Rician Noise Removal via Weighted Nuclear Norm Penalization

Jian Lu[a,b], Jiapeng Tian[a], Qingtang Jiang[c], Xiaoxia Liu[a], Zhenwei Hu[a], Yuru Zou[a*]

[a] *Shenzhen Key Laboratory of Advanced Machine Learning and Applications, College of Mathematics and Statistics, Shenzhen University, Shenzhen 518060, P.R. China*

[b] *Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen 518060, P.R. China*

[c] *Department of Mathematics and Computer Science, University of Missouri-St. Louis, St. Louis 63121, USA*

Abstract

Magnetic Resonance Imaging (MRI) is one of the most important techniques in medical imaging and Rician noise is a common noise that naturally appears in MRI images. Low rank matrix approximation approaches have been widely used in image processing such as image denoising, which takes advantage of the idea of non-local self-similarity between patches in a natural image. The weighted nuclear norm minimization method as a low rank matrix approximation approach has shown to be an effective approach for image denoising. Inspired by this, we propose in this paper a maximum a posteriori (MAP) model with the weighted nuclear norm as a regularization constraint to remove Rician noise. The MAP data fidelity term has a Lipschitz continuous gradient and the weighted nuclear norm can be efficiently minimized. We propose an iterative weighted nuclear norm minimization algorithm (IWNNM) to solve the proposed non-convex model and analyze the convergence of our algorithm. The computational results show that our proposed method is promising in restoring images corrupted with Rician noise.

Key Words: Rician noise removal, low rank, weighted nuclear norm.

1 Introduction

As Magnetic Resonance Imaging (MRI) plays a more and more important role in medical imaging, Rician noise that frequently presents in MRI images attracts more and more attention. In this paper, we mainly propose a new model to efficiently remove Rician noise. Mathematically, the image y degraded by Rician noise can be given by

$$y = \sqrt{(x+\eta_1)^2 + \eta_2^2}, \tag{1}$$

where x is the original image, $\eta_1, \eta_2 \sim N(0, \sigma^2)$ and σ^2 is the noise variance.

In recent years, there have been many methods proposed for denoising images contaminated by Rician noise, such as filtering methods, wavelet methods, convex optimization methods and non-convex optimization methods. Based on the anisotropic diffusion process that proposed by Perona and Malik [20], Gerig et al. [11] presented the nonlinear anisotropic filtering method for Rician noise removal with the edge information preserved, but some other details may be lost. Studying the non-local means algorithm, Prima et al. [21] proposed a non-local means variants method for denoising the diffusion-weighted and diffusion tensor MRI images. And Manjón et al. [17] also improved a new filter (we call it NLM model) which is based on the Non-Local Means to reduce the noise in MRI images. Inspired by it, Wiest-Daessle et al. [25] adapted the non-local means filter to data corrupted by Rician noise, and presented non-local means filtering method for better respecting the image details and structures. In [19], Nowak came up with wavelet domain filtering method which adapts to variations in both the signal and the noise. And in [26], Wood and Johnson proposed

[*] Corresponding author. E-mail: yuruzou@szu.edu.cn (Y. Zou).

wavelet packet denoising method for Rician noise removal at low signal-to-noise ratio (SNR). Wavelet methods have been efficient to remove Rician noise with the image details and edge preserved, but the problem that small dots influence image analysis process remains unresolved. In the meantime, the maximum a posteriori (MAP) estimation model was proposed, which is considered from the feature of the noise-free image. The MAP model with total variation (TV) regularization constraint is as follows

$$\min_x \frac{1}{2\sigma^2}\int_\Omega x^2 dt - \int_\Omega \log B_0\left(\frac{xy}{\sigma^2}\right)dt + \beta\int_\Omega |Dx|dt, \tag{2}$$

where the first two terms represent the data fidelity term, the last term represents the TV regularization term. Here, B_0 is the modified Bessel function of the first kind with order zero [3] defined by

$$B_0(x) = \frac{1}{\pi}\int_0^\pi e^{x\cos\theta}d\theta. \tag{3}$$

One challenge with problem (2) is that the objective function is a non-convex function, which leads to a difficult problem to solve. In view of the MAP model, GTV model [12] was put forward by Getreuer et al., which is a convex approximation of the MAP model and can be easily solved, but the fidelity item of GTV model is a complicated piecewise function. Recently, Chen and Zeng [7] (we call it CZ model) proposed a new convex model which added a statistical property of Ricain noise into the MAP model, leading to a new strictly convex model under mild condition that can be easily solved by primal-dual algorithm. Very recently, based on the CZ model, Chen et al. [8] put forward a new model (called the C-KSVD model below) which adds sparse representation and dictionary learning terms into the CZ model.

In the last decade, many non-local based methods have been proposed for image denoising. They exploit the redundancy of non-local similar patches and achieve great denoising results. One of the well-known non-local based methods is the block matching 3-D (BM3D) algorithm [9]. To address Rician noise, Foi [10] developed forward and inverse variance-stabilizing transformations (VSTs) and VSTs can work with BM3D to efficiently remove Rician noise.

Lately, low rank matrix approximation approaches (LRMA) have shown great performance in image denoising. For example, see [4, 5, 22, 23, 13, 14]. The idea behind this is that the matrix formed by non-local similar patches in a natural image is of low rank. The LRMA method using nuclear norm penalization is one of the most commonly used methods. However, this method, like many other methods, ignores some prior knowledge of singular values, and hence it is not flexible to deal with many real problems. To improve the flexibility of nuclear norm minimization method, Gu et al. [13, 14] proposed the weighted nuclear norm minimization method for image denoising. Also, Nie et al. [18] replaced the nuclear norm with the Schatten-p norm to regularize the low rank problem. When $p \to 0$, the Schatten-0 norm is equivalent to the rank of matrix, and when $p \to 1$, the Schatten-1 norm is equivalent to the nuclear norm. Inspired by the weighted nuclear norm minimization method, in this paper, we incorporate the weighted nuclear norm into the MAP model for Rician noise removal.

The following is the outline of our paper. In section 2, we first briefly introduce low rank matrix approximation problems including the weighted nuclear norm minimization. After that we propose our model by replacing the total variation regularization constraint with the weighted nuclear norm as to the MAP model. Then we propose an iterative weighted nuclear norm minimization algorithm (IWNNM) for solving our proposed model. In section 3, we demonstrate with numerical results that our model outperforms other methods with application to Rician noise removal. In the end, we draw our conclusion in section 4.

2 MAP model with weighted nuclear norm penalization

In this section, we first give a review of low rank matrix approximation problems. Then we propose a MAP model with the weighted nuclear norm penalization and an iterative weighted nuclear norm minimization algorithm (IWNNM) to solve the proposed model.

2.1 Low rank matrix approximation

Rank minimization problem can be generally formulated as

$$\begin{aligned} &\min_{X}\ \operatorname{rank}(X) \\ &s.t.\ \ X \in \mathcal{C}, \end{aligned} \tag{4}$$

where $\operatorname{rank}(\cdot)$ is the rank of the given matrix, $X \in \mathbb{R}^{M\times N}$ is the unknown low rank matrix, and $\mathcal{C}$ is a convex set. The problem (4) is NP-hard and non-convex, so it is very difficult to solve.

In low rank matrix approximation problems, the rank function is frequently replaced by its tightest convex surrogate, which is the nuclear norm defined as the sum of its singular values, i.e.,

$$\|X\|_* = \sum_{i=1}^{\ell} \delta_i(X),$$

where $\delta_i(X)$ is the i-th singular value of X and $\ell = \min\{M, N\}$. We assume that the singular values of X are in a non-increasing order, namely, $\delta_1(X) \geq \delta_2(X) \geq \cdots \geq \delta_r(X) > \delta_{r+1}(X) = \cdots = \delta_\ell(X) = 0$, where $r = \operatorname{rank}(X)$.

Candès and Recht [6] proved that the rank minimization problem for matrix completion can be converted into a nuclear norm minimization problem under certain conditions. An unconstrained nuclear norm minimization problem is as follows

$$\hat{X} = \operatorname*{argmin}_{X} \frac{1}{2}\|Y - X\|_F^2 + \lambda\|X\|_*, \tag{5}$$

where $\|X\|_F = \sqrt{\sum_{i,j} |X_{i,j}|^2}$ is the Frobenius norm of X, and λ is a positive constant.

Cai et al. [5] showed that problem (5) can be solved by a soft-thresholding operation on the singular values of the observation matrix Y, and has a closed form solution as follows

$$\hat{X} = U S_\lambda(\Sigma) V^T,$$

where $Y = U\Sigma V^T$ is the singular value decomposition (SVD) of Y, and $S_\lambda(\Sigma)$ is the soft-thresholding function on diagonal matrix Σ with parameter λ. That is, for diagonal matrix Σ with diagonal element Σ_{ii}, $S_\lambda(\Sigma)$ is a diagonal matrix with the (i, i)-entry $S_\lambda(\Sigma)_{ii}$ given by

$$S_\lambda(\Sigma)_{ii} = \max(\Sigma_{ii} - \lambda, 0).$$

Although problem (5) can be explicitly computed and has been widely used for low rank matrix approximation problems, the nuclear norm ignored some prior knowledge of the singular values by shrinking each singular value with the same parameter λ, and it also cannot be flexibly applied to many real problems. To overcome this drawback, Gu et al. [13] introduced the weighted nuclear norm

$$\|X\|_{\omega,*} = \sum_{i=1}^{\ell} \omega_i \delta_i(X), \quad \omega = [\omega_1, \cdots, \omega_\ell], \quad \omega_i \geq 0,$$

and proposed the following weighted nuclear norm minimization method for image denoising

$$\hat{X} = \operatorname*{argmin}_{X} \frac{1}{2}\|Y - X\|_F^2 + \|X\|_{\omega,*}. \tag{6}$$

For natural images, we have the general prior knowledge that the larger singular values of X are more important than the smaller ones since they represent the energy of the major components of X. Thus, to shrink less on the larger singular values, it is natural to assume $0 < \omega_1 \leq \omega_2 \leq \cdots \leq \omega_\ell$. And the solution of problem (6) with such ω can be computed using the following result from Corollary 1 in [13] and a special case of Theorem 3.1 in [16].

