain. It can be represented as the L_1 norm of all the singular vectors. The parameter a in the TL1 function controls the level of concavity. If a takes an appropriate value, the function can more precisely approximate the rank. This can easily be seen in Figure 5, in which the different values of a are plotted. Therefore, the parameter a should be flexibly adjusted based on the characteristics of the data and noise level: If a is too small, the contraction effect of the threshold function on singular values is too strong, which may lead to excessive suppression of useful signals and loss of important low-rank structural information; conversely, if a is too large, the threshold effect weakens, and the model's low-rank constraint capability decreases.

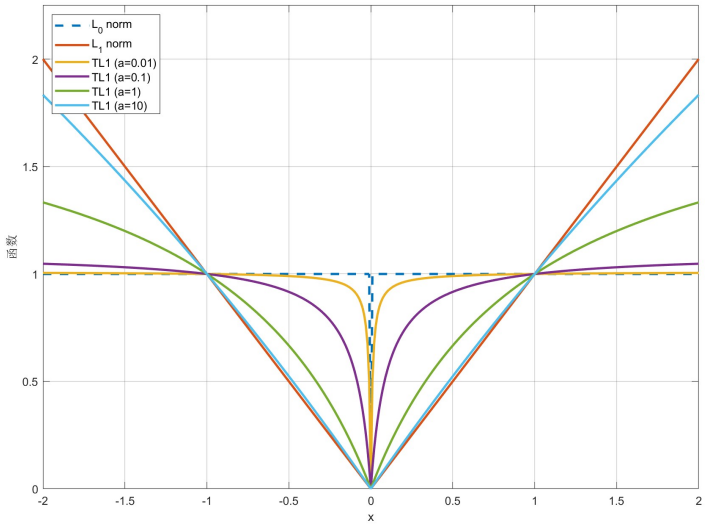


Figure 5. Illustration of the TL1 function with varying parameter a.

Table 5. Results of PSNR and SSIM for different values of a.

a	0.01	0.1	1	10	15	20	25	30	50	100
PSNR	14.5901	14.5901	14.5901	18.5234	28.5573	28.5898	28.4625	28.3201	27.8517	27.234
SSIM	0.2378	0.2378	0.2378	0.4557	0.8503	0.8561	0.8564	0.8555	0.8507	0.8441
PSNR	12.5436	12.5436	12.5436	13.6798	26.1087	26.2988	26.1014	25.92	25.4711	25.017
SSIM	0.1972	0.1972	0.1972	0.2307	0.7997	0.8327	0.8378	0.8391	0.8368	0.8300

Furthermore, we use the two images from BSD100 with 30% noise to observe the effect of parameter a on PSNR and SSIM values. Figure 6 presents the experimental results under various parameter settings, where a is selected from {0.01, 0.1, 1, 10, 15, 20, 25, 30, 50, 100}. As shown in Figure 6 and Table 5, we observed that PSNR can reach its maximum values when $a = 20$ and the SSIM values are similar when $a = 20, 25$. Therefore, to achieve the best performance, we set $a = 20$ for the 30% noise level in all subsequent experiments. Also, for color images with 20% noise, HSI, and video data, we adopt a similar analysis procedure, and can get better results.

