

2017

Multiplicative Noise Removal with a Sparsity-Aware Optimization Model

Jian Lu

Lixin Shen

Chen Xu

Yuesheng Xu

Follow this and additional works at: https://digitalcommons.odu.edu/mathstat_fac_pubs



Part of the [Applied Mathematics Commons](#), and the [Physics Commons](#)

Repository Citation

Lu, Jian; Shen, Lixin; Xu, Chen; and Xu, Yuesheng, "Multiplicative Noise Removal with a Sparsity-Aware Optimization Model" (2017). *Mathematics & Statistics Faculty Publications*. 44.
https://digitalcommons.odu.edu/mathstat_fac_pubs/44

Original Publication Citation

Lu, J., Shen, L. X., Xu, C., & Xu, Y. S. (2017). Multiplicative noise removal with a sparsity-aware optimization model. *Inverse Problems and Imaging*, 11(6), 949-974. doi:10.3934/ipi.2017044

MULTIPLICATIVE NOISE REMOVAL WITH A SPARSITY-AWARE OPTIMIZATION MODEL

JIAN LU

College of Mathematics and Statistics
Shenzhen University, Shenzhen 518060, China

LIXIN SHEN*

Department of Mathematics
Syracuse University, Syracuse, NY 13244, USA

CHEN XU

College of Mathematics and Statistics
Shenzhen University, Shenzhen 518060, China

YUESHENG XU

School of Data and Computer Science
Guangdong Provincial Key Lab of Computational Science
Sun Yat-sen University, Guangzhou 510275, China
and

Department of Mathematics and Statistics
Old Dominion University, Norfolk, VA 23529, USA

(Communicated by Xiaoqun Zhang)

ABSTRACT. Restoration of images contaminated by multiplicative noise (also known as *speckle* noise) is a key issue in coherent image processing. Notice that images under consideration are often highly compressible in certain suitably chosen transform domains. By exploring this intrinsic feature embedded in images, this paper introduces a variational restoration model for multiplicative noise reduction that consists of a term reflecting the observed image and multiplicative noise, a quadratic term measuring the closeness of the underlying image in a transform domain to a sparse vector, and a sparse regularizer for removing multiplicative noise. Being different from popular existing models which focus on pursuing convexity, the proposed sparsity-aware model may be nonconvex depending on the conditions of the parameters of the model for achieving the optimal denoising performance. An algorithm for finding a critical point of the objective function of the model is developed based on coupled fixed-point equations expressed in terms of the proximity operator of functions that appear in the objective function. Convergence analysis of the algorithm is provided. Experimental results are shown to demonstrate that the proposed iterative algorithm is sensitive to some initializations for obtaining the best restoration results. We observe that the proposed method with SAR-BM3D filtering images as initial estimates can remarkably outperform several state-of-art methods in terms of the quality of the restored images.

2010 *Mathematics Subject Classification.* Primary: 94A08; Secondary: 49N45, 68U10.

Key words and phrases. Multiplicative noise, proximity operator, nonconvex, variational model, image restoration.

* Corresponding author: Lixin Shen.

1. Introduction. Multiplicative noise (i.e., *speckle* noise) appears naturally in coherent imaging systems, such as ultrasound imaging [31], laser imaging [27] and synthetic aperture radar (SAR) [24], due to random interference between coherent returns. Observed intensity images through coherent imaging systems usually have low signal-to-noise (SNR) ratios and high noise levels [2]. Suppressing noise in these images is therefore of vital importance in follow-up processes by human observers or computer programs. The goal of this paper is to develop efficient methods for restoration of images contaminated by multiplicative noise.

An image f degraded by multiplicative noise is modeled in the form of

$$(1) \quad f = \tilde{u} \odot \eta,$$

where $\tilde{u} \odot \eta$ denotes the componentwise multiplication of a clean image \tilde{u} and noise η . We assume that each component of the clean image \tilde{u} and the noise η is positive, and so is that of f . The multiplicative noise in each pixel is usually assumed to follow a Gamma distribution with probability density function [13]

$$p_{\eta}(v) = \frac{L^L}{\Gamma(L)} e^{-Lv},$$

where Γ is the classical Gamma function and L is the number of looks. Hence η has mean 1 and standard deviation $1/\sqrt{L}$. A high number L corresponds to a high signal-to-noise ratio of the observed image. Using the Bayesian maximum *a posteriori* (MAP) estimator for multiplicative Gamma noise, various variational models [1, 4, 10–12, 14, 16, 17, 19, 28, 32] for removing multiplicative noise were proposed recently. In [1], a model, referred to as AA model, has a nonconvex data fidelity term and a convex total variation regularization term. Through the logarithm transform, many methods form optimization convex models by using the log-image data and the exponential of the obtained minimizer is considered as the restored image, see e.g. [4, 12, 14, 16, 28] and the reference therein. The m th root transformation for relaxing the nonconvexity of AA model has considered in [17, 32]. Based on the statistics of the noise, an additional quadratic penalty term in [11] is added to the AA model to obtain a strictly convex one under certain conditions. In addition, the convex I-divergence (I-DIV) model [29], typically designed for dealing with Poisson noise, was used for speckle reduction. Most recently, by using Box-Cox transformation, the multiplicative noise removal problem was converted into the additive noise removal problem followed by applying BM3D method to get the final recovered image [15].

In our recent work [20], the following model was proposed in the log-image domain

$$(2) \quad \min_{x \in \mathbb{R}^n} \left\{ \left\langle x + \frac{f}{e^x}, \mathbf{1} \right\rangle + \alpha \left\| \sqrt{\frac{e^x}{f}} - \beta \mathbf{1} \right\|_2^2 + \lambda \|x\|_{\text{TV}} \right\},$$

where $\langle \cdot, \cdot \rangle$ and $\|\cdot\|_2$ denote the standard inner product and the usual ℓ^2 -norm over \mathbb{R}^n , respectively, α is a positive number, $\mathbf{1}$ denotes the vector having one for all components, $\beta \geq 1$ depends on the noise level, and λ is the positive regularization parameter. The term $\|x\|_{\text{TV}}$ denotes the total variation (TV) of x . The operation $\frac{f}{w}$ denotes the componentwise division of f by w for $f, w \in \mathbb{R}_+^n$. Let \mathbb{N} be the set of all natural numbers and $\mathbb{N}_n := \{1, 2, \dots, n\}$. The exponential function at $x \in \mathbb{R}^d$ is defined as $e^x := (e^{x_i} : i \in \mathbb{N}_n)$ and the square root function at $w \in \mathbb{R}_+^d$ as $\sqrt{w} := (\sqrt{w_i} : i \in \mathbb{N}_n)$. The three terms in the objective function of (2) are: a data fitting term resulting directly from the presence of the multiplicative noise,

a quadratic term reflecting the statistics of the noise, and a sparse regularizer. Numerical experiments presented in [20] show that this model compares favorably to several state-of-the-art reference models in terms of the peak signal-to-noise ratios of the denoised images and the CPU time consumed.

Certain restrictions were imposed on the parameters α and β to ensure the convexity of the optimization model (2), which certainly limit the ability of the model to yield denoised images of high quality. Furthermore, for the TV regularization, it is only optimal in describing piecewise constant images and sometime causes undesirable staircase artifacts. We can view the TV regularization in a different way. The total variation $\|x\|_{\text{TV}}$ can be rewritten as $\psi(Hx)$, where ψ is an $\ell_{2,1}$ -like sparse promoting function and H is from the first order difference matrix. Therefore, the total variation regularization prefers Hx to be sparse. Unfortunately, Hx is not sparse in general, and, as a result, the denoised image with model (2) may suffer from over-constraints on its sparsity under the first order difference matrix H . Instead, Hx is approximately sparse in the sense that it consists of a few significant entries while the rest of the entries are essentially negligible.

In this paper we propose a new model for multiplicative noise removal that can address the aforementioned issues in model (2). This model is nonsmooth, and could be nonconvex for obtaining the restoration performance. We characterize the solutions of the model via a coupled fixed-point equations derived from the proximity operators of the functions associated with the model. An algorithm based on the characterization is developed to find the solutions of the fixed-point equations. Convergence analysis of the algorithm is provided. The performance of our method is demonstrated with numerical examples.

The rest of this paper is organized as follows. In section 2, we introduce our model for multiplicative noise removal and study mathematical properties of the model. In section 3, we develop an algorithm for the model and then analyze its convergence property. In section 4, we specify the proposed model regularized by TV and present the corresponding algorithm for solving it. We review some existing models for multiplicative noise removal in section 5. We demonstrate performance of our model with numerical examples in section 6 and give a conclusion in section 7.

2. Proposed model and its characterization. This section contains main results concerning multiplicative noise removal. We present a variational model for restoring images degraded by multiplicative noise. The proposed model is motivated from addressing the issues in model (2) mentioned above. We study the existence of solutions to the proposed model and explore conditions that make the resulting model convex. The solutions of the model are characterized in terms of the fixed-points of a nonlinear map formed via the proximity operators of the functions appearing in the model. Based on the characterization, we develop an algorithm for finding solutions of the model. Convergence analysis of the proposed algorithm is provided.

2.1. Model. An image is often assumed to have a sparse representation under a basis having vanishing moments. The total variation [26] and wavelets (such as orthogonal wavelets, framelets, curvelets, bandedlets, and the discrete cosine transform [8, 9]) are appropriate bases for fulfilling this requirement. Let H denote the transform matrix associated with the filters of an wavelet system. For an image x , Hx is the vector of the wavelet coefficients of x . From the viewpoint of wavelet analysis, the vector Hx is approximately sparse, that is, this vector has few significant components while the rest of its components are essentially negligible. With

this in mind, we propose the following model to reconstruct the clean image \tilde{u} in (1):

$$(3) \quad \min \{J(x, y) : (x, y) \in \mathbb{R}^n \times \mathbb{R}^m\},$$

where for $(x, y) \in \mathbb{R}^n \times \mathbb{R}^m$, we define

$$(4) \quad J(x, y) := \Phi(x) + h(x, y) + \Psi(y),$$

and

$$(5) \quad \Phi(x) := \left\langle x + \frac{f}{e^x}, \mathbf{1} \right\rangle + \alpha \left\| \sqrt{\frac{e^x}{f}} - \beta \mathbf{1} \right\|_2^2,$$

$$(6) \quad h(x, y) := \frac{\mu}{2} \|Hx - y\|_2^2,$$

$$(7) \quad \Psi(y) := \lambda \psi(y),$$

where α , β , μ , and λ are all positive. In (6), we assume that the matrix H of size $m \times n$ is able to sparsify the image x while ψ is a sparsity promoting function. We assume that ψ is a proper lower semicontinuous convex function on \mathbb{R}^m and is able to promote sparsity. Examples of such functions include the ℓ_p norm with $0 \leq p \leq 1$ and the $\ell_{2,1}$ norm among many others. Suppose that a pair (x^*, y^*) is a solution of problem (3). Then e^{x^*} is viewed as an approximation to the clean image \tilde{u} in (1).

Let us explain the proposed model (3). There are two variables x and y in the model. The first term $\Phi(x)$ from the first two terms in model (2) reflects the presence of the multiplicative noise and the statistics of the noise. The third term $\Psi(y)$ enforces the sparsity of y . The image x is assumed to be almost sparse in its information content under the transform H . We leverage this approximate sparsity by measuring the closeness of Hx to y with their Euclidean distance given in the second term.

2.2. Properties. In this subsection, we discuss the convexity properties of the model (3) which will be utilized in the next subsection to characterize the solutions of the model. We first show the condition ensuring that the function $J(x, y)$, $(x, y) \in \mathbb{R}^n \times \mathbb{R}^m$, is convex with respect to (x, y) , which leads to the uniqueness of the solution of model (3). Then, we will show that, when taken under some relaxing conditions, though the function J may be nonconvex with respect to (x, y) , it is definitely convex with each of x and y , separately.

Proposition 1. *If $\alpha, \beta, \mu, \lambda$ are all positive numbers, ψ is a proper convex function on \mathbb{R}^m , and H is an $m \times n$ matrix, then the solution set of model (3) is nonempty. Furthermore, if the positive numbers α, β satisfy inequality $\alpha\beta^4 \leq \frac{4096}{27}$, then the solution of model (3) is unique.*

Proof. Since the vectors \tilde{u} and η in (1) are positive, so is f . Hence, Φ is a continuous function on \mathbb{R}^n and is coercive, that is, $\lim_{\|x\|_2 \rightarrow +\infty} \Phi(x) = +\infty$ which can be verified directly. Together with the coercivity assumption of Ψ , the function J in (3) is coercive on $\mathbb{R}^n \times \mathbb{R}^m$, that is $\lim_{\|(x,y)\|_2 \rightarrow +\infty} J(x, y) = +\infty$. It implies that there exists at least one solution to the optimization problem (3).