Theorem 2.1. [13, 16] *If $\omega = [\omega_1, \cdots, \omega_\ell]$ satisfies $0 < \omega_1 \leq \omega_2 \leq \cdots \leq \omega_\ell$, then problem* (6) *has the global optimal solution which is given as follows:*

$$\hat{X} = US_\omega(\Sigma)V^T,$$

where $Y = U\Sigma V^T$ is the SVD of Y, and $S_\omega(\Sigma)$ is the generalized soft-thresholding operator of the diagonal matrix Σ with respect to ω with the (i, i)-entry of diagonal matrix $S_\omega(\Sigma)$ given by

$$S_\omega(\Sigma)_{ii} = \max\left(\Sigma_{ii} - \omega_i, 0\right).$$

2.2 Our proposed model

To remove Rician noise, we take advantage of the non-local self-similarity of natural images and apply a low rank matrix approximation (LRMA) method to image patches. An image $y \in \mathbb{R}^d$ is divided into overlapped patches of size $\sqrt{m} \times \sqrt{m}$ denoted by the vector $y_i \in \mathbb{R}^m$. Then for a local patch y_j in image y, we search for a non-local similar patch in a relatively large area around it by methods such as block matching [9] that uses the Euclidean distance as a block similarity measure. By stacking those non-local similar patches into a matrix, denoted by $Y_j \in \mathbb{R}^{m\times n}$, we have

$$Y_j = X_j + N_j,$$

where X_j and N_j are the original clean patch matrix and the corresponding corrupted matrix, respectively. Intuitively, X_j should be a low rank matrix, and LRMA methods can be used to estimate X_j from Y_j. The entire image can be reconstructed by aggregating all estimated patches Y_j.

Inspired by the facts that the MAP-based models are particularly efficient for Rician noise removal and that the low rank approximation method provides a powerful tool for image denoising in general, we propose in this paper a new MAP model with the weighted nuclear norm penalization for Rician noise removal and apply the proposed model to estimate X_j from Y_j as follows

$$\hat{X}_j = \underset{X_j}{\operatorname{argmin}} \frac{1}{2\sigma^2}\left\|X_j\right\|_F^2 - \left\langle \log B_0\left(\frac{X_j Y_j}{\sigma^2}\right), \mathbf{1}\right\rangle + \|X_j\|_{\omega_j,*}, \tag{7}$$

where the inner product on the right-hand side of (7) denotes $\sum_{i,k} \log B_0\left((X_j)_{ik}(Y_j)_{ik}/\sigma^2\right)$, and ω_j is a given weight vector with $0 < (\omega_j)_1 \leq (\omega_j)_2 \leq \cdots \leq (\omega_j)_\ell$. Here and below, we use $(\omega_j)_i$ to denote the i-th component of a vector ω_j.

To further analyze model (7), we rewrite the proposed model as follows

$$\min_{X_j} \; F_j(X_j) := f_j(X_j) + \Psi(X_j), \tag{8}$$

where the MAP-based data fidelity term is

$$f_j(X_j) = \frac{1}{2\sigma^2}\left\|X_j\right\|_F^2 - \langle \log B_0\left(\frac{X_j Y_j}{\sigma^2}\right), \mathbf{1}\rangle, \tag{9}$$

and the weighted nuclear norm penalization is

$$\Psi(X_j) = \psi \circ \delta(X_j), \tag{10}$$

with $\delta(X_j) = (\delta_1(X), \delta_2(X), \cdots, \delta_\ell(X_j))$, $\psi(x) = \sum\limits_{i=1}^{\ell} \omega_i \psi_i(x)$, $\ell = \min\{m, n\}$, and $\psi_i(x)$ is the i-th largest element of $\{|x_1|, |x_2|, \cdots, |x_\ell|\}$, for all $x \in \mathbb{R}^\ell$.

Model (8) is a non-convex optimization problem, which may have no solution or is challenging to solve in general. Fortunately, the function f_j is lower bounded, which leads to the existence of minimizers. Also, the function f_j is smooth with a Lipschitz-continuous gradient, so efficient algorithms can be developed to solve the proposed model. To proceed, we verify the lower boundedness and smoothness of the function f_j in the following lemma and proposition.

Lemma 2.2. *Suppose $X, Y \in \mathbb{R}^{m\times n}$, Y is bounded on Ω. Then*

$$f(X) = \frac{1}{2\sigma^2}\|X\|_F^2 - \left\langle \log B_0\left(\frac{XY}{\sigma^2}\right), \mathbf{1}\right\rangle \tag{11}$$

has a lower bound for all $X \in \mathbb{R}^{m\times n}$, that is, $\inf_{X\in\mathbb{R}^{m\times n}} f(X) > -\infty$.

Proof. First, we consider $g(x,y) = \frac{1}{2\sigma^2}x^2 - \log B_0\left(\frac{xy}{\sigma^2}\right)$, $x, y \in \mathbb{R}$. Since

$$B_0(x) = \frac{1}{\pi}\int_0^\pi e^{x\cos\theta}d\theta \le \frac{1}{\pi}\int_0^\pi e^{|x|}d\theta \le e^{|x|},$$

we have

$$g(x,y) \ge \frac{1}{2\sigma^2}x^2 - \log\left(e^{\frac{|xy|}{\sigma^2}}\right) = \frac{1}{2\sigma^2}x^2 - \frac{|xy|}{\sigma^2} \ge -\frac{1}{2\sigma^2}y^2,$$

for any $x \in \mathbb{R}$. Hence for any $X \in \mathbb{R}^{m\times n}$, we have

$$\begin{aligned} f(X) &= \frac{1}{2\sigma^2}\|X\|_F^2 - \left\langle \log B_0\left(\frac{XY}{\sigma^2}\right), \mathbf{1}\right\rangle \\ &= \sum_{i=1}^m\sum_{j=1}^n \left(\frac{1}{2\sigma^2}x_{ij}^2 - \log B_0\left(\frac{x_{ij}y_{ij}}{\sigma^2}\right)\right) \\ &= \sum_{i=1}^m\sum_{j=1}^n g\left(x_{ij}, y_{ij}\right) \\ &\ge -\frac{1}{2\sigma^2}\sum_{i=1}^m\sum_{j=1}^n y_{ij}^2. \end{aligned}$$

This proves $\inf_{X\in\mathbb{R}^{m\times n}} f(X) > -\infty$. □

Clearly, $f(X)$ defined by (11) is smooth with

$$\frac{\partial f(X)}{\partial x_{ij}} = \frac{1}{\sigma^2}x_{ij} - \frac{y_{ij}}{\sigma^2}\frac{B_0'(\frac{x_{ij}y_{ij}}{\sigma^2})}{B_0(\frac{x_{ij}y_{ij}}{\sigma^2})}.$$

Next, we show that it has a Lipschitz-continuous gradient in $\mathbb{R}^{m\times n}$.

Proposition 2.3. *Suppose $X, Y \in \mathbb{R}^{m\times n}$, Y is bounded on Ω. Then $f(X)$ defined by* (11) *has Lipschitz-continuous gradient in $\mathbb{R}^{m\times n}$ with Lipschitz constant $L_f = \frac{1}{\sigma^2}$.*

Proof. Denote $h(x) = \log B_0(x)$, $x \in \mathbb{R}$. Then

$$h''(x) = \frac{B_0''(x)B_0(x) - (B_0'(x))^2}{B_0^2(x)}.$$

By the definition of $B_0(x)$ in (3), we have

$$B_0'(x) = \frac{1}{\pi}\int_0^\pi \cos\theta e^{x\cos\theta}d\theta \quad \text{and} \quad B_0''(x) = \frac{1}{\pi}\int_0^\pi \cos^2\theta e^{x\cos\theta}d\theta.$$

By Cauchy-Schwarz inequality, we have

$$\begin{aligned}
&(B_0'(x))^2 = \left(\frac{1}{\pi}\int_0^\pi \cos\theta e^{x\cos\theta} d\theta\right)^2 \\
\le& \frac{1}{\pi}\int_0^\pi \cos^2\theta e^{x\cos\theta} d\theta \cdot \frac{1}{\pi}\int_0^\pi e^{x\cos\theta} d\theta \\
=& B_0''(x) B_0(x).
\end{aligned}$$

This shows that $h''(x) \ge 0$. Clearly, $\frac{\partial^2 f(X)}{\partial x_{ij}\partial x_{i'j'}} = 0$ if $(i,j) \neq (i',j')$. In addition, we have

$$\frac{\partial^2 f(X)}{\partial x_{ij}^2} = \frac{1}{\sigma^2} - \frac{\partial^2}{\partial x_{ij}^2}\left(h\left(x_{ij}y_{ij}\right)\right) = \frac{1}{\sigma^2} - h''\left(x_{ij}y_{ij}\right) y_{ij}^2 \le \frac{1}{\sigma^2}.$$