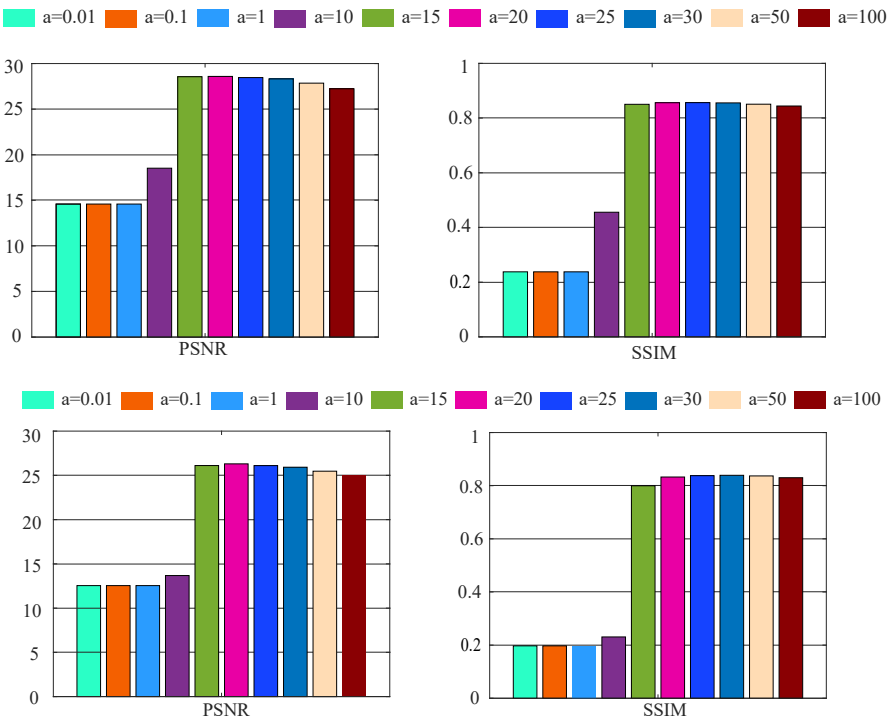


Figure 6. Visual analysis of the impact of parameter a.

7. Conclusion

In this paper, to improve upon the limitations of TRPCA, we propose a novel non-convex optimization model by adopting the TL1 function as a non-convex surrogate for the TNN in TRPCA. The TL1 penalty employs a nonlinear strategy that avoids over-penalizing larger singular values while being more aggressive on small ones. The proposed approach provides a more effective approximation of the low-rank tensor rank function compared to the traditional TNN formulation. To solve the optimization problem efficiently, the ADMM is employed, which yields closed-form solutions for each subproblem and ensures algorithmic stability. Extensive experiments demonstrate that the proposed method significantly outperforms the other three methods in terms of restoration quality. However, our proposed method has two main limitations. First, the parameter a introduced in the TL1 function governs the strength of the non-convex penalty, and its optimal value is highly dependent on the noise level and data distribution. Although we provide a recommended value of a through experiments, there is currently no general criterion for adaptively determining this parameter. Second, the non-convexity of TL1 leads to multiple local minima in the optimization objective. Although theoretical analysis shows that the algorithm can converge to a stationary point satisfying the KKT conditions, global optimality is not guaranteed. In future work, we will further explore adaptive parameter adjustment strategies (such as a data-driven method) to reduce dependence of the method on parameters and discuss the global convergence.

Author contributions

Xiao Guo: Writing–review & editing, Writing–original draft, Methodology. Chuanpei Xu: Writing–review & editing, Writing–original draft, Resources, Supervision, Conceptualization. Zhibin Zhu: Writing–review & editing, Writing–original draft, Conceptualization, Funding acquisition. Benxin Zhang: Writing–review & editing, Writing–original draft, Resources, Funding acquisition.

Use of Generative-AI tools declaration

The author(s) declare(s) they have used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 12561055 and 12561054, in part by Guangxi Natural Science Foundation under Grant No. 2024GXNSFAA010512, and in part by the Guangxi Key Research and Development Program under Grant NO. GUIKE-AB25069275.

Conflict of interest

The authors declare that they have no competing interests.

References

1. C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, S. Yan, Tensor robust principal component analysis with a new tensor nuclear norm, *IEEE Trans. Pattern Anal. Mach. Intell.*, **42** (2020), 925–938. https://doi.org/10.1109/TPAMI.2019.2891760
2. E. J. Candès, X. Li, Y. Ma, J. Wright, Robust principal component analysis? *J. ACM*, **58** (2011), 1–37. https://doi.org/10.1145/1970392.1970395
3. T. Yokota, Q. Zhao, A. Cichocki, Smooth parafac decomposition for tensor completion, *IEEE Trans. Signal Process.*, **64** (2016), 5423–5436. https://doi.org/10.1109/TSP.2016.2586759
4. T. G. Kolda, B. W. Bader, Tensor decompositions and applications, *SIAM Rev.*, **51** (2009), 455–500. https://doi.org/10.1137/07070111X
5. L. R. Tucker, Some mathematical notes on three-mode factor analysis, *Psychometrika*, **31** (1966), 279–311. https://doi.org/10.1007/BF02289464
6. J. Liu, P. Musialski, P. Wonka, J. Ye, Tensor completion for estimating missing values in visual data, *IEEE Trans. Pattern Anal. Mach. Intell.*, **35** (2013), 208–220. https://doi.org/10.1109/TPAMI.2012.39
7. L. Zhang, Z. Peng, Infrared small target detection based on partial sum of the tensor nuclear norm, *Remote Sens.*, **11** (2019), 382. https://doi.org/10.3390/rs11040382