Next, we prove that problem (3) has only one solution under the condition $\alpha\beta^4 \leq \frac{4096}{27}$ and convexity assumption of Ψ . To this end, it will be sufficient to show that $\Phi + h$ is strictly convex on $\mathbb{R}^n \times \mathbb{R}^m$. By Theorem 2.4 in [20], Φ in (5) is strictly convex on \mathbb{R}^n if $\alpha\beta^4 \leq \frac{4096}{27}$. That is,

$$(8) \quad \lambda\Phi(x) + (1 - \lambda)\Phi(\tilde{x}) > \Phi(\lambda x + (1 - \lambda)\tilde{x})$$

hold for all distinct points x and \tilde{x} in \mathbb{R}^n , and $\lambda \in (0, 1)$. We further notice that h is convex on $\mathbb{R}^n \times \mathbb{R}^m$ and $h(x, \cdot)$ is strictly convex on \mathbb{R}^m for any fixed $x \in \mathbb{R}^n$. That is, for any (x, y) and (\tilde{x}, \tilde{y}) in $\mathbb{R}^n \times \mathbb{R}^m$, and $\lambda \in [0, 1]$

$$(9) \quad \lambda h(x, y) + (1 - \lambda)h(\tilde{x}, \tilde{y}) \geq h(\lambda x + (1 - \lambda)\tilde{x}, \lambda y + (1 - \lambda)\tilde{y})$$

and for any fixed $x \in \mathbb{R}^n$, any two distinct points y and \tilde{y} in \mathbb{R}^m , and $\lambda \in (0, 1)$

$$(10) \quad \lambda h(x, y) + (1 - \lambda)h(x, \tilde{y}) > h(x, \lambda y + (1 - \lambda)\tilde{y}).$$

By (8)-(10), for all distinct pairs (x, y) and (\tilde{x}, \tilde{y}) in $\mathbb{R}^n \times \mathbb{R}^m$, and $\lambda \in (0, 1)$,

$$(11) \quad \begin{aligned} & \lambda(\Phi(x) + h(x, y)) + (1 - \lambda)(\Phi(\tilde{x}) + h(\tilde{x}, \tilde{y})) \\ & > \Phi(\lambda x + (1 - \lambda)\tilde{x}) + h(\lambda x + (1 - \lambda)\tilde{x}, \lambda y + (1 - \lambda)\tilde{y}). \end{aligned}$$

which, by the definition of strict convexity, implies the strict convexity of $\Phi + h$. As a consequence, J is strictly convex on $\mathbb{R}^n \times \mathbb{R}^m$. \square

Next, we explore other choices of α and β to ensure the convexity of the function $\Phi + \frac{\rho}{2} \|\cdot\|_2^2$ with some $\rho > 0$. To this end, we adopt assumptions on positive numbers α, β , and ρ :

H1. The parameters α, β , and ρ are positive and satisfy one of the following three conditions:

- (i) $\alpha\beta^4 \leq \frac{4096}{27}$;
- (ii) $\rho \geq \frac{\alpha\beta^2}{16}$; or
- (iii) $g(t_+^2) \geq 0$ if $\rho < \frac{\alpha\beta^2}{16}$, where

$$g(t) := \frac{1}{t} + \alpha \left(t - \frac{\beta\sqrt{t}}{2} \right) + \rho$$

and

$$t_+ := \frac{1}{16\alpha} (3\alpha\beta + \sqrt{9\alpha^2\beta^2 - 128\alpha\rho}).$$

Associated with the parameters α, β , and ρ in **H1**, for a fixed positive number a , we define

$$(12) \quad g_a(t) := t + \frac{a}{e^t} + \alpha \left(\sqrt{\frac{e^t}{a}} - \beta \right)^2 + \frac{\rho}{2} t^2, \quad \text{for } t \in \mathbb{R},$$

which can be viewed as the one-dimensional counterpart of $\Phi + \frac{\rho}{2} \|\cdot\|_2^2$. With the function g_a , the role of the conditions in **H1** becomes clear in the following result.

Lemma 2.1. *Let **H1** hold. Then the function g_a given in (12) is strictly convex on \mathbb{R} .*

Proof. Note that $g_a''(t) = \frac{a}{e^t} + \alpha(\frac{e^t}{a} - \frac{\beta}{2}\sqrt{\frac{e^t}{a}}) + \rho$ is not constant on any open interval of \mathbb{R} . Then $g_a''(t) \geq 0$ for all $t \in \mathbb{R}$ ensures the strict convexity of g_a by the definition of strict convexity. Actually, for any two distinct numbers t_1 and t_2 with $t_1 < t_2$ and for any $\tau \in (0, 1)$, by the mean-value theorem, there exist $\xi_1 \in (t_1, \tau t_1 + (1 - \tau)t_2)$ and $\xi_2 \in (\tau t_1 + (1 - \tau)t_2, t_2)$ such that

$$g_a(t_1) - g_a(\tau t_1 + (1 - \tau)t_2) = -(1 - \tau)(t_2 - t_1)g_a'(\xi_1)$$

and

$$g_a(t_2) - g_a(\tau t_1 + (1 - \tau)t_2) = \tau(t_2 - t_1)g_a'(\xi_2).$$

Hence,

$$(13) \quad \tau g_a(t_1) + (1-\tau)g_a(t_2) - g_a(\tau t_1 + (1-\tau)t_2) = \tau(1-\tau)(t_2 - t_1) \int_{\xi_1}^{\xi_2} g_a''(t) dt > 0.$$

The inequality in (13) holds due to g_a'' being nonnegative and not constant on any open interval of \mathbb{R} . We conclude that g_a is strictly convex. In the rest of the proof, we just verify $g_a'' \geq 0$ on \mathbb{R} under the conditions in **H1**.

If the first condition of **H1** holds, then $g_a''(t) \geq \rho$, for all $t \in \mathbb{R}$, is a direct consequence of Lemma 2.3 in [20] (or Theorem 1 in [18]).

If the second condition of **H1** holds, then $g_a''(t) = \frac{a}{e^t} + \alpha(\sqrt{\frac{e^t}{a} - \frac{\beta}{4}})^2 + (\rho - \frac{\alpha\beta^2}{16}) > 0$ for all $t \in \mathbb{R}$.

Finally, we show $g_a''(t) \geq 0$ under the third condition of **H1**. Note that $g_a''(t) = \frac{a}{e^t} q(\frac{e^t}{a})$, where $q(s) = 1 + \alpha s^2 - \frac{\alpha\beta}{2} s^{3/2} + \rho s$ with $s > 0$. Hence, $g_a''(t) \geq 0$ for all $t \in \mathbb{R}$ if and only if $q(s) \geq 0$ for all $s \geq 0$. If $\rho < \frac{\alpha\beta^2}{16}$, then the derivative of q at s is $q'(s) = 2\alpha s - \frac{3\alpha\beta}{4} s^{1/2} + \rho = 2(\sqrt{s} - t_-)(\sqrt{s} - t_+)$, where $t_{\pm} := \frac{1}{16\alpha}(3\alpha\beta \pm \sqrt{9\alpha^2\beta^2 - 128\alpha\rho})$. We know that q achieves its local maximum at t_-^2 with $q(t_-^2) > q(0) = 1$ and has its local minimum at t_+^2 with $q(t_+^2) \geq 0$ by the given condition. We conclude that $q(s) \geq 0$ for all $s \geq 0$. \square

An extension of Lemma 2.1 to a higher dimension is as follows.

Lemma 2.2. *Let **H1** hold. Then the function $\tilde{\Phi} := \Phi + \frac{\rho}{2} \|\cdot\|_2^2$ is strictly convex, where Φ is given in (5).*

Proof. For $x \in \mathbb{R}^n$, we clearly have $\tilde{\Phi}(x) = \sum_{i=1}^n g_{f_i}(x_i)$, where g_a is given in (12). By Lemma 2.1, g_{f_i} are strictly convex for all $1 \leq i \leq n$, and so does $\tilde{\Phi}$. \square

Next result is a direct consequence of Lemma 2.2. For the matrix H in (6), we define

$$(14) \quad a := \min\{\|H_j\|_2^2 : j = 1, 2, \dots, n\},$$

where, H_j is the j th column of the matrix H .

Lemma 2.3. *If a in (14) and μ are positive, H is an $m \times n$ matrix, and the parameters α, β , and ρ with $\rho := a\mu$ fulfill **H1**, then for a given $y \in \mathbb{R}^m$, $J_y(x) := \Phi(x) + h(x, y)$, the sum of the first two terms in the objective function of (3), is strictly convex with respect to each component of $x \in \mathbb{R}^n$.*

Proof. Set $D := \text{diag}(\|H_1\|_2^2, \|H_2\|_2^2, \dots, \|H_n\|_2^2)$. We have $\|Hx - y\|_2^2 = \frac{1}{2}x^\top Dx + r(x, y)$, where

$$r(x, y) := \frac{1}{2}x^\top (H^\top H - D)x - \langle x, H^\top v \rangle + \frac{1}{2}\|y\|_2^2.$$

Then

$$J_y(x) = \sum_{j=1}^n g_{\mu\|H_j\|_2^2}(x_j) + r(x, y).$$

Note that all diagonal entries of $H^\top H - D$ are zeros, in other words, $r(x, y)$ does not contain terms with x_j^2 for all possible j . Therefore, the strict convexity of $J_y(x)$ with respect to each component of x , say x_j , is completely determined by that of $g_{\mu\|H_j\|_2^2}$. Since $g_{\mu\|H_j\|_2^2}(x_j) = g_{a\mu}(x_j) + \frac{\mu(\|H_j\|_2^2 - a)}{2}x_j^2$ and $g_{a\mu}$ is strictly convex by

Lemma 2.2, indeed, $J_y(x)$ is strictly convex with respect to each element of x for a fixed y . \square

Lemma 2.4. *If $\alpha, \beta, \lambda, \mu$ are positive numbers, H is an $m \times n$ matrix, and ψ in (7) is convex, then for any fixed $x \in \mathbb{R}^n$, $J(x, \cdot)$ in model (3) is strictly convex.*

Proof. Recall that $J(x, y) = \Phi(x) + h(x, y) + \Psi(y)$ for $(x, y) \in \mathbb{R}^n \times \mathbb{R}^m$. It is clear that for any fixed $x \in \mathbb{R}^n$, $h(x, \cdot)$ is strictly convex. Together with the convexity of Ψ , we conclude that $J(x, \cdot)$ is strictly convex with respect to y . \square

In the next subsection, we can apply those results developed in this subsection to characterize the solutions of the model (3).

2.3. Characterizations. From Proposition 1, we know that the optimization problem (3) always has a solution. In this subsection, we will characterize this solution through the notions of the proximity operator and subdifferentials which are two fundamental tools in nonlinear optimization.

Recall that for $\varphi : \mathbb{R}^d \rightarrow (-\infty, +\infty]$ a proper and lower semicontinuous function, the domain of φ is defined by $\text{dom}\varphi := \{z \in \mathbb{R}^d : \varphi(z) < +\infty\}$. For a given $z \in \text{dom}\varphi$, the Fréchet subdifferential of φ at $z \in \mathbb{R}^d$, denoted by $\partial\varphi(z)$, is the set of all vectors $w \in \mathbb{R}^d$ which satisfy

$$\partial\varphi(z) := \left\{ w \in \mathbb{R}^d : \liminf_{v \neq z, v \rightarrow z} \frac{\varphi(v) - \varphi(z) - \langle w, v - z \rangle}{\|v - z\|_2} \geq 0 \right\}.$$

When $w \notin \text{dom}\varphi$, we set $\partial\varphi(z) = \emptyset$. If $0 \in \partial\varphi(z)$, then z is called a critical point of φ . Fermat’s rule is that if $z \in \mathbb{R}^d$ is a local minimizer of φ then $0 \in \partial\varphi(z)$.

If $\varphi_1 : \mathbb{R}^d \rightarrow (-\infty, +\infty]$ and $\varphi_2 : \mathbb{R}^d \rightarrow (-\infty, +\infty]$ are subdifferentiable at z , then $\varphi_1 + \varphi_2$ is subdifferentiable at z and $\partial(\varphi_1 + \varphi_2)(z) \supset \partial\varphi_1(z) + \partial\varphi_2(z)$. In particular, if φ_1 is differentiable at z , then $\partial(\varphi_1 + \varphi_2)(z) = \nabla\varphi_1(z) + \partial\varphi_2(z)$. As an application to our problem (3), we have that for $(x, y) \in \text{dom}J$

$$(15) \quad \partial J(x, y) = (\partial\Phi(x) + \nabla_x h(x, y), \nabla_y h(x, y) + \partial\Psi(y)),$$

where ∇_x and ∇_y are gradient operators with respect to x and y , respectively.