Thus,

$$\|\nabla^2 f(X)\|_F \le \frac{1}{\sigma^2}.$$

With $L_f = \frac{1}{\sigma^2}$, we have

$$\|\nabla f(X_1) - \nabla f(X_2)\|_F \le L_f \|X_1 - X_2\|_F, \ \ \forall X_1, X_2 \in \mathbb{R}^{m\times n}.$$

This proves that $\nabla f(X)$ is of L_f-Lipschitz-continuous, as desired. □

2.3 Stationary points of the proposed model

In model (8), the function $f_j(X_j)$ has a Lipschitz-continuous gradient, so we only need to consider $\Psi(X_j)$ for characterizing the stationary points of the proposed model. Before that, we give some notations. The matrices that we consider are in $\mathbb{R}^{m\times n}$ and ℓ is the smaller number between m and n as stated above. For a vector $a = (a_1, \cdots a_\ell) \in \mathbb{R}^\ell$, we let $\operatorname{Diag}(a)$ denote the diagonal matrix with a_i's as its main diagonal entries:

$$\operatorname{Diag}(a) := \begin{bmatrix} a_1 & & \\ & \ddots & \\ & & a_\ell \end{bmatrix}.$$

Throughout this paper, unless it is stated otherwise, a singular value decomposition (SVD) of a matrix $X \in \mathbb{R}^{m\times n}$ means $X = U\Sigma V^T$, where $U \in \mathbb{R}^{m\times\ell}$,$V \in \mathbb{R}^{n\times\ell}$ with $U^TU = V^TV = I$, and Σ is the diagonal matrix with ordered singular values $\delta_i(X)$ of X as its main diagonal entries. That is, with $\delta(X) := (\delta_1(X), \cdots, \delta_\ell(X))$, we have $\Sigma = \operatorname{Diag}(\delta(X))$. We also write the matrices U, V, Σ resulted from the SVD of X as

$$[U, \Sigma, V] = SVD(X).$$

The pair (U, V) of matrices with orthonormal columns for SVD of X is not unique. We let $\mathcal{M}(X)$ denote the set of all such pairs (U, V):

$$\mathcal{M}(X) := \left\{(U,V) \in \mathbb{R}^{m\times\ell} \times \mathbb{R}^{n\times\ell} : U^TU = V^TV = I, X = U \operatorname{Diag}(\delta(X))V^T\right\}.$$

Now, we are ready to characterize the Clarke subdifferential of the function Ψ defined by (10) and the stationary points of model (8). First, according to Theorem 3.7 of [15], the Clarke subdifferential of Ψ at X_j is given by

$$\partial\Psi(X_j) = \{U_j \operatorname{Diag}(d)V_j^T : d \in \partial\psi(\delta(X_j)), (U_j, V_j) \in \mathcal{M}(X_j)\},$$

where $\partial\psi$ is the Clarke subdifferential of ψ.

We say $X_j^* \in \mathbb{R}^{m\times n}$ is a first-order stationary point of problem (8) if

$$0 \in \left\{U_j^T \nabla f_j(X_j^*)V_j + \operatorname{Diag}(d^*) : (U_j, V_j) \in \mathcal{M}(X_j^*), d^* \in \partial\psi(\delta(X_j^*))\right\}. \tag{12}$$

It is shown in the next theorem that a local minimizer of problem (8) is a first-order stationary point.

Theorem 2.4. *Suppose that X_j^* is a local minimizer of problem* (8)*. Then X_j^* is a first-order stationary point of problem* (8)*, that is,* (12) *holds at X_j^*.*

Proof. Let $X_j^* = \bar{U}_j \operatorname{Diag}(\delta(X_j^*))\bar{V}_j^T$ for some $(\bar{U}_j, \bar{V}_j) \in \mathcal{M}(X_j^*)$. By the assumption that X_j^* is a local minimizer of problem (8), one can see that 0 is a local minimizer of the problem

$$\min_{Z\in\mathbb{R}^{\ell\times\ell}} f_j(X_j^* + \bar{U}_j Z \bar{V}_j^T) + \Psi(X_j^* + \bar{U}_j Z \bar{V}_j^T).$$

This, together with the fact

$$\Psi\left(X_j^* + \bar{U}_j Z \bar{V}_j^T\right) = \Psi\left(\bar{U}_j \operatorname{Diag}\left(\delta\left(X_j^*\right)\right)\bar{V}_j^T + \bar{U}_j Z \bar{V}_j^T\right) = \Psi\left(\operatorname{Diag}\left(\delta\left(X_j^*\right)\right) + Z\right),$$

implies that 0 is a local minimizer of the problem

$$\min_{Z\in\mathbb{R}^{\ell\times\ell}} \underbrace{f_j(X_j^* + \bar{U}_j Z \bar{V}_j^T) + \Psi(\operatorname{Diag}(\delta(X_j^*)) + Z)}_{S(Z)}. \tag{13}$$

By Theorem 3.7 of [15] and the definition of $\mathcal{M}(\cdot)$, the Clarke subdifferential of $S(Z)$ at $Z = 0$ is given by

$$\partial S(0) = \{\bar{U}_j^T \nabla f_j(X_j^*)\bar{V}_j + \operatorname{Diag}(d^*) : d^* \in \partial\psi(\delta(X_j^*))\}.$$

Since 0 is a local minimizer of problem (13), the first-order optimality condition of (13) yields $0 \in \partial S(0)$. Hence, there exists some $d^* \in \partial\psi(\delta(X_j^*))$ such that

$$\bar{U}_j^T \nabla f_j(X_j^*)\bar{V}_j + \operatorname{Diag}(d^*) = 0, \tag{14}$$

which implies that (12) holds at X_j^* with $U_j = \bar{U}_j, V_j = \bar{V}_j$. □

2.4 Iterative weighted nuclear norm minimization algorithm

In this section, we propose an iterative weighted nuclear norm minimization algorithm (IWNNM), which is inspired by the iterative reweighted singular value minimization (IRSVM) algorithm introduced in [16]. IRSVM aims to solve minimization problems of the form as model (8) under the assumption that f_j has a Lipschitz-continuous gradient and Ψ is exactly the Schatten-p quasi-norm. In our proposed model, Ψ is the weighted nuclear norm with weights being fixed. In fact, IRSVM algorithm together with its acceleration strategy works well even with our choice of Ψ. A key step of the IWNNM method is to solve the following problem:

$$X_j^{k+1} = \operatorname*{argmin}_{X_j\in\mathbb{R}^{m\times n}} \langle \nabla f_j(X_j^k), X_j - X_j^k\rangle + \frac{L_j^k}{2}\|X_j - X_j^k\|_F^2 + \|X_j\|_{\omega_j,*}, \tag{15}$$

where $\nabla f_j(X_j)$ is the gradient of function $f_j(X_j)$, L_j^k is a positive parameter, X_j^k is a matrix with size $m\times n$, and ω_j with $0 \le (\omega_j)_1 \le \cdots \le (\omega_j)_\ell$ is the given weight vector.

We observe that problem (15) is equivalent to

$$X_j^{k+1} = \operatorname*{argmin}_{X_j\in\mathbb{R}^{m\times n}} \frac{L_j^k}{2}\left\|X_j - \left(X_j^k - \frac{\nabla f_j(X_j^k)}{L_j^k}\right)\right\|_F^2 + \|X_j\|_{\omega_j,*}. \tag{16}$$

Let $Z_j^k = X_j^k - \nabla f_j\left(X_j^k\right)/L_j^k$, and $[U, \Sigma, V] = SVD\left(Z_j^k\right)$. Then, according to Theorem 2.1, the minimizer of (16) is given by

$$X_j^{k+1} = US_{\omega_j/L_j^k}(\Sigma)V^T.$$

Next we provide our algorithm in Algorithm 1.

Algorithm 1: Iterative Weighted Nuclear Norm Minimization Algorithm

```
Input: Degraded image patch group $Y_j$, weight vector $\omega_j$ with $0 < (\omega_j)_1 \leq \cdots \leq (\omega_j)_\ell$, $0 < L_{\min} < L_{\max}$, $\tau > 1$, $c > 0$ and integer $N_0 \geq 0$.
Initialization: $X_j^{(0)} = Y_j$, $k = 0$.
while not converge do
    (a) Choose $L_j^k \in [L_{\min}, L_{\max}]$.
    (b) $Z_j^k = X_j^k - \nabla f_j\left(X_j^k\right)/L_j^k$.
    (c) $[U, \Sigma, V] = SVD(Z_j^k)$.
    (d) $X_j^{k+1} = US_{\omega_j/L_j^k}(\Sigma)V^T$.
    (e) If
        $F_j\left(X_j^{k+1}\right) \leq \max_{[k-N_0]^+ \leq l \leq k} F_j\left(X_j^l\right) - \frac{c}{2}\left\|X_j^{k+1} - X_j^k\right\|_F^2$    (17)
    is satisfied, then set $\bar{L}_j^k = L_j^k$, $k \leftarrow k+1$ and go to the next while loop iteration.
    (f) Set $L_j^k \leftarrow \tau L_j^k$ and go to step (b).
end while
Output: $\hat{X}_j = X_j^\infty$.
```

Remark 2.5. *Based on the Proposition* 2.3 *and* [16]*, we can obtain that for each $k \geq 0$, the inner termination criterion* (17) *is satisfied after at most*

$$\max\left\{\left\lfloor \frac{\log(L_f + c) - \log(L_{\min})}{\log(\tau)} + 1 \right\rfloor, 1\right\}$$

inner iterations, where $L_f = \frac{1}{\sigma^2}$ is the Lipschitz constant of ∇f given in Proposition 2.3*.*