8. T. X. Jiang, T. Z. Huang, X. L. Zhao, L. J. Deng, Multi-dimensional imaging data recovery via minimizing the partial sum of tubal nuclear norm, *J. Comput. Appl. Math.*, **372** (2020), 112680. https://doi.org/10.1016/j.cam.2019.112680

9. Y. Mu, P. Wang, L. Lu, X. Zhang, L. Qi, Weighted tensor nuclear norm minimization for tensor completion using tensor-SVD, *Pattern Recognit. Lett.*, **130** (2020), 4–11. https://doi.org/10.1016/j.patrec.2018.12.012

10. F. Zhang, J. Wang, W. Wang, C. Xu, Low-tubal-rank plus sparse tensor recovery with prior subspace information, *IEEE Trans. Pattern Anal. Mach. Intell.*, **43** (2021), 3492–3507. https://doi.org/10.1109/TPAMI.2020.2986773

11. H. Wang, F. Zhang, J. Wang, T. Huang, J. Huang, X. Liu, Generalized nonconvex approach for low-tubal-rank tensor recovery, *IEEE Trans. Neural Networks Learn. Syst.*, **33** (2022), 3305–3319. https://doi.org/10.1109/TNNLS.2021.3051650

12. Q. Gao, P. Zhang, W. Xia, D. Xie, X. Gao, D. Tao, Enhanced tensor RPCA and its application, *IEEE Trans. Pattern Anal. Mach. Intell.*, **43** (2021), 2133–2140. https://doi.org/10.1109/TPAMI.2020.3017672

13. M. Yang, Q. Luo, W. Li, M. Xiao, Nonconvex 3D array image data recovery and pattern recognition under tensor framework, *Pattern Recognit.*, **122** (2022), 108311. https://doi.org/10.1016/j.patcog.2021.108311

14. Y. Yang, Y. Feng, J. A. Suykens, Robust low-rank tensor recovery with regularized redescending M-estimator, *IEEE Trans. Neural Networks Learn. Syst.*, **27** (2016), 1933–1946. https://doi.org/10.1109/TNNLS.2015.2465178

15. Y. Li, K. Shang, Z. Huang, Low Tucker rank tensor recovery via ADMM based on exact and inexact iteratively reweighted algorithms, *J. Comput. Appl. Math.*, **331** (2018), 64–81. https://doi.org/10.1016/j.cam.2017.09.029

16. T. X. Jiang, T. Z. Huang, X. L. Zhao, L. J. Deng, A novel nonconvex approach to recover the low-tubal-rank tensor data: when t-svd meets PSSV, *arXiv preprint* arXiv:1712.05870, 2017.

17. W. H. Xu, X. L. Zhao, T. Y. Ji, J. Q. Miao, T. H. Ma, S. Wang, et al., Laplace function based nonconvex surrogate for low-rank tensor completion, *Signal Process-Image*, **73** (2019), 62–69. https://doi.org/10.1016/j.image.2018.11.007

18. T. Li, J. Ma, T-SVD based non-convex tensor completion and robust principal component analysis, *Proc. ICPR*, (2021), 6980–6987.

19. Q. L. Zhou, H. J. Che, W. Guo, X. He, M. F. Leung, S. P. Wen, Robust low-rank tensor constrained orthogonal symmetric non-negative matrix factorization for multi-layer networks community detection, *IEEE Trans. Emerg. Top. Comput. Intell.*, **9** (2025), 4145–4160. https://doi.org/10.1109/TETCI.2025.3572129

20. X. Y. Pu, H. J. Che, B. C. Pan, M. F. Leung, S. Wen, Robust weighted low-rank tensor approximation for multiview clustering with mixed noise, *IEEE Trans. Comput. Soc. Syst.*, **11** (2024), 3268–3285. https://doi.org/10.1109/TCSS.2023.3331366