The proximity operator is another fundamental tool for developing algorithms for nonsmooth and nonconvex optimization problems. Let $\varphi : \mathbb{R}^d \rightarrow (-\infty, +\infty]$ be a proper and lower semicontinuous function. The proximity operator of φ at $z \in \mathbb{R}^d$ is defined by (see. e.g., [3, 22])

$$\text{prox}_\varphi(z) := \text{argmin} \left\{ \frac{1}{2} \|v - z\|_2^2 + \varphi(v) : v \in \mathbb{R}^d \right\}.$$

The proximity operator of φ is a set-valued operator from \mathbb{R}^d to $2^{\mathbb{R}^d}$. If $\inf_{\mathbb{R}^d} \varphi > -\infty$, then the set $\text{prox}_\varphi(z)$ is nonempty and compact. In addition, we say that φ is *supercoercive* if $\varphi(v)/\|v\|_2 \rightarrow +\infty$ as $\|v\|_2 \rightarrow +\infty$.

The following result characterizes the relationship between the proximity operator and subdifferential of a proper lower semicontinuous (nonconvex) function. This proposition will serve as a basic tool for the algorithmic development of model (3).

Proposition 2. *Let $\varphi : \mathbb{R}^d \rightarrow (-\infty, +\infty]$ be a proper lower semicontinuous function. If there exist vectors $z, w \in \mathbb{R}^d$ satisfying*

$$(16) \quad z \in \text{prox}_\varphi(z + w),$$

then

$$(17) \quad w \in \partial\varphi(z).$$

Conversely, if vectors $z, w \in \mathbb{R}^d$ satisfy (17) and the function $\tilde{\varphi} := \frac{1}{2}\|\cdot\|_2^2 + \varphi$ is convex and supercoercive, then the inclusion (16) holds. If $\tilde{\varphi}$ is further assumed to be strictly convex, the equation $z = \text{prox}_\varphi(z + w)$ holds.

Proof. If the vectors $z, w \in \mathbb{R}^d$ satisfy (16), it follows from the definition of the proximity operator that

$$(18) \quad z \in \operatorname{argmin} \left\{ \frac{1}{2}\|v - (z + w)\|_2^2 + \varphi(v) : v \in \mathbb{R}^d \right\}.$$

Fermat’s rule implies that

$$(19) \quad 0 \in (z - (z + w)) + \partial\varphi(z),$$

which is (17).

Next we show the converse of the proposition. Since $\tilde{\varphi}$ is supercoercive, then

$$\begin{aligned} \left(\frac{1}{2}\|v - (z + w)\|_2^2 + \varphi(v)\right)/\|v\|_2 &= (\tilde{\varphi}(v) - \langle v, z + w \rangle + \frac{1}{2}\|z + w\|_2^2)/\|v\|_2 \\ &\geq \tilde{\varphi}(v)/\|v\|_2 - \|z + w\|_2 \rightarrow +\infty \end{aligned}$$

as $\|v\|_2 \rightarrow +\infty$. We conclude that for given $z, w \in \mathbb{R}^d$ the function $\frac{1}{2}\|\cdot - (z + w)\|_2^2 + \varphi$ is coercive and proper convex, and therefore by (19) it has z as one of its minimizers. Hence, (16) holds. The strict convexity of $\tilde{\varphi}$ clearly implies the strict convexity of $\frac{1}{2}\|\cdot - (z + w)\|_2^2 + \varphi$. It follows that $\frac{1}{2}\|\cdot - (z + w)\|_2^2 + \varphi$ has a unique minimizer, that is, inclusion (18) becomes $z = \text{prox}_\varphi(z + w)$. \square

The following two examples concretely indicate that the convexity assumption on $\tilde{\varphi}$ is necessary to the “converse” part of Proposition 2. For instance, in Example 2.5, even if φ is a concave function, the “converse” part of Proposition 2 holds provided that $\tilde{\varphi}$ is convex. Example 2.6 shows that (17) can not lead to (16) if $\tilde{\varphi}$ is nonconvex.

Example 2.5. Given a real number c , define $\varphi(z) := cz^2$ for $z \in \mathbb{R}$. We get $0 \in \partial\varphi(0)$ and

$$\text{prox}_\varphi(0) = \begin{cases} \emptyset, & \text{if } c < -\frac{1}{2}; \\ \mathbb{R}, & \text{if } c = -\frac{1}{2}; \\ \{0\}, & \text{otherwise.} \end{cases}$$

We can see that if $c > -\frac{1}{2}$ then $\tilde{\varphi}(z) := (c + \frac{1}{2})z^2$ is supercoercive, strictly convex. Hence, by identifying both z and w in Proposition 2 as zero, the above result is consistent with the converse part of Proposition 2.

Example 2.6. Given a positive number c , define $\varphi(z) := -c|z|$ for $z \in \mathbb{R}$. We have $\partial\varphi(0) = [-c, c] \ni 0$ and $\text{prox}_\varphi(0) = \{-c, c\}$. The function $\tilde{\varphi}(z) := -c|z| + \frac{1}{2}z^2$ is supercoercive, but, nonconvex. We can see that the conditions of the converse part in Proposition 2 are not fully satisfied and $0 \notin \text{prox}_\varphi(0)$.

With above preparations, we now characterize the solutions of model (3) in terms of the proximity operators.

Proposition 3. Let α, β, ρ and λ be positive numbers, H be an $m \times n$ matrix, and ψ be a proper convex function. If α, β , and ρ satisfy **H1** and the pair $(x, y) \in \mathbb{R}^n \times \mathbb{R}^m$

is a critical point of the optimization problem (3), then for any $\sigma > 0$ the pair (x, y) satisfies the following coupled equations

$$(20) \quad x = \text{prox}_{\frac{1}{\rho}\Phi}(x - \frac{\mu}{\rho}H^\top(Hx - y)),$$

$$(21) \quad y = \text{prox}_{\frac{1}{\sigma}\Psi}(y + \frac{\mu}{\sigma}(Hx - y)).$$

Conversely, if a pair $(x, y) \in \mathbb{R}^n \times \mathbb{R}^m$ satisfies (21) and

$$(22) \quad x \in \text{prox}_{\frac{1}{\rho}\Phi}(x - \frac{\mu}{\rho}H^\top(Hx - y)),$$

then the pair (x, y) is a critical point of J in (3). Moreover, if α given in (14) is positive and the parameters α, β , and ρ with $\rho := \alpha\mu$ satisfy **H1**, then (x, y) is a (local) minimizer or a saddle (but not a (local) maximizer) of J . In particular, if the first condition in **H1** holds, then (x, y) is definitely a global minimizer of (3).

Proof. If $(x, y) \in \mathbb{R}^n \times \mathbb{R}^m$ is a critical point of the optimization problem (3), then $0 \in \partial J(x, y)$ which, by using (15), can be equivalently written as

$$(23) \quad \begin{cases} 0 \in \partial \left(\frac{1}{\rho}\Phi \right) (x) + \frac{\mu}{\rho}H^\top(Hx - y), \\ 0 \in \partial \left(\frac{1}{\sigma}\Psi \right) (y) - \frac{\mu}{\sigma}(Hx - y). \end{cases}$$

By Lemma 2.2, $\Phi + \frac{\rho}{2}\|\cdot\|_2^2$ is strictly convex which, together with the first inclusion in (23) and Proposition 2, yields equation (20). Since Ψ is convex, for any $\sigma > 0$, $\Psi + \frac{\sigma}{2}\|\cdot\|_2^2$ is coercive and strictly convex. It follows from Proposition 2 that the second inclusion in (23) lead to equation (21).

Next, we show the converse of the proposition. From Proposition (2), relations (22) and (21) lead to (23), which is equivalent to the relation $0 \in \partial J(x, y)$, i.e., (x, y) is a critical point of J . Further, by Lemmas 2.3 and 2.4, the set of critical points of J will not contain any (local) maximizer. Therefore, the pair (x, y) must not be a (local) maximizer of (3). In particular, if α and β satisfy $\alpha\beta^4 \leq \frac{4096}{27}$, from Proposition 1, the pair (x, y) must be the unique minimizer of (3). \square

We next show that a global minimizer of model (3) is a solution of the fixed-point equations (20) and (21). To this end, we recall the so-called descent lemma which is essential in our analysis.

Lemma 2.7 (Descent Lemma). *If $\varphi : \mathbb{R}^d \rightarrow \mathbb{R}$ is a continuously differentiable function whose gradient is Lipschitz with constant L , then for any $u, v \in \mathbb{R}^d$,*

$$\varphi(u) \leq \varphi(v) + \langle \nabla\varphi(v), u - v \rangle + \frac{L}{2}\|u - v\|_2^2.$$

This proof of the descent lemma can be found, see for example, in [23].

Proposition 4. *Let $\alpha, \beta, \rho, \lambda, \sigma$, and μ be positive numbers, H be an $m \times n$ matrix, and ψ be a proper convex function. If the pair $(x, y) \in \mathbb{R}^n \times \mathbb{R}^m$ is a solution of model (3) and ρ, σ, μ satisfy the inequality $\min\{\rho, \sigma\} \geq \mu(\|H\|_2^2 + 1)$, then (x, y) is a solution of the coupled fixed-point equations (20) and (21).*

Proof. Let $L := \mu(\|H\|_2^2 + 1)$. First, one can directly verify that the gradient of h in (6) is L -Lipschitz continuous. By the Descent Lemma and the condition $\min\{\rho, \sigma\} \geq \mu(\|H\|_2^2 + 1)$, for any $(\tilde{x}, \tilde{y}) \in \mathbb{R}^n \times \mathbb{R}^m$,

$$\begin{aligned} h(\tilde{x}, \tilde{y}) &\leq h(x, y) + \mu\langle \tilde{x} - x, H^\top(Hx - y) \rangle - \mu\langle \tilde{y} - y, (Hx - y) \rangle \\ &\quad + \frac{\rho}{2}\|\tilde{x} - x\|_2^2 + \frac{\sigma}{2}\|\tilde{y} - y\|_2^2. \end{aligned}$$

On the other hand, since the pair $(x, y) \in \mathbb{R}^n \times \mathbb{R}^m$ is a solution of model (3), we have that

$$\Phi(x) + h(x, y) + \Psi(y) \leq \Phi(\tilde{x}) + h(\tilde{x}, \tilde{y}) + \Psi(\tilde{y}).$$

Combining the above inequalities together yields

$$\begin{aligned} \Phi(x) + \Psi(y) &\leq \Phi(\tilde{x}) + \mu \langle \tilde{x} - x, H^\top(Hx - y) \rangle + \frac{\rho}{2} \|\tilde{x} - x\|_2^2 \\ &\quad + \Psi(\tilde{y}) - \mu \langle \tilde{y} - y, (Hx - y) \rangle + \frac{\sigma}{2} \|\tilde{y} - y\|_2^2. \end{aligned}$$

Since the above inequality holds for an arbitrary pair $(\tilde{x}, \tilde{y}) \in \mathbb{R}^n \times \mathbb{R}^m$, we conclude that

$$\Phi(x) \leq \Phi(\tilde{x}) + \mu \langle \tilde{x} - x, H^\top(Hx - y) \rangle + \frac{\rho}{2} \|\tilde{x} - x\|_2^2$$

and

$$\Psi(y) \leq \Psi(\tilde{y}) - \mu \langle \tilde{y} - y, (Hx - y) \rangle + \frac{\sigma}{2} \|\tilde{y} - y\|_2^2,$$

which imply (20) and (21), respectively. □

3. Convergence analysis. The coupled fixed-point equations (20) and (21) presented in Proposition 3 naturally leads to an iterative scheme to find a critical point of the objective function of model (3). This scheme is as follows: Set an initial estimate $(x^0, y^0) \in \mathbb{R}^n \times \mathbb{R}^m$, iterate for $k = 0, 1, \dots$,

$$(24) \quad \begin{cases} x^{(k+1)} \in \text{prox}_{\frac{1}{\rho}\Phi}(x^{(k)} - \frac{\mu}{\rho} H^\top(Hx^{(k)} - y^{(k)})), \\ y^{(k+1)} = \text{prox}_{\frac{1}{\sigma}\Psi}(x^{(k)} + \frac{\mu}{\sigma}(Hx^{(k+1)} - y^{(k)})). \end{cases}$$

We rewrite iterative scheme (24) in a compact form. For the positive parameters ρ and σ , proper lower semi-continuous function Φ and convex function Ψ , we define a set-valued mapping $\mathcal{T} : \mathbb{R}^{n+m} \rightarrow 2^{\mathbb{R}^{n+m}}$ at $u := (x, y) \in \mathbb{R}^n \times \mathbb{R}^m$ by

$$(25) \quad \mathcal{T}(u) := \text{prox}_{\frac{1}{\rho}\Phi}(x) \times \text{prox}_{\frac{1}{\sigma}\Psi}(y).$$

And we define

$$R_1 := \begin{bmatrix} 0 & 0 \\ \frac{\mu}{\sigma}H & 0 \end{bmatrix}, \quad R_2 := \begin{bmatrix} I_m - \frac{\mu}{\rho}H^\top H & \frac{\mu}{\rho}H^\top \\ I_m & -\frac{\mu}{\sigma}I_n \end{bmatrix},$$

and

$$R := R_1 + R_2.$$

With the above notation, the iterative scheme (24) can be rewritten in a compact equivalent form as

$$(26) \quad u^{(k+1)} \in \mathcal{T}(R_1 u^{(k+1)} + R_2 u^{(k)}).$$

We next analyze convergence of the iterative scheme (24).