Theorem 2.6. *Let $\{X_j^k\}_{k\in\mathbb{N}}$ be a sequence generated by Algorithm 1. Then the following statements hold:*

(i) The sequence $\{X_j^k\}_{k\in\mathbb{N}}$ is bounded;

(ii) Let X_j^* be any accumulation point of $\{X_j^k\}_{k\in\mathbb{N}}$. Then X_j^* is a first-order stationary point of problem (8)*, i.e.,* (12) *holds at X_j^*.*

Proof. (i) Using (17) and an inductive argument, we can conclude that $F_j(X_j^k) \leq F_j(X_j^0)$ for all $k \in \mathbb{N}$. This together with $\underline{f_j} = \inf_{X_j \in \mathbb{R}^{m\times n}} f_j(X_j)$, $(\omega_j)_i > 0$ for $1 \leq i \leq \ell$, and the definition of F_j implies that

$$\underline{f_j} + \|X_j^k\|_{\omega_j,*} \leq f(X_j^k) + \|X_j^k\|_{\omega_j,*} = F_j(X_j^k) \leq F_j(X_j^0).$$

It follows that $(\omega_j)_1\delta_1(X_j^k) \leq \|X_j^k\|_{\omega_j,*} \leq F_j(X_j^0) - \underline{f_j}$ and hence $\{X_j^k\}_{k\in\mathbb{N}}$ is bounded.

(ii) From the proof of statement of (i) , we know that

$$\left\{X_j^k\right\}_{k\in\mathbb{N}} \subset \Omega_j = \left\{X_j \in \mathbb{R}^{m\times n} : \|X_j\|_{\omega,*} \leq F_j(X_j^0) - \underline{f_j}\right\}.$$

Observe that $F_j(X_j)$, viewed as a function of X_j, is uniformly continuous in Ω_j. Using this fact, (17), and a similar argument as used in the proof of Lemma 4 in [27], one can show that $\|X_j^{k+1} - X_j^k\|_F \to 0$ as $k \to \infty$. By the updating rule of Algorithm 1 and Remark 2.5, we can observe that $\{\bar{L}_j^k\}_{k\in\mathbb{N}}$ is bounded.

Let $\bar{Z}_j^k = X_j^k - \nabla f_j(X_j^k)/\bar{L}_j^k$, and let $U_j^k \operatorname{Diag}(d_j^k)(V_j^k)^T$ be the SVD of $\bar{Z}_j^k$, where $U_j^k \in \mathbb{R}^{m\times\ell}, V_j^k \in \mathbb{R}^{n\times\ell}$ and d_j^k consists of all singular values of $\bar{Z}_j^k$ arranged in a descending order. It then follows from Algorithm 1 that X_j^{k+1} is the solution of (16) with $L_j^k = \bar{L}_j^k$ given by

$$X_j^{k+1} = U_j^k \operatorname{Diag}(s_j^{k+1})(V_j^k)^T, \tag{18}$$

where

$$s_j^{k+1} = \max(d_j^k - \omega_j/\bar{L}_j^k, 0). \tag{19}$$

By the properties of d_j^k and ω_j, we see that $(d_j^k)_1 \geq \cdots \geq (d_j^k)_\ell$ and $(\omega_j)_1 \leq \cdots \leq (\omega_j)_\ell$. These together with (19) imply that $(s_j^{k+1})_1 \geq \cdots \geq (s_j^{k+1})_\ell \geq 0$. This relation and (18) yield

$$(s_j^{k+1})_i = \delta_i(X_j^{k+1}), \;\; \forall i = 1, \cdots, \ell. \tag{20}$$

Since X_j^* is an accumulation point of $\{X_j^k\}_{k\in\mathbb{N}}$, there exists a subsequence $\{X_j^k\}_{k\in\mathcal{K}}$ such that $\mathcal{K} \subset \mathbb{N}$ and $\{X_j^k\}_{k\in\mathcal{K}} \to X_j^*$. Due to $\|X_j^{k+1} - X_j^k\|_F \to 0$, we also have $\{X_j^{k+1}\}_{k\in\mathcal{K}} \to X_j^*$. This along with (20) leads to

$$\left\{(s_j^{k+1})_i\right\}_{k\in\mathcal{K}} \to \delta_i(X_j^*), \;\; \forall i = 1, \cdots, \ell. \tag{21}$$

Let $r = \operatorname{rank}\left(X_j^*\right)$. Then one can see from (21) that there exists some $k_0 > 0$ such that $(s_j^{k+1})_i > 0$ for all $1 \leq i \leq r$ and $k \in \mathcal{K}_0 = \{j \in \mathcal{K} : j > k_0\}$. This and (19) lead to that

$$(s_j^{k+1})_i = (d_j^k)_i - (\omega_j)_i/\bar{L}_j^k, \;\; \forall 1 \leq i \leq r, k \in \mathcal{K}_0.$$

Hence,

$$\bar{L}_j^k((s_j^{k+1})_i - (d_j^k)_i) + (\omega_j)_i = 0, \;\; \forall 1 \leq i \leq r, k \in \mathcal{K}_0,$$

which implies that for all $k \in \mathcal{K}_0$,

$$\bar{L}_j^k \sum_{i=1}^r ((s_j^{k+1})_i - (d_j^k)_i)(U_j^k)_i((V_j^k)_i)^T + \sum_{i=1}^r (\omega_j)_i(U_j^k)_i((V_j^k)_i)^T = 0. \tag{22}$$

In the first term on the left hand side of (22), using (18) and $\bar{Z}_j^k = U_j^k \operatorname{Diag}(d_j^k)(V_j^k)^T$, one can see that

$$\sum_{i=1}^r ((s_j^{k+1})_i - (d_j^k)_i)(U_j^k)_i((V_j^k)_i)^T = X_j^{k+1} - \bar{Z}_j^k - \sum_{i=r+1}^{\ell} ((s_j)_i^{k+1} - (d_j)_i^k)(U_j^k)_i((V_j^k)_i)^T,$$

which together with $\bar{Z}_j^k = X_j^k - \nabla f_j(X_j^k)/\bar{L}_k$ yields

$$\sum_{i=1}^r ((s_j^{k+1})_i - (d_j^k)_i)(U_j^k)_i((V_j)_i^k)^T = X_j^{k+1} - X_j^k + \frac{\nabla f_j(X_j^k)}{\bar{L}_k} - \sum_{i=r+1}^{\ell} ((s_j^{k+1})_i - d_i^k)(U_j^k)_i((V_j^k)_i)^T.$$

Substituting it into (22), we can get that for all $k \in \mathcal{K}_0$,

$$\bar{L}_j^k(X_j^{k+1} - X_j^k) + \nabla f_j(X_j^k) - \bar{L}_j^k \sum_{i=r+1}^{\ell} ((s_j^{k+1})_i - (d_j^k)_i)(U_j^k)_i((V_j^k)_i)^T + \sum_{i=1}^r (\omega_j)_i(U_j^k)_i((V_j^k)_i)^T = 0. \tag{23}$$

Let $\bar{U}_j^k = [(U_j^k)_1 \cdots (U_j^k)_r]$ and $\bar{V}_j^k = [(V_j^k)_1 \cdots (V_j^k)_r]$. Upon pre- and post-multiplying (23) by $(\bar{U}_j^k)^T$ and $\bar{V}_j^k$, and using $(\bar{U}_j^k)^T\bar{U}_j^k = (\bar{V}_j^k)^T\bar{V}_j^k = I$, we can see that for all $k \in \mathcal{K}_0$,

$$\bar{L}_j^k(\bar{U}_j^k)^T(X_j^{k+1} - X_j^k)\bar{V}_j^k + (\bar{U}_j^k)^T\nabla f_j(X_j^k)\bar{V}_j^k + \operatorname{Diag}((\omega_j)_1, \cdots, (\omega_j)_r) = 0. \tag{24}$$

Notice that $\{\bar{U}_j^k\}_{k\in\mathcal{K}}$ and $\{\bar{V}_j^k\}_{k\in\mathcal{K}}$ are bounded. Considering a convergent subsequence if necessary, assume without loss of generality that $\{\bar{U}_j^k\}_{k\in\mathcal{K}} \to \bar{U}_j^*$ and $\{\bar{V}_j^k\}_{k\in\mathcal{K}} \to \bar{V}_j^*$. Using the boundedness of $\{\bar{L}_j^k\}_{k\in\mathcal{K}}$,

$\|X_j^{k+1} - X_j^k\|_F \to 0$, $\{X_j^k\}_{k\in\mathcal{K}} \to X_j^*$, and taking limits on both sides of (24) as $k \in \mathcal{K}_0 \to \infty$, we have

$$(\bar{U}_j^*)^T \nabla f_j(X_j^*) \bar{V}_j^* + \mathrm{Diag}((\omega_j)_1, \cdots, (\omega_j)_r) = 0. \tag{25}$$

Taking the limits as $k \in \mathcal{K}_0 \to \infty$ to $(\bar{U}_j^k)^T \bar{U}_j^k = (\bar{V}_j^k)^T \bar{V}_j^k = I$, we have $(\bar{U}_j^*)^T \bar{U}_j^* = (\bar{V}_j^*)^T \bar{V}_j^* = I$. Using (18), (21), $r = \mathrm{rank}(X_j^*)$, $\{X_j^{k+1}\}_{k\in\mathcal{K}} \to X_j^*$, $\{\bar{U}_j^k\}_{k\in\mathcal{K}} \to \bar{U}_j^*$ and $\{\bar{V}_j^k\}_{k\in\mathcal{K}} \to \bar{V}_j^*$, one can get that $X_j^* = \bar{U}_j^* \,\mathrm{Diag}(\delta(X_j^*))(\bar{V}_j^*)^T$. Hence, $(\bar{U}_j^*, \bar{V}_j^*) \in \mathcal{M}(X_j^*)$. Moreover, $\{\psi(\delta(X_j^{k+1}))\}_{k\in\mathcal{K}} \to \psi(\delta(X_j^*))$ and $\omega_j \in \partial\psi(\delta(X_j^{k+1}))$ imply that $\omega_j \in \partial\psi(\delta(X_j^*))$. Using those relations and (25), we can conclude that (12) holds at X_j^* with $U_j = \bar{U}_j^*$, $V_j = \bar{V}_j^*$, $d_j^* = \omega_j$. This proves Theorem 2.6.