21. J. Lv, Y. Fan, A unified approach to model selection and sparse recovery using regularized least squares, *Ann. Stat.*, **37** (2009), 3498–3528. https://doi.org/10.1214/09-AOS683

22. H. A. Le Thi, B. N. Thi, H. M. Le, Sparse signal recovery by difference of convex functions algorithms, *ACIDS*, 2013, 387–397. https://doi.org/10.1007/978-3-642-36543-0_40

23. M. Nikolova, Local strong homogeneity of a regularized estimator, *SIAM J. Appl. Math.*, **61** (2000), 633–658. https://doi.org/10.1137/s0036139997327794

24. S. Zhang, J. Xin, Minimization of transformed L_1 penalty: closed form representation and iterative thresholding algorithms, *Commun. Math. Sci.*, **15** (2017), 511–537. https://doi.org/10.4310/CMS.2017.V15.N2.A9

25. S. Zhang, J. Xin, Minimization of transformed L_1 penalty: theory, difference of convex function algorithm, and robust application in compressed sensing, *Math. Program.*, **169** (2018), 307–336. https://doi.org/10.1007/s10107-018-1236-x

26. S. Zhang, P. Yin, J. Xin, Transformed Schatten-1 iterative thresholding algorithms for low rank matrix completion, *Commun. Math. Sci.*, **15** (2017), 839–862. https://doi.org/10.4310/CMS.2017.v15.n3.a12

27. X. You, N. Cao, W. Wang, An MTL1TV non-convex regularization model for MR image reconstruction using the alternating direction method of multipliers, *Electron Res. Arch.*, **32** (2024), 3433–3456. https://doi.org/10.3934/era.2024159

28. M. E. Kilmer, C. D. Martin, Factorization strategies for third-order tensors, *Linear Algebra Appl.*, **435** (2011), 641–658. https://doi.org/10.1016/j.laa.2010.09.020

29. C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, S. Yan, Tensor robust principal component analysis: Exact recovery of corrupted low-rank tensors via convex optimization, *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, 5249–5257.

30. M. E. Kilmer, K. Braman, N. Hao, R. C. Hoover, Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging, *SIAM J. Matrix Anal. Appl.*, **34** (2013), 148–172. https://doi.org/10.1137/110837711

31. D. R. Han, A survey on some recent developments of alternating direction method of multipliers, *J. Oper. Res. Soc.*, **10** (2022), 1–52. https://doi.org/10.1007/s40305-021-00368-3

32. Y. Wang, W. Yin, J. Zeng, Global convergence of ADMM in nonconvex nonsmooth optimization, *J. Sci. Comput.*, **78** (2019), 29–63. https://doi.org/10.1007/s10915-018-0757-z

33. J. You, Y. Jiao, X. Lu, T. Zeng, A nonconvex model with minimax concave penalty for image restoration, *J. Sci. Comput.*, **78** (2019), 1063–1086. https://doi.org/10.1007/s10915-018-0801-z

34. Y. Mei, C. Luo, J. Cao, Z. Liu, Y. Fei, F. Kong, et al., Intelligent recovery of low-rank sparse tensor for noisy hydroacoustic with use of nonconvex regularization, *Digit. Signal Process*, **173** (2026), 105927. https://doi.org/10.1016/j.dsp.2026.105927

35. S. Cai, Q. Luo , M. Yang, W. Li, M. Xiao, Tensor robust principal component analysis via non-convex low rank approximation, *Appl. Sci.*, **9** (2019), 1411. https://doi.org/10.3390/app9071411

36. G. Li, T. K. Pong, Calculus of the exponent of Kurdyka-Lojasiewicz inequality and its applications to linear convergence of first-order methods, *Found Comput. Math.*, **18** (2018), 1199–1232. https://doi.org/10.1007/s10208-017-9366-8

37. Y. Zhou, Y. M. Cheung, Bayesian low-tubal-rank robust tensor factorization with multi-rank determination, *IEEE Trans. Pattern Anal. Mach. Intell.*, **43** (2021), 62–76. https://doi.org/10.1109/TPAMI.2019.2923240

38. D. Martin, C. Fowlkes, D. Tal, J. Malik, A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, *Proc. IEEE Int. Conf. Comput. Vis.*, 2001, 416–423. https://doi.org/10.1109/ICCV.2001.937655

AIMS Press