Lemma 3.1. *The set-valued mapping \mathcal{T} defined by (25) is continuous.*

Proof. Note that the mapping \mathcal{T} consists of the two proximity operators. The conclusion can be directly followed by the fact that the proximity operator of a proper lower semi-continuous function is continuous [7]. □

The generated sequence is denoted by

$$\mathcal{W} := \{u^{(k)} : k \in \mathbb{N}\}, \text{ where } \mathbb{N} := \{1, 2, \dots\}$$

To prove the convergence of the sequence \mathcal{W} , the following lemmas are useful.

Lemma 3.2. *Let $\alpha, \beta, \rho, \lambda, \sigma$, and μ be positive numbers, H be an $m \times n$ matrix, ψ be a proper convex function, and \mathcal{W} be the sequence generated by the iterative scheme (24). If the following conditions hold:*

- (i) *the sequence \mathcal{W} is bounded, and*
- (ii) $\lim_{k \rightarrow \infty} \|u^{(k+1)} - u^{(k)}\|_2 = 0$,

then there exists a subsequence of \mathcal{W} that converges to a solution of the coupled relations (20) and (21).

Proof. Since \mathcal{W} is a bounded, there exists a subsequence $\{u^{(k_i)} : i \in \mathbb{N}\}$ such that $u^{(k_i)} \rightarrow \bar{u} := (\bar{x}, \bar{y})$ as $i \rightarrow \infty$. By item (ii), both the subsequences $\{u^{(k_i+1)} : i \in \mathbb{N}\}$ converges to \bar{u} as $i \rightarrow \infty$. Therefore, by choosing $k := k_i$ and letting $i \rightarrow \infty$ in (24), we conclude that \bar{u} (i.e., (\bar{x}, \bar{y})) satisfies the coupled equations (20) and (21). \square

Lemma 3.3. *Let $\alpha, \beta, \rho, \lambda, \sigma$, and μ be positive numbers, H be an $m \times n$ matrix, ψ be a proper convex function, and \mathcal{W} be the sequence generated by the iterative scheme (24). If $\mu < \sigma$ and $\mu \|H\|_2^2 < \rho$, then*

- (i) $J(u^{(k+1)}) \leq J(u^{(k)})$ for all $k \in \mathbb{N}$ and the sequence $\{J(x^{(k)}) : k \in \mathbb{N}\}$ converges.
- (ii) *the sequence \mathcal{W} is bounded and $\lim_{k \rightarrow \infty} \|u^{(k+1)} - u^{(k)}\|_2 = 0$.*

Proof. Notice that

$$R_1 u^{(k+1)} + R_2 u^{(k)} = \begin{bmatrix} x^{(k)} - \frac{\mu}{\rho} H^\top H x^{(k)} + \frac{\mu}{\rho} H^\top y^{(k)} \\ x^{(k)} + \frac{\mu}{\sigma} H x^{(k+1)} - \frac{\mu}{\sigma} y^{(k)} \end{bmatrix}.$$

Define $r_1(x) := \frac{\rho}{2} \|x - x^{(k)} + \frac{\mu}{\rho} H^\top (Hx^{(k)} - y^{(k)})\|_2^2$ and $r_2(y) := \frac{\sigma}{2} \|y - y^{(k)} - \frac{\mu}{\sigma} (Hx^{(k+1)} - y^{(k)})\|_2^2$. From the definition of the proximity operator and (24), we know that $\Phi(x^{(k+1)}) + r_1(x^{(k+1)}) \leq \Phi(x^k) + r_1(x^k)$ and $\Psi(y^{(k+1)}) + r_2(y^{(k+1)}) \leq \Psi(y^k) + r_2(y^k)$. Summing the last two inequalities leads to

$$(27) \quad \Phi(x^{(k+1)}) + \Psi(y^{(k+1)}) + r_1(x^{(k+1)}) + r_2(y^{(k+1)}) \leq \Phi(x^k) + \Psi(y^k) + r_1(x^k) + r_2(y^k).$$

A direct calculation shows that

$$r_1(x^{(k+1)}) - r_1(x^{(k)}) = \frac{\rho}{2} \|x^{(k+1)} - x^{(k)}\|_2^2 + \mu \langle Hx^{(k)} - y^{(k)}, H(x^{(k+1)} - y^{(k)}) \rangle,$$

and

$$\begin{aligned} & r_2(y^{(k+1)}) - r_2(y^{(k)}) \\ &= \frac{\sigma - \mu}{2} \|y^{(k+1)} - y^{(k)}\|_2^2 + \frac{\mu}{2} \|Hx^{(k+1)} - y^{(k+1)}\|_2^2 - \frac{\mu}{2} \|Hx^{(k+1)} - y^{(k)}\|_2^2. \end{aligned}$$

From (27), (3), and the above two inequalities, we get

$$\begin{aligned} & J(u^{(k+1)}) + \frac{\rho}{2} \|x^{(k+1)} - x^{(k)}\|_2^2 + \frac{\sigma - \mu}{2} \|y^{(k+1)} - y^{(k)}\|_2^2 - \frac{\mu}{2} \|Hx^{(k+1)} - y^{(k)}\|_2^2 \\ & \leq J(u^{(k)}) - \frac{\mu}{2} \|Hx^{(k)} - y^{(k)}\|_2^2 - \mu \langle Hx^{(k)} - y^{(k)}, H(x^{(k+1)} - y^{(k)}) \rangle, \end{aligned}$$

which, by the cosine rule

$$\begin{aligned} \langle Hx^{(k)} - y^{(k)}, H(x^{(k+1)} - y^{(k)}) \rangle &= \frac{1}{2} \|Hx^{(k+1)} - y^k\|_2^2 - \frac{1}{2} \|Hx^k - y^{(k)}\|_2^2 \\ & \quad - \frac{1}{2} \|Hx^{(k+1)} - Hx^{(k)}\|_2^2 \end{aligned}$$

and the inequality

$$\|Hx^{(k+1)} - Hx^{(k)}\|_2^2 \leq \|H\|_2^2 \|x^{(k+1)} - x^{(k)}\|_2^2,$$

becomes

$$(28) \quad J(u^{(k+1)}) + \frac{1}{2}(\rho - \mu \|H\|_2^2) \|x^{(k+1)} - x^{(k)}\|_2^2 + \frac{1}{2}(\sigma - \mu) \|y^{(k+1)} - y^{(k)}\|_2^2 \leq J(u^{(k)}).$$

Since both Φ and Ψ are bounded below, when $\mu < \sigma$ and $\mu \|H\|_2^2 < \rho$ inequality (28) implies that item (i) holds, and the sequence $\{J(u^{(k)}) : k \in \mathbb{N}\}$ is decreasing and convergent. Hence, from inequality (28) again, we can further conclude that item (ii) holds as well. \square

Theorem 3.4. *Let $\alpha, \beta, \rho, \lambda, \sigma,$ and μ be positive numbers, let H be an $m \times n$ matrix, let ψ be a proper convex function on \mathbb{R}^m , and let \mathcal{W} be the sequence generated by the iterative scheme (24). If $\mu < \sigma$ and $\mu \|H\|_2^2 < \rho$, and J has only finite number of critical points on $\mathbb{R}^n \times \mathbb{R}^m$, then the sequence \mathcal{W} converges to a critical point, say (\bar{x}, \bar{y}) , of J . Moreover, if the number a given in (14) is positive and the parameters $\alpha, \beta,$ and μ satisfy **H1** (with ρ in **H1** being $a\mu$), then (\bar{x}, \bar{y}) is a (local) minimizer or a saddle (but not a (local) maximizer) of J . In particular, if the first condition of **H1** holds, then (\bar{x}, \bar{y}) is a solution of model (3).*

Proof. By Lemma 3.3 (ii), the sequence \mathcal{W} is bounded and $\lim_{k \rightarrow \infty} \|u^{(k+1)} - u^{(k)}\|_2 = 0$. And by Lemma 3.2, there exists a subsequence $\{u^{k_j} : j \in \mathbb{N}\}$ of \mathcal{W} converges to a solution of the coupled relations (20) and (21), denoted by $\bar{u} := (\bar{x}, \bar{y})$. Then the subsequence $\{u^{k_j+1} : j \in \mathbb{N}\}$ also converges to \bar{u} . By the iteration scheme (26) and Lemma 3.1, we get that \bar{u} is a fixed-point of $\mathcal{T} \circ R$. Therefore, all accumulation points of sequence \mathcal{W} are fixed-points of mapping $\mathcal{T} \circ R$. Since the number of critical points of J is finite on $\mathbb{R}^n \times \mathbb{R}^m$, we have that the number of accumulation points of \mathcal{W} is finite. Recalling the result of point set topology that if \mathcal{W} is bounded and $\lim_{k \rightarrow \infty} \|u^{(k+1)} - u^{(k)}\| = 0$, the set of accumulations of sequence \mathcal{W} is connected, the sequence \mathcal{W} has only one accumulation point. So, the sequence \mathcal{W} converges to \bar{u} , the fixed-point of $\mathcal{T} \circ R$. And then, we can complete the proof by directly applying Proposition 3. \square

4. Algorithm for the proposed model regularized by TV. In this section, we restrict ourselves on the selections of Ψ (i.e., ψ) and H arising from the TV norm [26] although many other selections, for example, ψ is the ℓ_1 norm and H is a matrix from a tight framelet system, can be considered. For ease of exposition, we assume the vector $z \in \mathbb{R}^n$ is formed from a square image of size $\sqrt{n} \times \sqrt{n}$ by sequentially concatenating the columns of the image.

Let D denote the $\sqrt{n} \times \sqrt{n}$ matrix defined by the equation

$$D := \begin{bmatrix} 0 & & & & & \\ -1 & 1 & & & & \\ & & \ddots & \ddots & & \\ & & & & -1 & 1 \end{bmatrix}$$

and choose H to be an $m \times n$ matrix with $m = 2n$ given by

$$(29) \quad H := \begin{bmatrix} I_{\sqrt{n}} \otimes D \\ D \otimes I_{\sqrt{n}} \end{bmatrix}$$

where $I_{\sqrt{n}}$ is the $\sqrt{n} \times \sqrt{n}$ identity matrix and the notation $P \otimes Q$ denotes the Kronecker product of matrices P and Q . It is known in [21] that $\|H\|_2^2 = 8 \sin^2 \frac{(\sqrt{n}-1)\pi}{2\sqrt{n}}$ (see, e.g., [21]). From the structure of H , one can easily get $a = \min\{\|H_j\|_2^2 : j = 1, 2, \dots, n\} = 4$.

We further define $\psi : \mathbb{R}^m \rightarrow \mathbb{R}$ in (7) at $y \in \mathbb{R}^m$ as

$$(30) \quad \psi(y) := \sum_{j=1}^n \sqrt{y_j^2 + y_{n+i}^2},$$

which is convex.

Lemma 4.1. *Let $\alpha, \beta, \mu, \lambda$ are all positive numbers, if ψ is defined by (30) and H is an $2n \times n$ matrix, then the function J defined by (4) has only finite number of critical points.*

Proof. From the proof of Proposition 1, If $\alpha\beta^4 \leq \frac{4096}{27}$, J is strictly convex and has only one critical point (i.e., the global minimizer) on $\mathbb{R}^n \times \mathbb{R}^m$. So, it remains to prove that $J(x, y), (x, y) \in \mathbb{R}^n \times \mathbb{R}^m$ has only finite number of critical points for $\alpha\beta^4 > \frac{4096}{27}$. To this end, we define

$$g(\omega) := \frac{1}{\omega} + \alpha \left(\omega - \frac{\beta\sqrt{\omega}}{2} \right), \omega \in \mathbb{R}_+.$$

Direct computation shows that

$$\nabla^2 \Phi(x) = \text{diag} \{g(e^{x_i}/f_i) : i = 1, 2, \dots, n\}.$$

Then, if the positive numbers α, β satisfy the restriction $\alpha\beta^4 > \frac{4096}{27}$, there exist ω_1, ω_2 with $0 < \omega_1 \leq \omega_2 < +\infty$ such that $g(\omega) > 0$ holds for $\omega \in (0, \omega_1] \cup [\omega_2, +\infty)$. Let $M_1 := \log(\omega_1 \cdot \inf_i f_i), M_2 := \log(\omega_2 \cdot \sup_i f_i)$. Let $\Omega_1 := \{x : x_i \in (-\infty, M_1], i = 1, 2, \dots, n\}$ and $\Omega_2 := \{x : x_i \in [M_2, +\infty), i = 1, 2, \dots, n\}$. So, for any $x \in \Omega_1 \cup \Omega_2$, we have $\nabla^2 \Phi(x) > 0$. Thus, $\Phi(x)$ is strictly convex on each of Ω_1 and Ω_2 , separately. This, together with the convexity of h and Ψ , implies that $J(x, y)$ is strictly convex for $(x, y) \in \Omega_1 \times \mathbb{R}^m$ and $(x, y) \in \Omega_2 \times \mathbb{R}^m$, separately. As a result, function J has totally no more than two critical points (i.e., local minimizers) on $(\Omega_1 \cup \Omega_2) \times \mathbb{R}^m$.