□

The convergence result in Theorem 2.6 still holds for a simplified version of Algorithm 1, in which L_j^k is set to be the Lipschitz constant $L_f = \frac{1}{\sigma^2}$ and step (a), (e) and (f) are removed. However, Algorithm 1 with an accelerated scheme can speed up the convergence of the algorithm, because it allows to choose L_j^k that is smaller than L_f.

2.5 Rician noise removal

The proposed method presented in Algorithm 1 is used to estimate the clean patch matrix $\hat{X}_j$ from the patch matrix Y_j by solving the MAP model with the weighted nuclear norm penalization. By aggregating all estimated patches $\hat{X}_j$, the entire image can be reconstructed. The overall algorithm for Rician noise removal is presented in Algorithm 2.

Algorithm 2: Rician Noise Removal

```
Input: Degraded image $y$ and parameter $\gamma \in (0,1)$.
Initialization: $x^{(0)} = y, y^{(0)} = y$.
for $k = 1 : K$
    Iterative regularization $y^{(k)} = x^{(k-1)} + \gamma\left(y - x^{(k-1)}\right)$.
    for each patch $y_j$ in $y^{(k)}$
        Find similar patch group $Y_j$.
        Estimate weight vector $\omega_j$ using (26).
        Estimate $\hat{X}_j$ from $Y_j$ via model (8) using Algorithm 1.
    end for
    Aggregate $\hat{X}_j$ to form the image $x^{k+1}$.
end for
Output: $\hat{x} = x^{(K+1)}$.
```

The choice of the weight vector ω_j can be estimated by

$$(\omega_j)_i = \frac{\tilde{C}}{\delta_i(\tilde{X}_j^*) + \varepsilon}, \tag{26}$$

where $\tilde{C}$ and ε are positive constants and $\tilde{X}_j^*$ is a low rank solution the approximates the given noisy patch group Y_j. In particular, we compute $\tilde{X}_j^*$ explicitly using the following lemma with $Y = Y_j$ and $X^* = \tilde{X}_j^*$.

Lemma 2.7. [13] *Let $Y = U\Sigma V^T$ be the SVD of Y, where $\Sigma = \begin{bmatrix} \mathrm{Diag}(\delta(Y)) \\ 0 \end{bmatrix}$. Suppose that the regularization parameter C is positive and the positive value ε is small enough to make the inequality $\varepsilon < \min(\sqrt{C}, \frac{C}{\delta_1(Y)})$ hold. Then the sequence $\{X^k\}$ generated by*

$$X^{k+1} = \underset{X}{\operatorname{argmin}} \frac{1}{2}\|Y - X\|_F + \|X\|_{\omega^k,*}$$

with the reweighted formula $\omega_i^k = \frac{C}{\delta_i(X^k)+\varepsilon}$ and initial estimation $X_0 = Y$, converges to the closed-form solution: $X^* = U\tilde{\Sigma}V^T$, where

$$\tilde{\Sigma} = \begin{bmatrix} \mathrm{Diag}(\delta(X^*)) \\ 0 \end{bmatrix},$$

and

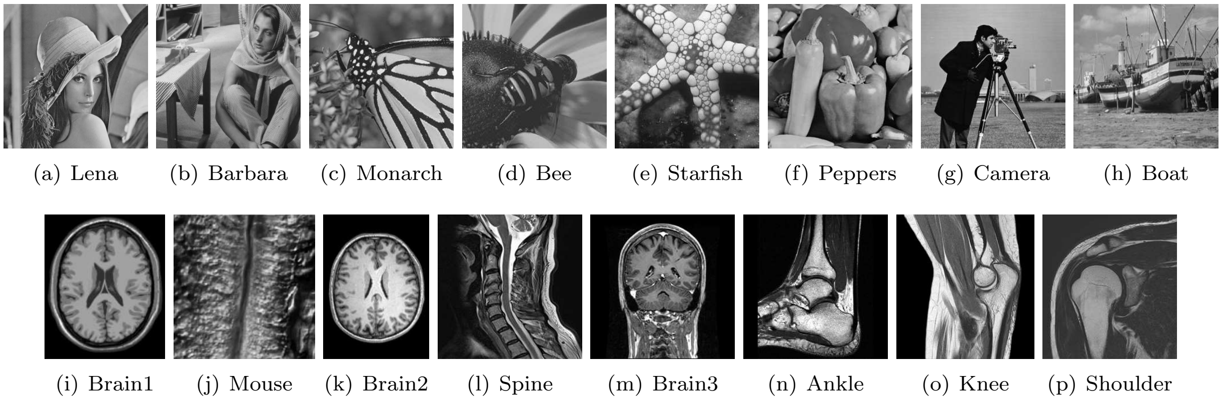
$$\delta_i(X^*) = \begin{cases} 0, & \text{if } d_i < 0; \\ \frac{c_i + \sqrt{d_i}}{2}, & \text{if } d_i \geq 0 \end{cases}$$

where

$$c_i = \delta_i(Y) - \varepsilon \quad \text{and} \quad d_i = (\delta_i(Y) + \varepsilon)^2 - 4C.$$

3 Numerical experiments

In this section, we conduct numerical experiments to demonstrate the effectiveness of our proposed algorithm for Rician noise removal. We compare our method with the GTV model [12], the CZ model [7], the NLM model [17], the C-KSVD model [8], the WNNM model [14] and the VST-BM3D method [9, 10]. All the experiments are performed under Windows 10 and MATLAB R2017a running on a PC equipped with 2.90GHz CPU and 4G RAM.


(a) Lena (b) Barbara (c) Monarch (d) Bee (e) Starfish (f) Peppers (g) Camera (h) Boat

(i) Brain1 (j) Mouse (k) Brain2 (l) Spine (m) Brain3 (n) Ankle (o) Knee (p) Shoulder

Figure 1: Original images.

We test all the image denoising methods on both standard images and medical images. Standard test images are shown in Figure 1 (a)-(h). "Lena" and "Barbara" are of size 512×512, and other images are of size 256×256. Medical test images are shown in Figure 1 (i)-(p). "Brain1" is of size 181×217, "Mouse intestine" (Briefly, we call it "Mouse" throughout this paper) is of size 248×254, "Brain2" is of size 174×238, "Spine" is of size 256×256, "Brain3" and "Ankle" are of size 250×250, "Knee" is of size 262×250 and "Shoulder" is of size 247×230. The test images are degraded by Rician noise with $\sigma = 10, 20, 30$, respectively. We evaluate the quality of recovered images obtained from various denoising algorithms by using the peal-signal-to-noise ratio (PSNR) and the structural similarity index (SSIM) [24]. The PSNR is defined by:

$$\mathrm{PSNR}(x, \hat{x}) = 10 \log_{10} \Big(\frac{255^2}{\|x - \hat{x}\|_2^2} \Big),$$

where x and $\hat{x}$ denote the original image and the recovered image respectively. And the SSIM is defined by:

$$\mathrm{SSIM}(x,\hat{x}) = \frac{(2\mu_x\mu_{\hat{x}} + c_1)(2\sigma_{x\hat{x}} + c_2)}{(\mu_x^2 + \mu_{\hat{x}}^2 + c_1)(\sigma_x^2 + \sigma_{\hat{x}}^2 + c_1)},$$

where μ_x (resp., $\mu_{\hat{x}}$) represents the mean of x (resp., $\hat{x}$), σ_x^2 (resp., $\sigma_{\hat{x}}^2$) denotes the variance of x (resp., $\hat{x}$), $\sigma_{x\hat{x}}^2$ is the covariance of x and $\hat{x}$, c_1 and c_2 are constants. The larger the values of PSNR and SSIM are, the better the recovered image is.

Firstly, we have a brief overview of the models with which we compare our method.

The GTV model is given by

$$\min_x \beta \int_\Omega G_\sigma(x,y) + \int_\Omega |Dx|dt, \tag{27}$$

where

$$G_\sigma(x,y) = \begin{cases} H_\sigma(x) & \text{if } x \geq c\sigma, \\ H_\sigma(c\sigma) + H'_\sigma(c\sigma)(x - c\sigma) & \text{if } x \leq c\sigma, \end{cases}$$

$$H'_\sigma(x) = \frac{x}{\sigma^2} - \frac{y}{\sigma^2} B\left(\frac{xy}{\sigma^2}\right),$$

$$B(s) \equiv \frac{B_1(s)}{B_0(s)} \approx \frac{s^3 + 0.950037s^2 + 2.38944s}{s^3 + 1.48937s^2 + 2.57541s + 4.65314},$$

where $c = 0.8426$, B_1 is the modified Bessel function of the first kind with order one [3]. Because it is an approximate equality, we cannot write down the final explicit restoration model.

The CZ model is given by

$$\min_{x\in S(\Omega)} \frac{1}{2\sigma^2}\int_\Omega x^2 dt - \int_\Omega \log B_0\left(\frac{xy}{\sigma^2}\right)dt + \frac{1}{\sigma}\int_\Omega (\sqrt{x} - \sqrt{y})^2 dt + \beta\int_\Omega |Dx|dt, \tag{28}$$

where $S(\Omega) := \{\nu \in BV(\Omega) : 0 \leq \nu \leq 255\}$. And since $BV(\Omega)$ [1, 2] is the subspace of function $x \in L^1(\Omega)$, the following quantity is finite:

$$J(x) = \sup\left\{\int_\Omega x(t)\,\mathrm{div}(\xi(t))dt \mid \xi \in C_0^\infty(\Omega,\mathbb{R}^2), \parallel \xi \parallel_{L^\infty(\Omega,\mathbb{R}^2)} \leq 1\right\}.$$

Before introducing the NLM model, we have a description of the sign that will be used. For a user-defined radius R_{sim}, we define a square neighborhood window centered around pixel i as N_i. And a Gaussian weighted Euclidean distance of all the pixels of each neighborhood is defined as:

$$L(m,n) = G_\rho\|y(N_m) - y(N_n)\|_{R_{sim}}^2,$$

where G_ρ is a normalized Gaussian weighting function with zero mean and ρ standard deviation (usually set to be 1). By giving more weight to pixels near the center, we can through G_ρ penalize pixels far from the center of the neighborhood window. And based on the similarity between the neighborhoods N_m and N_n of pixels m and n, we calculate the similarity $f(m,n)$ as

$$f(m,n) = \frac{1}{C(m)} e^{-\frac{L(m,n)}{h^2}},$$

where $C(m) = \sum_n e^{-\frac{L(m,n)}{h^2}}$ is the normalizing constant and h is a exponential decay control parameter.

Then given an image y, using the NLM method we can calculated the filtered value at a point m by the following formula:

$$\begin{gathered} NLM(y(m)) = \sum_{\forall n \in y} f(m,n)y(n), \\ \text{with } 0 \leq f(m,n) \leq 1, \sum_{\forall n\in y} f(m,n) = 1. \end{gathered} \tag{29}$$

Table 1: Settings of parameter β for GTV and CZ testing algorithms

Images	$\sigma = 10$		$\sigma = 20$		$\sigma = 30$	
	GTV	CZ	GTV	CZ	GTV	CZ
Lena, Monarch	15	0.06	25	0.05	35	0.04
Other standard images	25	0.06	35	0.05	45	0.04
Mouse	15	0.06	25	0.05	35	0.04
Other medical images	25	0.055	35	0.045	45	0.035

The following is the C-KSVD model:

$$\min_{x,\alpha} \ \frac{1}{2\sigma^2}\|x\|_2^2 - \log B_0\left(\frac{xy}{\sigma^2}\right) + \frac{1}{\sigma}\|\sqrt{x} - \sqrt{y}\|_2^2 + \mu\|\alpha\|_0, \quad s.t. \ x = \mathbb{D}\alpha \tag{30}$$

where μ is a parameter, $\|\cdot\|_0$ and $\|\cdot\|_2$ represent ℓ_0-norm and ℓ_2-norm respectively, and $\mathbb{D}$ is an over-complete dictionary matrix.

Secondly, we give the parameter settings for all the methods. For our proposed method, we set in Algorithm 1 that $L_{\min} = 0.1/\sigma^2$, $L_{\max} = 1/\sigma^2$, $\tau = 2$, $c = 10^{-4}$ and $N_0 = 3$; as for block matching, we set the size of the patch to be 6×6 and the number of similar patches $n = 70$ for $\sigma = 10, 20, 30$; and we set in Algorithm 2 that $\gamma = 0.3$, $C = 3.2\sqrt{n}$, $\tilde{C} = 5.6\sqrt{n}$ and $\varepsilon = 10^{-16}$. Moreover, for GTV model and CZ model, the values of parameter β are listed in Table 1.

Now, we compare the numerical results of different denoising methods on standard images and medical images. In Table 2 and 3, the first column gives the test image names, the second column gives the noise standard deviation σ, the third column gives the performance metrics, and Columns 4-11 are numerical results of different denoising methods. We observe that the PSNR values of the restored images by our proposed method are higher than those of GTV method, CZ method, NLM method and C-KSVD method, and almost higher than WNNM method and benchmark VST-BM3D method. As far as "Barbara" is concerned, the PSNR value of our method is more than 6 dB higher than CZ model at the $\sigma = 20$. The characteristics of SSIM values are almost consistent with those of PSNR values. However, the SSIM values are composed of luminance comparison, contrast comparison, and structure comparison, which makes differences at some points.

Finally, we compare the visual quality of the restored images generated by the above six methods as well as our proposed method. The experiment results tested on standard test image "Lena" and medical image "Mouse" are shown in Figures 2 and 3. In order to observe the superiority of our proposed model more clearly, we magnify some local parts of the images. We can clearly see that fewer details and more noise remain in the results of GTV, CZ, NLM and C-KSVD methods. For example, in Row 4 of Figure 2, the face in "Lena" image restored by our method is more smooth and the hair area is more refined; in Row 4 of Figure 3 for the results of "Mouse", the denoised images by GTV method, CZ method, and NLM method still remain some noise and are blocky; the denoised image by C-KSVD method is too smooth with details missing due to the overprocessing of denoising; the denoised images by WNNM method and VST-BM3D method have issues with image contrast; and the image restored by our proposed method trades off between denoising and detail preservation, which makes it better than the other six methods.

In conclusion, the numerical results demonstrate that our proposed method remarkably performs better than the GTV method, CZ method, NLM method, C-KSVD method and WNNM method, and achieves comparable performance to the VST-BM3D method.

4 Conclusion

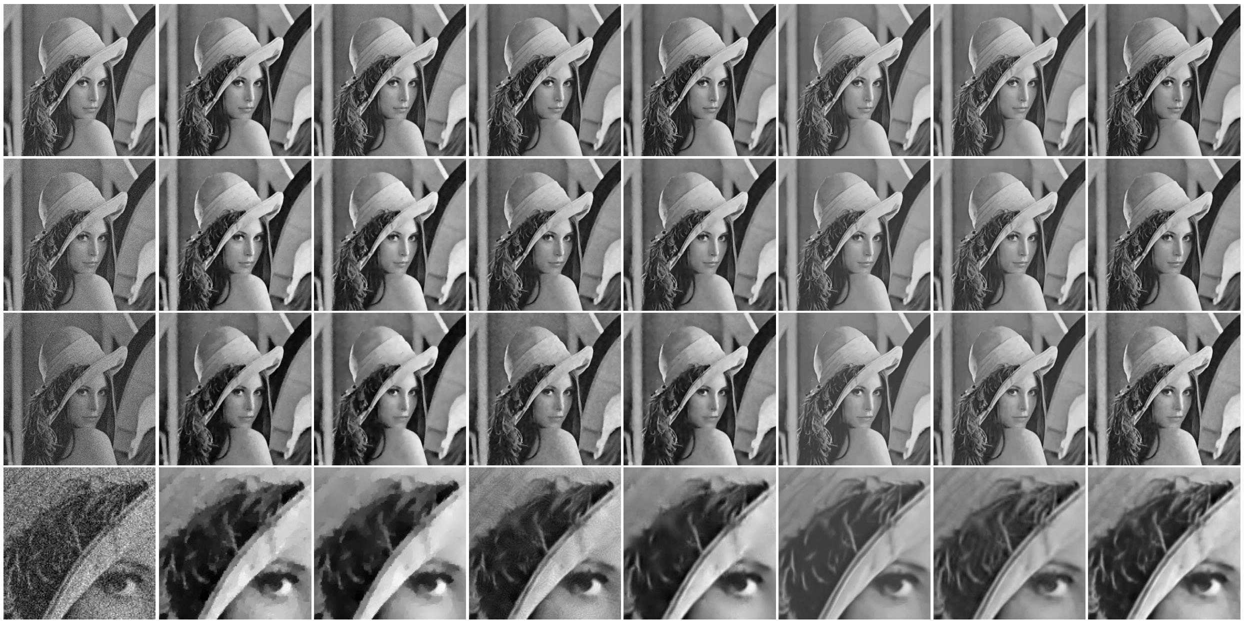
In this paper, we propose a new effective model via the weighted nuclear norm penalization for denoising images with Rician noise. More specifically, based on the low rank matrix approximation, we use the weighted nuclear norm as a regularization constraint to improve the MAP model. Using the facts that the MAP fidelity term has a Lipschitz continuous gradient and the weighted nuclear norm can be efficiently minimized, we propose the IWNNM algorithm to solve the proposed non-convex model and prove that our algorithm converges to a stationary point. Also, we carry out experiments on various images to demonstrate the effectiveness of our model. The numerical experiments show that our proposed model is promising in denoising