On the other hand, let $\bar{\Omega} := \mathbb{R}^n \setminus (\Omega_1 \cup \Omega_2)$. It is clear that the set $\bar{\Omega} \times \mathbb{R}^m$ is convex. If $(x, y) \in \bar{\Omega} \times \mathbb{R}^m$ is a critical point of J , then $0 \in \partial J(x, y)$, which can be equivalently written as

$$(31) \quad \begin{cases} 0 = \nabla \Phi(x) + \mu H^\top (Hx - y), \\ 0 \in \mu(y - Hx) + \lambda \partial \psi(y) \Rightarrow \frac{\mu}{\lambda} (Hx - y) \in \partial \psi(y). \end{cases}$$

Let $\tilde{y}_i := [y_i, y_{n+i}]^\top$. We consider the subdifferential of ψ defined by (30) at $y := (y_i) \in \mathbb{R}^m$, which, for $i = 1, 2, \dots, n$, is as follows.

$$(33) \quad \partial(\|\cdot\|_2)(\tilde{y}_i) = \begin{cases} \tilde{y}_i / \|\tilde{y}_i\|_2 & \text{if } \tilde{y}_i \neq 0 \\ \{v : v \in \mathbb{R}^2, \|v\|_2 \leq 1\} & \text{if } \tilde{y}_i = 0. \end{cases}$$

Let $d := Hx - y$ and $\tilde{d}_i := [d_i, d_{n+i}]^\top, i \in \{1, \dots, n\}$. By (32) and (33), if $\tilde{y}_i = 0$, then

$$[[Hx]_i, [Hx]_{n+i}]^\top \in \{v : v \in \mathbb{R}^2, \|v\|_2 \leq \lambda/\mu\}$$

and

$$\tilde{d}_i = [[Hx]_i, [Hx]_{n+i}]^\top;$$

otherwise,

$$(34) \quad \tilde{y}_i = \left(\frac{\lambda}{\mu \|[[Hx]_i, [Hx]_{n+i}]^\top \|_2} + 1 \right) [[Hx]_i, [Hx]_{n+i}]^\top,$$

and thus

$$\tilde{d}_i = -\frac{\lambda}{\mu \|[[Hx]_i, [Hx]_{n+i}]^\top\|_2} [[Hx]_i, [Hx]_{n+i}]^\top.$$

If $\tilde{y}_i \neq 0$, due to $\nabla\Phi(x) = 1 - \frac{f}{e^x} + \alpha(\frac{e^x}{f} - \beta\sqrt{\frac{e^x}{f}})$ being an exponent-based function, equation (31) with respect to x has a finite number of roots on $\bar{\Omega}$; hence, by relation (32), there are finite number of corresponding y -solutions. The finite number of solutions of the coupled equations (31) and (32) on $\bar{\Omega} \times \mathbb{R}^m$, together with the fact that there are at most two critical points for J on $(\Omega_1 \cup \Omega_2) \times \mathbb{R}^m$, implies that $J(x, y), (x, y) \in \mathbb{R}^n \times \mathbb{R}^m$ has only finite number of critical points. This completes the proof. \square

We remark that the above Lemma 4.1 also holds for many other popular selections of ψ and H , e.g., ψ being the ℓ_1 norm and H being a matrix from a tight framelet system.

Theorem 4.2. *Let $\alpha, \beta, \rho, \lambda, \sigma$, and μ be positive numbers, H be an $2n \times n$ matrix, and \mathcal{W} be the sequence generated by the iterative scheme (24). If $\mu < \sigma$ and $\mu \|H\|_2^2 < \rho$, and ψ is a proper convex function defined by (30), then the sequence \mathcal{W} converges to a critical point, say (\bar{x}, \bar{y}) , of J in model (3). Moreover, if a given ρ in (14) is positive and the parameters α, β , and μ satisfy **H1** (with letting ρ in **H1** be $\alpha\mu$), then (\bar{x}, \bar{y}) is a (local) minimizer or a saddle (but not a (local) maximizer) of J . In particular, if the first condition of **H1** holds, then (\bar{x}, \bar{y}) is a solution of model (3).*

Proof. The result can be readily obtained by Lemma 4.1 and Theorem 3.4. \square

In summary, based on the iterative scheme (24) and Theorem 4.2, we end up with Algorithm 1 for model (3).

Algorithm 1 Fixed-point algorithm based on the proximity operators for model (3).

Input: noisy image $f > 0$ in \mathbb{R}^n ; parameters $\lambda > 0, \mu, \beta > 1; \alpha > 0$,
 Initialization: $x^{(0)}$ and $y^{(0)} = 0$; positive numbers σ and ρ such $\mu < \sigma$ and $\frac{\rho}{\mu} > 8 \sin^2 \frac{(\sqrt{n}-1)\pi}{2\sqrt{n}}$.
repeat
 (a) $x^{(k+1)} \leftarrow \text{prox}_{\frac{1}{\rho}\Phi}(x^{(k)} - \frac{\mu}{\rho} H^\top(Hx^{(k)} - y^{(k)}))$,
 (b) $y^{(k+1)} \leftarrow \text{prox}_{\frac{\lambda}{\sigma}\psi}(y^{(k)} + \frac{\mu}{\sigma}(Hx^{(k+1)} - y^{(k)}))$,
until converges or satisfies a stopping criteria.
 Write the output of $x^{(k+1)}$ from the above iteration as \bar{x} .
 The restored image is $u^* = e^{\bar{x}}$.

Implementing Algorithm 1 requires the availability of a convenient way of calculating the two proximity operators, $\text{prox}_{\frac{1}{\rho}\Phi}$ and $\text{prox}_{\frac{\lambda}{\sigma}\psi}$. Firstly, we present the explicit form of $\text{prox}_{\frac{\lambda}{\sigma}\psi}$. For $w \in \mathbb{R}^m$, if $v = \text{prox}_{\frac{\lambda}{\sigma}\psi}(w)$, then

$$\begin{bmatrix} v_i \\ v_{i+m/2} \end{bmatrix} = \max \left\{ 1 - \frac{\lambda}{\sigma \sqrt{w_i^2 + w_{i+m/2}^2}}, 0 \right\} \begin{bmatrix} w_i \\ w_{i+m/2} \end{bmatrix}.$$

Secondly, we describe an algorithm to evaluate the operator $\text{prox}_{\frac{1}{\rho}\Phi}$ at $x \in \mathbb{R}^n$. Let $u = \text{prox}_{\frac{1}{\rho}\Phi}(x)$. Then by the definition of the proximity operator, z is a solution of the following equation

$$\rho(z - x) + \nabla\Phi(z) = 0.$$

For the parameters α , β , and ρ fulfilling assumption **H1**, the above equation has a unique solution by Lemma 2.2; thus we can determine the solution efficiently by Newton’s method.

5. A review of several existing models. To prepare a comparison study on the performance of our proposed model (3) for multiplicative noise removal with existing models in the next section, several state-of-the-art models and the associated algorithms will be briefly reviewed in this section. The models included in our study are the DZ model in [11], the TwL- m V model in [17], the I-DIV model in [29], and the HMNZ model in [14].

The DZ model reads as follows

$$(35) \quad \min_{u \in X} \left\langle \log u + \frac{f}{u}, \mathbf{1} \right\rangle + \alpha \left\| \sqrt{\frac{u}{f}} - \mathbf{1} \right\|_2^2 + \lambda\psi(Hu),$$

where $\alpha \geq \frac{2\sqrt{6}}{9}$, $\lambda > 0$, $X := \{u \in \mathbb{R}^n : u > 0 \text{ and } \langle u, \mathbf{1} \rangle = \langle f, \mathbf{1} \rangle\}$, H and ψ are given by (29) and (30), respectively. This model was solved by a primal-dual algorithm: Set an initial estimate $(\tilde{u}^{(0)}, u^{(0)}, p^{(0)}) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^{2n}$, iterate for $k = 0, 1, \dots$

$$(36) \quad \begin{cases} p^{(k+1)} = \underset{p \in Y}{\text{argmin}} \lambda \langle \tilde{u}^{(k)}, H^\top p \rangle + \frac{1}{2\beta} \|p - p^{(k)}\|_2^2, \\ u^{(k+1)} = \underset{u \in X}{\text{argmin}} \left\langle \log u + \frac{f}{u}, \mathbf{1} \right\rangle + \alpha \left\| \sqrt{\frac{u}{f}} - \mathbf{1} \right\|_2^2 \\ \quad + \lambda \langle u, H^\top p^{(k+1)} \rangle + \frac{1}{2\mu} \|u - u^{(k)}\|_2^2, \\ \tilde{u}^{(k+1)} = 2u^{(k+1)} - u^{(k)}, \end{cases}$$

where $Y := \{q \in \mathbb{R}^{2n} : \max_{i \in \mathbb{N}_n} \sqrt{q_i^2 + q_{i+n}^2} \leq 1\}$. The vector $p^{(k+1)}$ is actually the projection of $p^{(k)} + \lambda\beta H\tilde{u}^{(k)}$ onto the convex set Y , which can be easily obtained. As suggested in [11], the vector $u^{(k+1)}$ was approximated by using the Newton method following with one projection step onto the convex set X .

The TwL- m V model in [17] is a two-level relaxation of the m V model (see [32]) by using the m th root transformation [32] and the concave conjugate. It is

$$(37) \quad \begin{aligned} (v^*, a^*) &= \underset{v \in \sqrt[m]{U}, a > 0}{\text{argmin}} \frac{1}{s} \langle a, v^s \rangle - \frac{1}{s} \langle m \log a - \frac{sf}{v^m}, \mathbf{1} \rangle + \lambda\psi(Hv), \\ u^* &= (v^*)^m, \end{aligned}$$

where $\lambda > 0$, $m \geq 1$, $s \geq 1$, $U := [\epsilon, C]^n$, and $\sqrt[m]{U} = [\sqrt[m]{\epsilon}, \sqrt[m]{C}]^n$ with both ϵ and C being positive, H and ψ are given by (29) and (30), respectively. The following algorithm was suggested for solving (37) in [17],

$$(38) \quad \begin{cases} a^{(k+1)} = \underset{a > 0}{\text{argmin}} \frac{1}{s} \langle a, (v^{(k)})^s \rangle - \frac{m}{s} \langle \log a, \mathbf{1} \rangle, \\ v^{(k+1)} = \underset{v \in \sqrt[m]{U}}{\text{argmin}} \frac{1}{s} \langle a^{(k+1)}, v^s \rangle + \langle \frac{f}{v^m}, \mathbf{1} \rangle + \lambda\psi(Hv). \end{cases}$$

It is easy to see that $a^{(k+1)} = \frac{m}{(v^{(k)})^s}$. The update $v^{(k+1)}$ is approximated by the following iterative scheme

$$(39) \quad \begin{cases} v^{(k,\ell+1)} = \text{Proj}_{\sqrt[m]{U}} \left(v^{(k,\ell)} - \rho \left(a^{(k+1)} \odot (v^{(k,\ell)})^{s-1} - \frac{mf}{(v^{(k,\ell)})^{m+1}} + H^\top p^{(k,\ell)} \right) \right), \\ z^{(k,\ell+1)} = \text{prox}_{\frac{\lambda}{\alpha}\psi} \left(H v^{(k,\ell+1)} - \frac{1}{\alpha} p^{(k,\ell)} \right), \\ p^{(k,\ell+1)} = z^{(k,\ell)} + \alpha(z^{(k,\ell+1)} - H v^{(k,\ell+1)}), \end{cases}$$

where $\text{Proj}_{\sqrt[m]{U}}$ is the projection operator onto the set $\sqrt[m]{U}$ and $1 \leq \ell \leq T$ for a pre-given positive integer T . The output $v^{(k,T+1)}$ will be considered as $v^{(k+1)}$ in the second step of (38).