Table 2: Comparison of different denoising methods on standard images

Img	σ	Metric	Noisy	GTV	CZ	NLM	C-KSVD	WNNM	VST-BM3D	Proposed
Lena	10	PSNR	28.13	34.12	34.02	34.87	35.42	35.92	35.85	**35.99**
		SSIM	0.682	0.956	0.953	0.961	0.965	0.968	0.968	**0.969**
	20	PSNR	22.17	30.89	30.90	30.97	31.61	32.25	32.88	**32.96**
		SSIM	0.681	0.915	0.916	0.914	0.925	0.938	**0.940**	0.936
	30	PSNR	18.74	29.04	28.88	27.88	28.28	29.12	**31.02**	30.90
		SSIM	0.536	0.877	0.874	0.859	0.878	0.905	**0.911**	0.899
Barbara	10	PSNR	28.14	31.03	30.83	33.29	34.21	35.33	34.89	**35.36**
		SSIM	0.914	0.949	0.952	0.969	0.972	0.976	0.976	**0.977**
	20	PSNR	22.22	26.87	25.75	29.21	30.01	31.15	31.60	**31.93**
		SSIM	0.767	0.879	0.863	0.922	0.933	0.951	0.951	**0.952**
	30	PSNR	18.76	24.85	23.92	25.84	26.28	27.62	29.54	**29.55**
		SSIM	0.635	0.820	0.793	0.860	0.871	0.918	**0.923**	0.916
Monarch	10	PSNR	28.11	32.86	32.89	32.44	33.50	**34.91**	33.89	34.88
		SSIM	0.734	0.933	0.927	0.939	0.948	0.958	0.953	**0.959**
	20	PSNR	22.19	28.72	28.65	29.14	29.52	30.51	30.02	**30.99**
		SSIM	0.518	0.879	0.883	0.869	0.906	0.922	0.915	**0.923**
	30	PSNR	18.77	26.44	26.10	26.24	26.25	27.06	27.81	**28.65**
		SSIM	0.393	0.834	0.831	0.793	0.855	0.882	0.876	**0.882**
Bee	10	PSNR	26.30	33.12	32.96	32.89	33.69	34.75	34.36	**34.82**
		SSIM	0.658	0.895	0.874	0.890	0.919	0.927	0.919	**0.928**
	20	PSNR	21.84	29.64	29.19	29.62	30.86	30.62	30.90	**31.28**
		SSIM	0.411	0.804	0.755	0.808	0.844	0.856	0.846	**0.857**
	30	PSNR	18.66	27.77	27.94	27.02	28.62	27.50	29.19	**29.29**
		SSIM	0.295	0.754	0.760	0.732	0.780	0.800	**0.802**	0.800
Starfish	10	PSNR	24.99	31.83	31.64	31.80	30.18	**33.52**	33.16	33.50
		SSIM	0.778	0.912	0.902	0.906	0.919	0.931	0.927	**0.932**
	20	PSNR	19.32	27.88	27.85	28.08	29.24	29.12	29.26	**29.70**
		SSIM	0.561	0.827	0.806	0.821	0.853	0.865	0.859	**0.867**
	30	PSNR	16.16	25.81	25.90	25.26	27.01	25.88	27.23	**27.70**
		SSIM	0.429	0.767	0.765	0.728	0.789	0.803	0.807	**0.817**
Peppers	10	PSNR	25.78	33.52	33.16	33.93	34.88	35.15	35.05	**35.43**
		SSIM	0.658	0.919	0.895	0.923	0.935	0.941	0.939	**0.942**
	20	PSNR	21.86	29.47	29.10	30.00	30.86	30.16	31.24	**31.68**
		SSIM	0.443	0.835	0.798	0.850	0.881	0.891	0.894	**0.899**
	30	PSNR	18.44	27.29	27.37	27.20	28.32	26.56	29.16	**29.39**
		SSIM	0.323	0.790	0.808	0.778	0.833	0.843	0.858	**0.859**
Cameraman	10	PSNR	27.31	32.45	32.47	31.50	33.39	33.61	33.69	**34.15**
		SSIM	0.654	0.902	0.871	0.886	0.923	0.922	0.925	**0.929**
	20	PSNR	19.41	27.67	28.21	27.94	29.63	28.08	29.69	**30.22**
		SSIM	0.443	0.708	0.770	0.717	0.851	0.836	0.853	**0.856**
	30	PSNR	17.30	25.70	25.52	25.88	27.06	24.17	27.66	**28.07**
		SSIM	0.320	0.654	0.753	0.637	0.788	0.760	**0.802**	0.798
Boat	10	PSNR	28.17	32.13	31.83	31.84	33.04	**33.75**	33.44	33.72
		SSIM	0.733	0.895	0.878	0.890	0.922	**0.930**	0.926	0.928
	20	PSNR	22.17	27.93	28.05	27.98	29.43	29.22	29.41	**29.69**
		SSIM	0.490	0.796	0.783	0.780	0.834	0.846	0.843	**0.848**
	30	PSNR	18.66	25.67	26.12	25.52	26.87	26.09	27.38	**27.54**
		SSIM	0.346	0.711	0.716	0.687	0.745	0.771	0.779	**0.780**

Table 3: Comparison of different denoising methods on medical images

Img	σ	Metric	Noisy	GTV	CZ	NLM	C-KSVD	WNNM	VST-BM3D	Proposed
Brain1	10	PSNR	27.16	32.80	33.46	34.98	35.15	30.80	35.83	**37.00**
		SSIM	0.665	0.915	0.946	0.964	0.966	0.812	0.960	**0.979**
	20	PSNR	21.08	28.96	29.02	29.66	29.95	24.69	31.48	**32.65**
		SSIM	0.483	0.865	0.895	0.891	0.915	0.758	0.866	**0.940**
	30	PSNR	17.53	26.77	26.09	25.93	26.17	21.04	29.12	**29.66**
		SSIM	0.355	0.824	0.830	0.802	0.842	0.704	0.857	**0.876**
Mouse	10	PSNR	28.18	31.93	32.01	32.49	33.82	34.13	34.37	**34.72**
		SSIM	0.828	0.929	0.930	0.935	0.953	0.957	0.958	**0.961**
	20	PSNR	22.21	27.75	27.45	27.61	28.02	28.47	30.25	**30.35**
		SSIM	0.599	0.835	0.828	0.837	0.866	0.887	0.906	**0.909**
	30	PSNR	18.69	25.42	24.56	24.03	23.84	24.57	**27.64**	27.58
		SSIM	0.428	0.750	0.712	0.727	0.741	0.804	0.842	**0.845**
Brain2	10	PSNR	26.72	32.77	32.87	34.13	32.16	29.20	35.03	**35.68**
		SSIM	0.579	0.915	0.948	0.957	0.955	0.652	0.931	**0.961**
	20	PSNR	20.60	28.93	28.66	30.40	30.33	23.37	31.20	**31.91**
		SSIM	0.444	0.887	0.900	**0.921**	0.916	0.606	0.868	0.912
	30	PSNR	17.11	26.77	26.02	27.58	27.40	19.95	29.05	**29.58**
		SSIM	0.350	0.860	0.847	**0.875**	0.873	0.572	0.845	0.839
Spine	10	PSNR	26.85	31.59	30.69	31.94	26.26	32.96	33.83	**34.12**
		SSIM	0.766	0.924	0.899	0.914	0.928	0.936	0.948	**0.949**
	20	PSNR	21.50	26.40	27.47	26.93	29.12	26.31	29.70	**29.86**
		SSIM	0.567	0.649	0.819	0.677	0.868	0.827	0.883	**0.889**
	30	PSNR	18.37	23.90	24.40	24.08	25.62	21.98	26.88	**27.05**
		SSIM	0.434	0.516	0.737	0.534	0.759	0.711	0.772	**0.808**
Brain3	10	PSNR	28.30	30.55	31.08	31.38	25.29	32.61	34.68	**35.12**
		SSIM	0.695	0.917	0.785	0.837	0.941	0.929	**0.966**	0.965
	20	PSNR	21.70	27.04	26.98	27.19	29.13	25.08	29.86	**30.43**
		SSIM	0.456	0.832	0.524	0.529	0.873	0.775	0.863	**0.897**
	30	PSNR	17.81	23.62	25.17	25.13	25.51	20.77	27.32	**27.70**
		SSIM	0.331	0.709	0.461	0.454	0.756	0.655	0.742	**0.819**
Ankle	10	PSNR	27.14	29.20	30.92	29.50	25.30	28.69	31.84	**31.94**
		SSIM	0.636	0.715	0.790	0.765	0.717	0.684	0.842	**0.844**
	20	PSNR	20.87	25.10	27.27	26.92	26.42	23.06	28.17	**28.32**
		SSIM	0.446	0.566	0.660	0.642	0.608	0.554	**0.744**	0.729
	30	PSNR	17.29	22.32	25.38	25.00	23.61	19.65	26.27	**26.32**
		SSIM	0.337	0.470	0.600	0.585	0.512	0.477	**0.672**	0.648
Knee	10	PSNR	26.91	30.47	32.29	32.68	30.07	29.39	33.86	**34.17**
		SSIM	0.596	0.671	0.891	0.919	0.699	0.663	0.910	**0.934**
	20	PSNR	20.94	25.83	28.71	29.31	27.74	24.01	30.33	**30.81**
		SSIM	0.435	0.571	0.850	**0.855**	0.613	0.604	0.836	0.845
	30	PSNR	17.46	23.20	26.81	27.01	25.03	20.81	28.58	**28.74**
		SSIM	0.328	0.514	0.810	0.802	0.551	0.564	**0.819**	0.743
Shoulder	10	PSNR	28.16	33.54	32.92	34.29	35.14	35.28	35.13	**35.42**
		SSIM	0.620	0.864	0.835	0.889	0.903	0.906	0.901	**0.908**
	20	PSNR	22.30	30.63	30.15	30.47	32.05	31.46	32.44	**32.61**
		SSIM	0.366	0.828	0.778	0.819	0.852	**0.865**	0.862	0.862
	30	PSNR	18.96	28.67	28.71	27.08	30.01	27.62	**30.74**	**30.74**
		SSIM	0.248	0.792	0.760	0.740	0.809	**0.828**	0.825	0.824



(a) Noisy (b) GTV (c) CZ (d) NLM (e) C-KSVD (f) WNNM (g) VST-BM3D (h) Proposed

Figure 2: Results of “Lena” by different methods. From top to bottom, the first three rows are the noisy/denoised images corrupted/restored by Rician noise with $\sigma = 10, 20$, and 30; and the fourth row is the zoomed-in views of the third row. From left to right, the denoising methods are the noisy image, GTV method, CZ method, NLM method, C-KSVD method, WNNM method, VST-BM3D method, and our proposed method.

images with Rician noise.