The I-DIV model in [29] reads as

$$(40) \quad \min \{ \langle u - f \log u, \mathbf{1} \rangle + \lambda \psi(Hu), u \in \mathbb{R}_+^n \},$$

where $\lambda > 0$, H and ψ are given by (29) and (30), respectively. This model can be equivalently written as

$$(41) \quad \min_{(u,v,p) \in \mathbb{R}_+^n \times \mathbb{R}_+^n \times \mathbb{R}^{2n}} \left\{ \langle u - f \log u, \mathbf{1} \rangle + \lambda \psi(p) : \begin{bmatrix} u \\ p \end{bmatrix} = \begin{bmatrix} I \\ H \end{bmatrix} v \right\}.$$

The model above is solved by using the alternating direction method of multipliers (ADMM) in the following way:

$$(42) \quad \begin{cases} v^{(k+1)} = \underset{v \in \mathbb{R}_+^n}{\text{argmin}} \left\| \begin{bmatrix} b_1^{(k)} \\ b_2^{(k)} \end{bmatrix} + \begin{bmatrix} I \\ H \end{bmatrix} v - \begin{bmatrix} u^{(k)} \\ p^{(k)} \end{bmatrix} \right\|_2^2, \\ \begin{bmatrix} u^{(k+1)} \\ p^{(k+1)} \end{bmatrix} = \underset{u \in \mathbb{R}_+^n, p \in \mathbb{R}^{2n}}{\text{argmin}} \left\{ \langle u - f \log u, \mathbf{1} \rangle + \lambda \psi(p) \right. \\ \left. + \frac{\alpha}{2} \left\| \begin{bmatrix} b_1^{(k)} \\ b_2^{(k)} \end{bmatrix} + \begin{bmatrix} I \\ H \end{bmatrix} v^{(k+1)} - \begin{bmatrix} u \\ p \end{bmatrix} \right\|_2^2 \right\}, \\ b^{(k+1)} = b^{(k)} + \begin{bmatrix} I \\ H \end{bmatrix} v^{(k+1)} - \begin{bmatrix} u^{(k+1)} \\ p^{(k+1)} \end{bmatrix}. \end{cases}$$

The HMNZ method in [14] is a learned dictionary based de-speckling method. The first phase of the method is to learn a dictionary by using the K-SVD method. Let D be the learned dictionary obtained from $\log f$, $R_{i,j} \in \mathbb{R}^{d \times n}$ is the matrix corresponding to the extraction of the patch located in (i, j) , and $D\alpha_{i,j}$ be the sparse dictionary representation obtained for each patch $R_{i,j} \log f$. Define

$$G(v) := \frac{1}{\lambda} \left\{ \mu \langle v + f e^{-v}, \mathbf{1} \rangle + \frac{1}{2} \langle Wv, v \rangle - \langle M, v \rangle \right\},$$

where $\lambda, \mu > 0$, and $W = \sum_{i,j} R_{i,j}^T R_{i,j}$, $M = \sum_{i,j} R_{i,j} D\alpha_{i,j}$. Then the second phase is to solve the following variational model:

$$(43) \quad \begin{aligned} v^* &= \underset{v \in \mathbb{R}^n}{\text{argmin}} \lambda \{ G(v) + \psi(Hv) \}, \\ u^* &= e^{v^*}, \end{aligned}$$

where H and ψ are given by (29) and (30), respectively. And the model can be solved by using Chambolle-Pock algorithm in the following way:

$$(44) \quad \begin{cases} p^{(k+1)} = \underset{\|p\|_\infty \leq 1}{\text{argmin}} \{ -\langle p, \nabla \bar{v}^{(k)} \rangle + \beta \|p - p^{(k)}\|^2 \}, \\ v^{(k+1)} = \underset{v \in \mathbb{R}^n}{\text{argmin}} \{ G(v) - \langle \text{div} p^{(k+1)}, v \rangle + \beta \|v - v^{(k)}\|^2 \}, \\ \bar{v}^{(k+1)} = 2v^{(k+1)} - v^{(k)}. \end{cases}$$

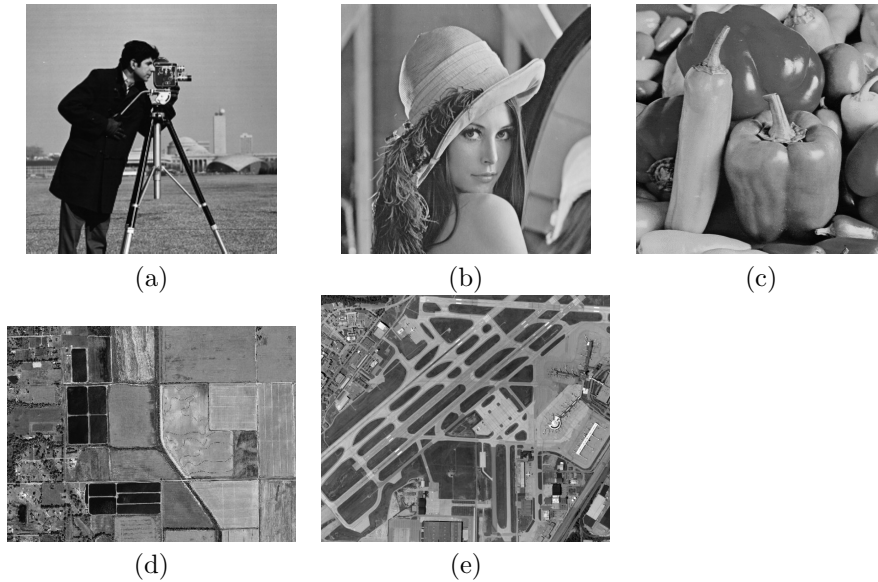


FIGURE 1. Five gray-level test images. (a) “Cameraman” (512×512). (b) “Lena” (512×512). (c) “Peppers” (512×512). (d) “Remote1” (768×574). (e) “Remote2” (632×540).

6. Experiments. In this section, we illustrate the effectiveness of the proposed model (3) and the corresponding algorithm (i.e., Algorithm 1) in producing high-quality images for images corrupted by multiplicative noise. In our experiments, we use test images of “Cameraman”, “Lena”, “Peppers” with size of 512×512 , “Remote1” with size of 768×574 and “Remote2” with size of 632×540 , as shown in Fig. 1. To generate the observed images f in (1), we degrade the original test images by multiplicative Gamma noise at various noise-levels.

TABLE 1. Parameter values in our algorithm (Algorithm 1) at various noise levels.

	λ	α	β	μ	ρ	σ
$L = 10$	0.306	0.0015	6.06	30	250	150
$L = 6$	0.406	0.00085	15.02	30	250	150
$L = 4$	0.506	0.000108	250	30	255.5	90.26
$L = 2$	0.8	0.00001	1655.05	30	290	156.26

In order to have a through comparison of the performance of the proposed model (3) and the corresponding algorithm, we will compare our method with some state-of-the-art methods, the DZ method in [11], the TwL- mV method in [17], the HMNZ method in [14] (which remarkably outperforms the BS [5], DFN [12], DRSM [29], and NL [30] methods in terms of the quality of the restored images, as discussed in [14]) and the I-DIV method in [29]; and these four methods were reviewed shortly in the previous section. All the algorithms are performed under Windows 7 using MATLAB 7.0(R14) on a Intel Dual-Core i5-4570 CPU 3.20 GHz PC with 4.0G RAM memory. The execution time required in each method is measured by seconds. We terminate the tested algorithms when $\|u^{(k+1)} - u^{(k)}\|_2 / \|u^{(k)}\|_2 \leq 3 \times 10^{-4}$. Here

$u^{(k)}$ is the resulting image of the k th iteration produced by the underlying algorithm. The quality of the results is evaluated in terms of the peak signal-to-signal ratio (PSNR) in dB as

$$\text{PSNR} = 10 \log_{10} \left(\frac{255^2 n}{\|\tilde{u} - u^*\|_2^2} \right),$$

where \tilde{u} is the original image with n the total number of pixels and u^* is the recovered image by a testing algorithm.

Two types of experiments are designed. Experiment 1 will discuss the influence of initialization $x^{(0)}$ for the performance of our proposed Algorithm 1. Experiment 2 will focus on the comparison of our proposed model (3) with others, namely, the DZ, TwL- m V, HMNZ and I-DIV methods.

6.1. Experiment 1. In this subsection, we present a numerical study on how an initial estimate $x^{(0)}$ affects the performance of our Algorithm 1.

TABLE 2. Parameter values for all testing algorithms.

L	Method	λ	α	β	μ	ρ	σ
10	Ours	0.569	1.0002	1.00053	20.1	425.5	20.140
	TwL-4V	$3.6/L$	1.0	—	—	0.3	—
	I-DIV	0.31	378.0	—	—	—	—
	DZ	0.07	19	3.0	3.0	—	—
	HMNZ	0.1	—	10	17.5	—	—
6	Ours	0.555	0.3885	1.005	19.6	285.5	19.667
	TwL-4V	$2.9/L$	1.0	—	—	0.29	—
	I-DIV	0.45	918.0	—	—	—	—
	DZ	0.06	3.8	3.0	3.0	—	—
	HMNZ	0.1	—	10	9	—	—
4	Ours	0.659	0.18515	1.0105	29.658	255.5	29.918
	TwL-4V	$2.4/L$	1.0	—	—	0.3	—
	I-DIV	0.55	658.0	—	—	—	—
	DZ	0.05	1.59	3.0	3.0	—	—
	HMNZ	0.1	—	10	6	—	—
2	Ours	0.8	0.000001	115.0	21.0	168	26.26
	TwL-4V	$1.8/L$	1.0	—	—	0.3	—
	I-DIV	0.84	1059.0	—	—	—	—
	DZ	0.065	0.45	3.0	3.0	—	—
	HMNZ	0.1	—	10	1.5	—	—

We first choose the test image of “Cameraman” and degrade it by multiplicative noise with noise levels $L = 10, 6, 4,$ and 2 . For each noise level, the following four different initializations are considered: (i) $x^{(0)} = \log(f)$; (ii) $x^{(0)} = \log(\text{MEAN}(f))$; (iii) $x^{(0)} = \log(\text{DZ}(f))$; and (iv) $x^{(0)} = \log(\text{SAR-BM3D}(f))$. The choice of $x^{(0)} = \log(f)$ is typical since the observation f is supposed to be close to the desired image. The image $\text{MEAN}(f)$ is the result of f filtered by the 5×5 average filter. The image $\text{DZ}(f)$ is the denoised image from the DZ model. Finally, the image $\text{SAR-BM3D}(f)$ is the denoised image by the SAR-BM3D method [25]. We note that the BM3D represents the state-of-the-art algorithm for recovering images affected by white Gaussian noise [6] and SAR-BM3D is a new version of BM3D which is adapted to denoise SAR images by taking into account their peculiar features. By using $x^{(0)} = \log(f)$ as the initial estimate, the parameters involved in Algorithm 1 are chosen differently for multiplicative noise at different levels to obtain the convergently stable PSNR values as high as possible. The values of these parameters are listed in Table 1. With these parameters, we can readily verify that model (3) is convex for $L = 10$ and 6 , but is nonconvex for $L = 4$ and 2 . Fig. 2 illustrates the curves of

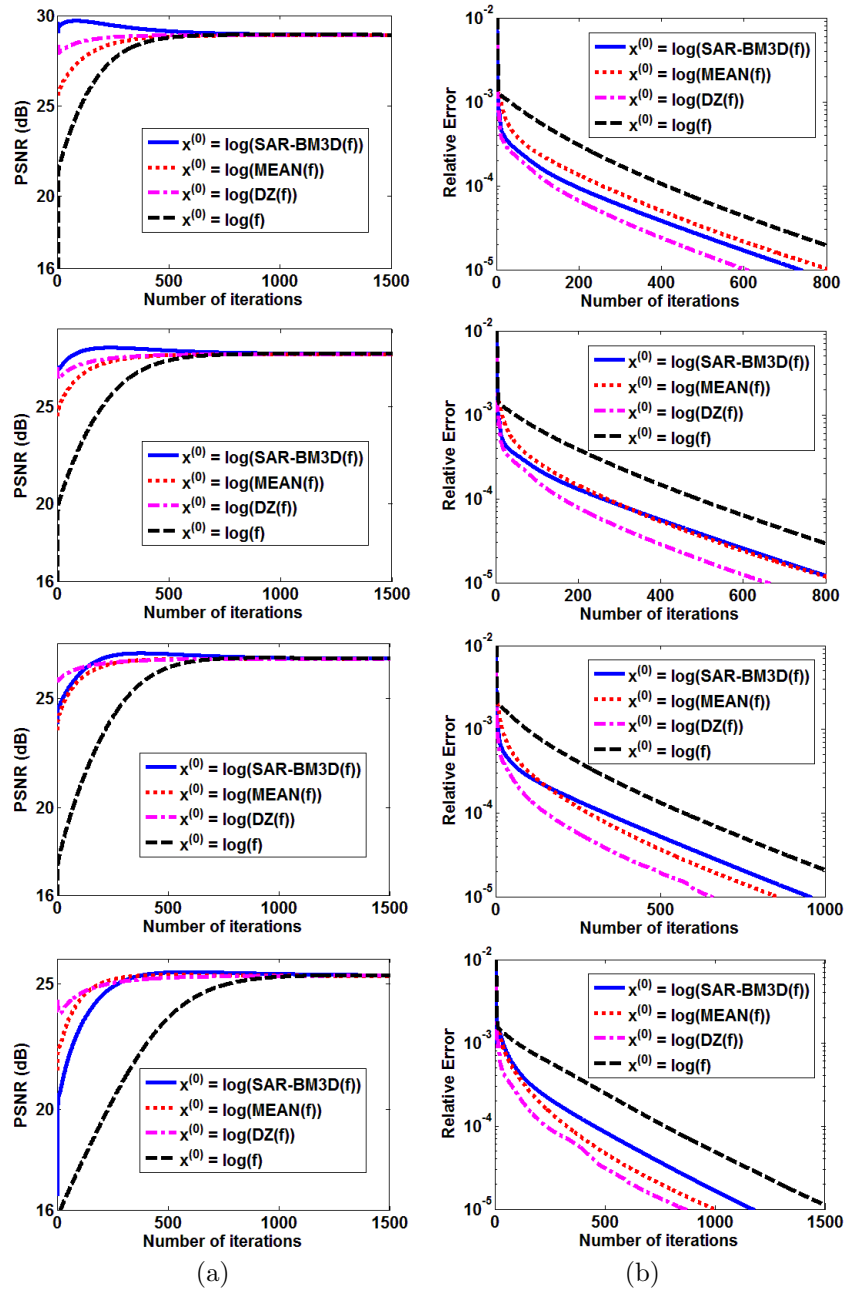


FIGURE 2. (a) PSNR versus number of iterations. (b) Relative error versus number of iterations. Here, various $x^{(0)}$ s are used as initial estimates in Algorithm 1. From top to bottom: the test images are the degraded “Cameraman” with multiplicative noise at levels $L = 10, 6, 4,$ and 2 .

PSNR values of the updates (column (a)) and the relative errors (column (b)) with

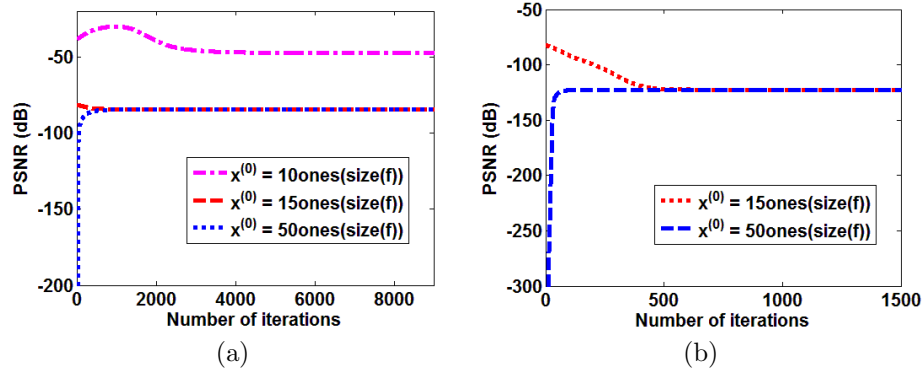


FIGURE 3. (a) PSNR versus number of iterations for $L = 4$. (b) PSNR versus number of iterations for $L = 2$. Here, various $x^{(0)}$ s are used as initializations in Algorithm 1 for the degraded “Cam-eraman” with multiplicative noise.

respect to the number of iterations at noise levels $L = 10, 6, 4$, and 2 (from top to bottom). It is not surprise that at noise level $L = 10$ and 6 with sufficient number of iterations Algorithm 1 with the four different initial estimates produces the restored images of similar quality in terms of PSNR values. For $L = 4$ and 2 , with the above four different initializations of $x^{(0)}$, Algorithm 1 also generates the restored images of similar quality in terms of PSNR values (see the third and fourth subfigures of Fig. 2(a)) even though model (3) is nonconvex. It can be observed that the curves of PSNR values associated with the initialization $x^{(0)} = \log(\text{SAR-BM3D}(f))$ always have clear peaks in comparison with the other three initializations, $x^{(0)} = \log(f)$, $x^{(0)} = \log(\text{MEAN}(f))$ and $x^{(0)} = \log(\text{DZ}(f))$. The relative errors corresponding to these peaks are 2.1×10^{-4} , 1.1×10^{-4} , 8.4×10^{-5} , and 6.5×10^{-5} for $L = 10, 6, 4$, and 2 , respectively. In addition, we also test the influence of other three initializations $x^{(0)} = \log(\text{TwL-4V}(f))$, $x^{(0)} = \log(\text{I-DIV}(f))$ and $x^{(0)} = \log(\text{HMNZ}(f))$ on our Algorithm 1. Here $\text{TwL-4V}(f)$, $\text{I-DIV}(f)$ and $\text{HMNZ}(f)$ are the denoised images from the TwL-4V, I-DIV and HMNZ methods, respectively. We observe that the PSNR-curves associated with those initializations for different noise levels are absent of peaks that are higher than the final stable values.

Recall that for $L = 4$ and 2 , model (3) may be nonconvex with the parameters in Table 1. According to Theorem 4.2, Algorithm 1 may converge to a local minimizer or a saddle, but not a local maximizer of J . To this end, three different initializations, $x^{(0)} = 10\text{ones}(\text{size}(f))$, $x^{(0)} = 15\text{ones}(\text{size}(f))$, $x^{(0)} = 50\text{ones}(\text{size}(f))$ for $L = 4$ and two different initializations, $x^{(0)} = 15\text{ones}(\text{size}(f))$, $x^{(0)} = 50\text{ones}(\text{size}(f))$ for $L = 2$ are tested. Here $\text{ones}(\text{size}(f))$ is the matrix whose entries are ones and has the same size as f . Fig. 3 illustrates the curves of PSNR values of the updates against the number of iterations. One can see that for $L = 4$, Algorithm 1 with $x^{(0)} = 10\text{ones}(\text{size}(f))$ converges to a stable PSNR of -47.43 dB while Algorithm 1 with $x^{(0)} = 15\text{ones}(\text{size}(f))$ and $x^{(0)} = 50\text{ones}(\text{size}(f))$ converges to a stable PSNR of -84.52 dB; for $L = 2$, Algorithm 1 with $x^{(0)} = 15\text{ones}(\text{size}(f))$ and $x^{(0)} = 50\text{ones}(\text{size}(f))$ uniformly converges to a stable PSNR of -122.58 dB. All stable PSNR values for $L = 4$ and 2 are different from the stable values in the third and fourth subfigures of Fig. 2(a), respectively. These results confirm that J with

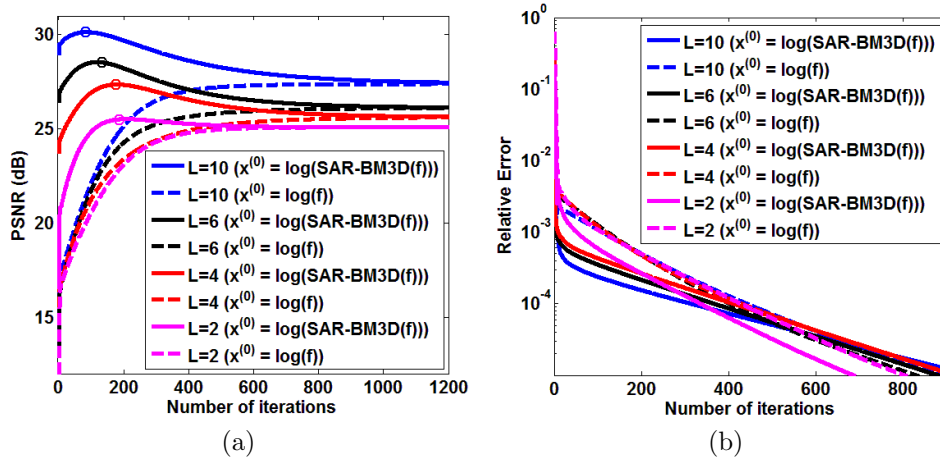


FIGURE 4. (a) PSNR versus number of iterations. (b) Relative error versus number of iterations. Here, two different $x^{(0)}$ s are used as initial estimates in Algorithm 1. The solid lines are plotted firstly by selecting the parameters of Algorithm 1 to obtain (nearly) optimal PSNR values (marked by ‘o’) under the prescribed tolerance $TOL = 3 \times 10^{-4}$; then the dashed lines are plotted by using the same parameters as those of corresponding solid lines. The test images are the degraded “Cameraman” with multiplicative noise at various noise levels ($L = 10, 6, 4,$ and 2).

the parameters in Table 2 has more than one critical points for $L = 4$ and 2 ; and the restored images may be sensitive to the selected initial estimates. Experimentally, we find that, for $L = 4, 2$, if the value of each entry of $x^{(0)}$ is restricted to the interval $(-\infty, \log 255]$, Algorithm 1 can consistently converge to their respective fixed stable PSNR values as shown in the third and fourth subfigures of Fig. 2(a).

6.2. Experiment 2. Motivated by the observations in our first experiment, we use $\log(\text{SAR-BM3D}(f))$ as the initial estimate and re-select the parameters for Algorithm 1 to make the PSNR value of a restored image as high as possible under the prescribed tolerance $TOL = 3 \times 10^{-4}$. These parameters are listed in Table 2. For the purpose of illustration, the solid lines record the PSNR value (in Fig. 4(a)) and the relative error (in Fig. 4(b)) of the denoised image at each iteration for different noise-levels, $L = 10, 6, 4,$ and 2 up to the first 12000 iterations. Algorithm 1 actually terminates with the PSNR values of 30.15 dB, 28.53 dB, 27.38 dB and 25.47 dB (marked by ‘o’) for $L = 10, 6, 4,$ and 2 respectively, when the stopping criteria is met. The PSNR values are almost the same as the values of the upmost points of the respective lines. For a comparison purpose, the curves (dashed lines) in Fig. 4 are generated from Algorithm 1 with $\log(f)$ as the initial estimate.

Table 3 reports the PSNR values of the noisy images restored through our proposed Algorithm 1 with using $\log(\text{SAR-BM3D}(f))$ as the initial estimate, the I-DIV method, the DZ method, the TwL-4V method, the HMNZ method, and SAR-BM3D for multiplicative noise removal. Correspondingly, Table 2 lists the values of parameters for our method, the I-DIV method, the DZ method, the HMNZ method, and

TABLE 3. PSNR (dB) and CPU time (s) for I-DIV [29], DZ [11], TwL-4V [17], HMNZ [14], our algorithm (Algorithm 1 with $x^{(0)} = \log(\text{SAR-BM3D}(f))$), and SAR-BM3D [25] for test images of Fig. 1 corrupted by multiplicative noise with $L = 10, 6, 4, 2$, respectively.

Image	L		Noisy	I-DIV	DZ	TwL-4V	HMNZ	Ours	SAR-BM3D	
Camer.	10	PSNR	15.61	28.69	28.30	28.83	29.96	30.15	28.89	
		Time	—	6.28	86.30	6.23	66.62	117.35+6.03	117.35	
	6	PSNR	13.39	27.35	26.93	27.59	28.46	28.53	26.35	
		Time	—	11.40	110.36	7.33	62.86	116.59+10.11	116.59	
	4	PSNR	11.64	26.52	25.85	26.72	27.33	27.38	23.67	
		Time	—	13.57	149.54	7.94	63.69	118.31+13.31	118.31	
	2	PSNR	8.63	24.98	24.27	25.28	25.30	25.47	16.45	
		Time	—	15.57	190.75	9.72	62.58	124.30+15.20	124.30	
	Lena	10	PSNR	15.64	28.47	27.51	28.60	29.41	29.65	28.48
			Time	—	6.57	85.90	6.85	65.15	118.64+6.33	118.64
		6	PSNR	13.42	27.34	26.24	27.48	28.07	28.14	25.92
			Time	—	12.06	106.87	7.29	63.69	116.62+10.42	116.62
4		PSNR	11.68	26.64	25.45	26.72	27.01	27.37	23.63	
		Time	—	13.78	144.34	8.48	63.03	118.32+13.14	118.32	
2		PSNR	8.71	25.07	24.01	25.17	25.21	25.50	16.46	
		Time	—	17.16	181.92	10.73	62.88	124.56+15.71	124.56	
Pepp.		10	PSNR	15.93	28.83	27.20	28.86	29.13	29.53	28.09
			Time	—	6.88	84.15	7.47	65.33	116.86+5.62	116.86
		6	PSNR	13.70	27.95	26.15	27.92	28.12	28.51	25.78
			Time	—	12.46	108.44	8.16	60.54	118.31+10.24	118.31
	4	PSNR	11.98	27.10	25.10	27.05	27.14	27.76	23.63	
		Time	—	13.30	145.13	8.54	63.75	119.16+12.90	119.16	
	2	PSNR	8.93	25.57	23.72	25.54	25.52	25.97	16.45	
		Time	—	17.44	188.50	10.59	61.59	115.71+15.03	115.71	
	Rem.1	10	PSNR	16.27	25.22	25.15	25.33	26.16	26.21	25.43
			Time	—	12.94	190.94	12.78	108.13	210.80+11.14	210.80
		6	PSNR	14.00	24.17	24.01	24.41	25.02	25.07	23.64
			Time	—	23.09	222.15	13.63	102.92	211.02+23.01	211.02
4		PSNR	12.28	23.48	23.01	23.68	24.03	23.98	22.20	
		Time	—	27.56	286.88	14.95	105.97	211.00+26.75	211.00	
2		PSNR	9.28	22.39	21.80	22.57	22.63	22.76	16.45	
		Time	—	32.09	347.21	17.75	99.72	211.89+28.51	211.89	
Rem.2		10	PSNR	16.23	25.56	25.59	25.63	26.76	26.69	25.92
			Time	—	9.16	144.40	9.58	84.96	163.00+8.90	163.00
		6	PSNR	14.02	24.55	24.30	24.45	25.25	25.43	24.07
			Time	—	17.46	164.92	9.94	85.24	162.70+17.03	162.70
	4	PSNR	12.27	23.45	23.38	23.63	23.93	24.26	22.26	
		Time	—	22.75	219.99	11.79	78.15	163.07+20.35	163.07	
	2	PSNR	9.22	22.01	21.86	22.18	22.12	22.72	16.08	
		Time	—	25.86	271.57	14.01	75.17	162.84+24.02	162.84	