Acknowledgments

The authors would like to thank one of the authors of Ref. [7] for providing the source code of CZ method, one of the authors of Ref. [8] for providing the source code of C-KSVD method, and the authors of Ref. [9, 14] for providing the source codes on their websites. Also, the authors would like to thank two anonymous reviewers for providing us valuable comments and suggestions which lead to improvements of our article. This work was supported by the National Natural Science Foundation of China under grants 61972265 and 11871348, by the Natural Science Foundation of Guangdong Province of China under grant 2020B1515310008, by the Educational Commission of Guangdong Province of China under grant 2019KZDZX1007.

References

[1] L. Ambrosio, N. Fusco and D. Pallara, Functions of bounded variation and free discontinuity problem, Oxford University Press, London, 2000.

[2] G. Aubert and J. Aujol, A variational approach to remove multiplicative noise, SIAM J. Appl. Math., 68(4) (2008) 925-946.

[3] F. Bowman, Introduction to bessel functions, Dover Publications, Mineola, (2012).

[4] A. M. Buchanan and A. W. Fitzgibbon, Damped newton algorithms for matrix factorization with missing data, IEEE Computer Society Conference on Computer Vision & Pattern Recognition, (2005).

[5] J. Cai, E. J. Candès and Z. Shen, A singular value thresholding algorithm for matrix completion, SIAM J. Optimiz., 20(4) (2010) 1956-1982.

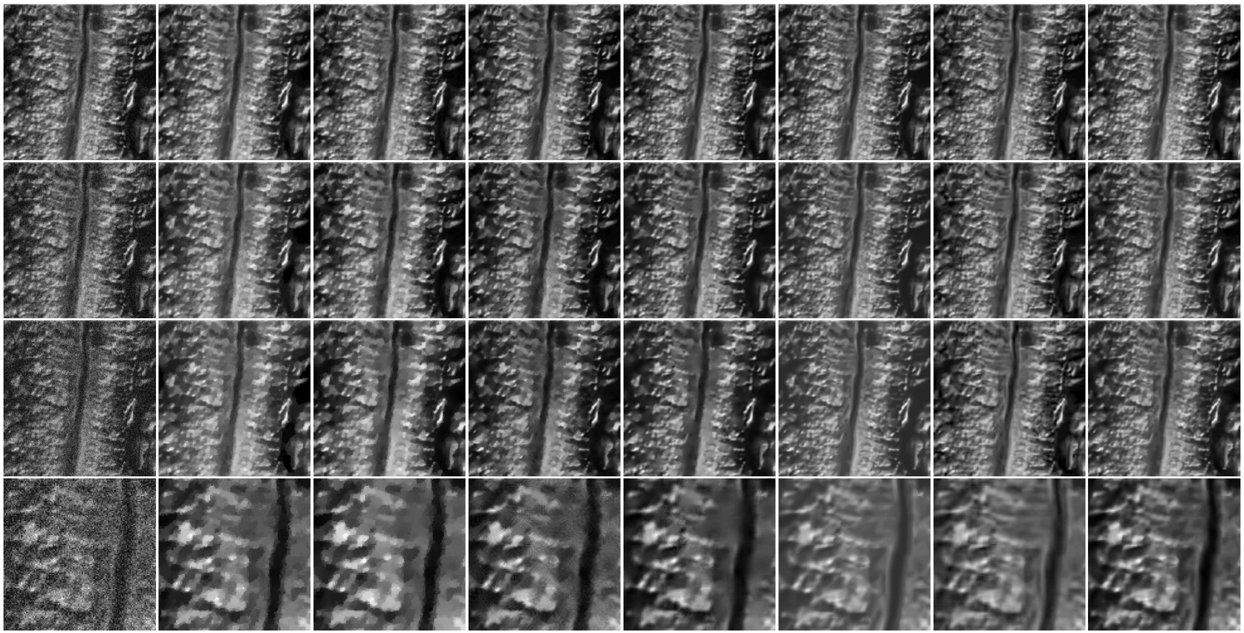


Figure 3: Results of "Mouse" by different methods. From top to bottom, the first three rows are the noisy/denoised images corrupted/restored by Rician noise with $\sigma = 10, 20$, and 30; and the fourth row is the zoomed-in views of the third row. From left to right, the denoising methods are the noisy image, GTV method, CZ method, NLM method, C-KSVD method, WNNM method, VST-BM3D method, and our proposed method.

[6] E. J. Candès and B. Recht, Exact matrix completion via convex optimization, Found. Comput. Math., 9 (6) (2009) 717-772.

[7] L. Chen and T. Zeng, A convex variational model for restoring blurred images with large Rician noise, J. Math. Imaging Vis., 53(1) (2015) 92-111.

[8] W. Chen, J. You, B. Chen, B. Pan, L. Li, M. Pomeroy and Z. Liang, A sparse representation and dictionary learning based algorithm for image restoration in the presence of Rician noise, Neurocomputing, 286 (2018) 130-140.

[9] K. Dabov, A. Foi, V. Katkovnik and K. Egiazarian, Image denoising by sparse 3-d transform-domain collaborative filtering, IEEE Transaction on Image Processing 16(8) (2007) 2080-2095.

[10] A. Foi, Noise estimation and removal in MR imaging: The variance stabilization approach, IEEE International Symposium on Biomedical Imaging: from Nano to Macro., (2011) 1809-1814.

[11] G. Gerig, O. Kubler, R. Kikinis and F. A. Jolesz, Nonlinear anisotropic filtering of MRI data, IEEE Trans. Med. Imaging, 11 (1992) 221-232.

[12] P. Getreuer, M. Tong and L. A. Vese, A variational model for the restoration of MR images corrupted by blur and Rician noise, In: Proceedings of the 7th international conference on Advances in visual computing. (2011) 686-698.

[13] S. Gu, Q. Xie, D. Meng, W. Zuo, X. F and L. Zhang, Weighted nuclear norm minimization and its applications to low level vision, Int. J. Comput. Math., 121(2) (2017) 183-208.

[14] S. Gu, L. Zhang, W. Zuo and X. F, Weighted nuclear norm minimization with application to image denoising, 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society. (2014), pp.2862-2869.

[15] A. S. Lewis and H. S. Sendov, Nonsmooth analysis of singular values. Part II: Applications, Set-Valued Analysis, 13(3) (2005) 243-264.

[16] Z. Lu, Y. Zhang and J. Lu, l_p Regularized low-rank approximation via iterative reweighted singular value minimization, Computational Optimization and Applications, 68 (2017), 619-642.

[17] J. V. Manjón, J. Carbonell-Caballero and J. J. Lull, MRI denoising using Non-Local Means, Med. Image Anal. 12 (4) (2008) 514-523.

[18] F. Nie, H. Huang and C. Ding, Low-rank matrix recovery via efficient schatten p-norm minimization, Twenty-sixth Aaai Conference on Artificial Intelligence, AAAI Press. 2012.

[19] R. Nowak, Wavelet-based Rician noise removal for magnetic resonance imaging, IEEE Trans. Image Process., 8 (1999) 1408-1419.

[20] P. Perona and J. Malik, Scale-space and edge detection using anisotropic diffusion, IEEE Trans. Pattern Anal. Machine Intell., 12 (1990) 629-639.

[21] S. Prima, S. P. Morrissey and C. Barillot, Non-local means variants for denoising of diffusion-weighted and diffusion tensor MRI, International Conference on Medical Image Computing and Computer-Assisted Intervention, (2007) 344-351.

[22] X. Ren and Z. Lin, Linearized alternating direction method with adaptive penalty and warm starts for fast solving transform invariant low-rank textures, Int. J. Comput. Vis., 104(1) (2013) 1-14.

[23] N. Srebro and T. Jaakkola, Weighted low-rank approximations, International Conference on Machine Learning, 2003.

[24] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, Image quality assessment: From error visibility to structural similarity, IEEE Trans. Image Process., 13(4) (2004) 600-612.

[25] N. Wiest-Daessle, S. Prima, P. Coupe and S. Morrissey, Rician noise removal by non-local means filtering for low signal-to-noise ratio MRI: Applications to DT-MRI, Medical image computing and computer-assisted intervention, 5242(2) (2008) 171-179.

[26] J. Wood and K. Johnson, Wavelet packet denoising of magnetic resonance images: Importance of Rician noise, Magnetic Resonance in Medicine, 41 (1999) 631-635.

[27] S. J. Wright, R. D. Nowak and M. A. T. Figueiredo, Sparse reconstruction by separable approximation, IEEE Trans Signal Proc., 57(7) (2009), 2479-2493.