the TwL-4V method at various noise levels. We can see that our method outperforms the other five state-of-the-art methods in terms of PSNR values. Especially for the low level noisy case $L = 10$, it is significantly superior to the I-DIV method, the DZ method, the TwL-4V method and the SAR-BM3D, about 1.0 dB. We remark that unlike the formulation of multiplicative noise in model (3), the DZ model, the TwL-4V model, the I-DIV model and HMNZ model, the multiplicative noise model discussed in SAR-BM3D [25] follows a square-root Gamma distribution. This is the reason why the PSNR values of the noisy images at various noise levels in Table 3 in present paper are inconsistent with those in Table I in [25] (eg. see the results of “Lena” image in both tables). When focusing on the CPU time of the algorithms in Table 3, we see that our Algorithm 1 is time consuming in comparison with the other four methods. It is due to the fact that the total execution time of our Algorithm 1 consists of two components: the time for initializing $x^{(0)}$ by running SAR-BM3D on f and the subsequent time for the developed fixed-point iterative

scheme, as listed in Algorithm 1; and the SAR-BM3D holds a large proportion of the time while the fixed-point iterative scheme is very fast.

For visual comparisons, the denoised images of “Cameraman” and “Remote1” by various methods are displayed in Figs. 5 and 6. We observe that the restored images by our proposed method have less artifacts than those by I-DIV, DZ, TwL-4V and HMNZ, can keep sharp edges and fine details (e.g., see the background and the remote columnar building in “Cameraman” image).

7. Conclusion. This paper introduces a new variational model for multiplicative noise removal by exploring approximate sparsity of underlying images. In order to solve the proposed model, we first study the convexity of the objective function of the model. We give the conditions that can ensure the convexity of the model. And then we relax the conditions such that the model may be nonconvex; and we show that the critical points of the objective function of the (nonconvex) model can be viewed as the solutions of a coupled fixed-point equations in terms of the proximity operators. In particular, under certain conditions, we show that the global minimizer of the model is a solution of the fixed-point equations. Based on the fixed-point equations, an proximity algorithm is developed; and we prove that the sequence generated from the algorithm converges to one of the critical points of the the objective function of the model for some specific cases. Furthermore, We examine the suitability of the model and the corresponding algorithm for restoring images contaminated by multiplicative noise. Our numerical results indicate that our method evidently performs better than existing state-of-the-art methods in terms of PSNR values.

Acknowledgments. The authors would like to thank Dr. Hyenkyun Woo for providing the source code of TwL- m V method, Dr. Tiejong Zeng for the DZ method, and Dr. Yu-mei Huang for the HMNZ method. Jian Lu and Chen Xu were supported in part by the National Natural Science Foundation of China, under Grant 61373087, Grant 61472257, Grant 11201312, and Grant 61272252, the Guangdong Natural Science Foundation of China under Grant 2015A030313557 and Grant 2015A030313550, the China Scholarship Council, the Foundation for Distinguished Young Teachers of Higher Education in Guangdong, China, under Grant Yq2013144, the Specialized Research Fund for the Doctoral Program of Chinese Higher Education under Grant 20134408110001. Lixin Shen was supported in part by the US National Science Foundation under Grant DMS-1522332. Yuesheng Xu was supported in part by the Special Project on High-performance Computing under the National Key R&D Program (No. 2016YFB0200602), and by the Natural Science Foundation of China under grant 11471013.

REFERENCES

- [1] G. Aubert and J. Aujol, [A variational approach to removing multiplicative noise](#), *SIAM J. Appl. Math.*, **68** (2008), 925–946.
- [2] R. Bamler, Principles of synthetic aperture radar, *Surv. Geophys.*, **21** (2000), 147–157.
- [3] H. L. Bauschke and P. L. Combettes, [Convex Analysis and Monotone Operator Theory in Hilbert Spaces](#), AMS Books in Mathematics, Springer New York, 2011.
- [4] J. M. Bioucas-Dias and M. A. T. Figueiredo, [Multiplicative noise removal using variable splitting and constrained optimization](#), *IEEE Trans. Image Process.*, **19** (2010), 1720–1730.
- [5] M. F. C. Chesneau and J. Starck, [Stein block thresholding for image denoising](#), *Appl. Computat. Harmon. Anal.*, **28** (2010), 67–88.

- [6] K. Dabov, A. Foi, V. Katkovnik and K. Egiazarian, [Image denoising by sparse 3d transform-domain collaborative filtering](#), *IEEE Trans. Image Process.*, **16** (2007), 2080–2095.
- [7] D. Dai, L. Shen, Y. Xu and N. Zhang, [Noisy 1-bit compressive sensing: Models and algorithms](#), *Appl. Computat. Harmon. Anal.*, **40** (2016), 1–32.
- [8] I. Daubechies, [Ten Lectures on Wavelets](#), vol. 61 of CBMS Conf. Series Appl. Math., SIAM, Philadelphia, 1992.
- [9] I. Daubechies, B. Han, A. Ron and Z. Shen, [Framelets: MRA-based constructions of wavelet frames](#), *Appl. Comput. Harmon. Anal.*, **14** (2003), 1–46.
- [10] L. Denis, F. Tupin, J. Darbon and M. Sigelle, [SAR image regularization with fast approximation discrete minimization](#), *IEEE Trans. Image Process.*, **18** (2009), 1588–1600.
- [11] Y. Dong and T. Zeng, [A convex variational model for restoring blurred images with multiplicative noise](#), *SIAM J. Imag. Sci.*, **6** (2013), 1598–1625.
- [12] S. Durand, J. Fadili and M. Nikolova, [Multiplicative noise removal using l1 fidelity on frame coefficients](#), *J. Math. Imag. Vis.*, **36** (2010), 201–226.
- [13] J. W. Goodman, [Some fundamental properties of speckle](#), *J. Opt. Soc. of Amer.*, **66** (1976), 1145–1150.
- [14] Y. Hang, L. Moisan, M. K. Ng and T. Zeng, [Multiplicative noise removal via a learned dictionary](#), *IEEE Trans. Image Process.*, **21** (2012), 4534–4543.
- [15] Y.-H. Huang, H. Yan and T. Zeng, [Multiplicative noise removal based on unbiased box-cox transformation](#), *Communications in Computational Physics*, **22** (2017), 803–828.
- [16] Y.-M. Huang, M. K. Ng and Y.-W. Wen, [A new total variation method for multiplicative noise removal](#), *SIAM J. Imag. Sci.*, **2** (2009), 20–40.
- [17] M. Kang, S. Yun and H. Woo, [Two-level convex relaxed variational model for multiplicative denoising](#), *SIAM J. Imag. Sci.*, **6** (2013), 875–903.
- [18] D. Lazard, [Quantifier elimination: Optimal solution for two classical examples](#), *J. Symbol. Comput.*, **5** (1988), 261–266.
- [19] F. Li, M. K. Ng and C. Shen, [Multiplicative noise removal with spatially varying regularization parameters](#), *SIAM J. Imag. Sci.*, **3** (2010), 1–20.
- [20] J. Lu, L. Shen, C. Xu and Y. Xu, [Multiplicative noise removal in imaging: An exp-model and its fixed-point proximity algorithm](#), *Appl. Comput. Harmon. Anal.*, **41** (2016), 518–539.
- [21] C. A. Micchelli, L. Shen and Y. Xu, [Proximity algorithms for image models: Denoising](#), *Inverse Probl.*, **27** (2011), 045009(30pp).
- [22] J.-J. Moreau, [Fonctions convexes duales et points proximaux dans un espace hilbertien](#), *C.R. Acad. Sci. Paris Sér. A Math.*, **255** (1962), 2897–2899.
- [23] Y. Nesterov, [Introductory Lectures on Convex Optimization](#), Kluwer, Boston, 2004.
- [24] C. Oliver and S. Quegan, [Understanding Synthetic Aperture Radar Imaging](#), SciTech Publishing, Raleigh, NC, 2004.
- [25] S. Parrilli, M. Poderico, C. V. Angelino and L. Verdoliva, [A nonlocal SAR image denoising algorithm based on LLMMSE wavelet shrinkage](#), *IEEE Trans. Geosci. Remote Sens.*, **50** (2012), 606–616.
- [26] L. Rudin, S. Osher and E. Fatemi, [Nonlinear total variation based noise removal algorithms](#), *Phys. D*, **60** (1992), 259–268.
- [27] J. M. Schmitt, S. H. Xiang and K. M. Yung, [Speckle in optical coherence tomography](#), *J. Biomed. Opt.*, **4** (1999), 95–105.
- [28] J. Shi and S. Osher, [A nonlinear inverse scale space method for a convex multiplicative noise removal](#), *SIAM J. Imag. Sci.*, **1** (2008), 294–321.
- [29] G. Steidl and T. Teuber, [Removing multiplicative noise by Douglas-Rachford splitting methods](#), *J. Math. Imag. Vis.*, **36** (2010), 168–184.
- [30] T. Teuber and A. Lang, [Nonlocal filters for removing multiplicative noise](#), *Proc. of SSVM, LNCS*, **6667** (2012), 50–61.
- [31] R. F. Wagner, S. W. Smith, J. M. Sandrik and H. Lopez, [Statistics of speckle in ultrasound B-scans](#), *IEEE Trans. Sonics and Ultrason.*, **30** (1983), 156–163.
- [32] S. Yun and H. Woo, [A new multiplicative denoising variational model based on m-th root transformation](#), *IEEE Trans. Image Process.*, **21** (2012), 2523–2533.

Received April 2016; revised August 2017.

E-mail address: jianlu@szu.edu.cn;lshen03@syr.edu

E-mail address: chenxuszu@sina.com;y1xu@odu.edu



FIGURE 5. Results of various denoising methods on “Cameraman” image corrupted by multiplicative noise with $L = 2$ (the first column) and $L = 4$ (the second column). From top to bottom: Noisy images (8.63 dB, 11.64 dB), DZ (24.27 dB, 25.85 dB), TwL-4V (25.28 dB, 26.72dB), I-DIV (24.98 dB, 26.52 dB), HMNZ (25.30 dB, 27.33 dB), and Ours (25.47 dB, 27.38 dB).

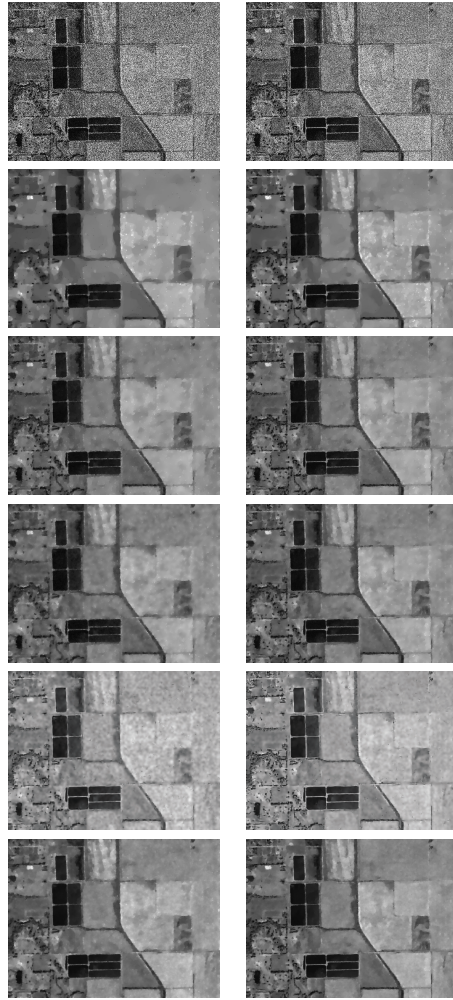


FIGURE 6. Results of various denoising methods on “Remote1” image corrupted by multiplicative noise with $L = 2$ (the first column) and $L = 4$ (the second column). From top to bottom: Noisy images (9.28 dB, 12.28 dB), DZ (21.80 dB, 23.01 dB), TwL-4V (22.57 dB, 23.68 dB), I-DIV (22.39 dB, 23.48 dB), HMNZ (22.63 dB, 24.03 dB), and Ours (22.76 dB, 23.98 dB).