Mathematics

Research article

Group sparse representation and saturation-value total variation based color image denoising under multiplicative noise

Miyoun Jung[*]

Department of Mathematics, Hankuk University of Foreign Studies, Yongin, 17035, Korea

* **Correspondence:** Email: mjung@hufs.ac.kr; Tel: +820313304261.

Abstract: In this article, we propose a novel group-based sparse representation (GSR) model for restoring color images in the presence of multiplicative noise. This model consists of a convex data-fidelity term, and two regularizations including GSR and saturation-value-based total variation (SVTV). The data-fidelity term is suitable for handling heavy multiplicative noise. GSR enables the retention of textures and details while sufficiently removing noise in smooth regions without producing the staircase artifacts engendered by total variation-based models. Furthermore, we introduce a multi-color channel-based GSR that involves coupling between three color channels. This avoids the generation of color artifacts caused by decoupled color channel-based methods. SVTV further improves the visual quality of restored images by diminishing certain artifacts induced by patch-based methods. To solve the proposed nonconvex model and its subproblem, we exploit the alternating direction method of multipliers, which contributes to an efficient iterative algorithm. Numerical results demonstrate the outstanding performance of the proposed model compared to other existing models regarding visual aspect and image quality evaluation values.

Keywords: color image denoising; multiplicative noise; group-based sparse representation; saturation-value total variation; alternating direction method of multipliers
Mathematics Subject Classification: 68U10, 65K10, 94A08, 49J40

1. Introduction

Images are inevitably deteriorated by noise that is generated during the image acquisition process. Thus, image denoising, which aims to retrieve a clean image by removing noise while conserving important structures and details, is a fundamental issue in image processing. This work focuses on color image denoising in the presence of multiplicative noise. Multiplicative noise usually occurs under coherent image systems such as synthetic aperture radar (SAR) [1], ultrasound imaging [2], and laser imaging [3]. Owing to the coherent nature of these image acquisition procedures, most of the

information in an original image may be lost when corrupted by multiplicative noise. Hence, removing multiplicative noise is a more difficult task than removing additive Gaussian noise.

Let $\Omega \subset \mathbb{R}^2$ be an open and bounded domain with a Lipschitz boundary. The observed image $f : \Omega \to \mathbb{R}$ corrupted by multiplicative noise can be formulated as

$$f = u \cdot \eta, \tag{1.1}$$

where $u : \Omega \to \mathbb{R}$ is the original clean image, and η denotes the multiplicative noise that follows a gamma distribution whose density function [4] is given by

$$g(\eta) = \frac{L^L}{\Gamma(L)} \eta^{L-1} e^{-L\eta}, \quad \text{for } \eta \geq 0, \tag{1.2}$$

where $L > 0$ is an integer representing the noise level, and Γ is the gamma function. Thus, the mean of η is 1, and its standard deviation is $1/\sqrt{L}$. This kind of noise commonly appears in SAR images.

For the removal of multiplicative gamma noise, a variety of filtering-based models [5–9] and variational models [10–22] have been developed. Among them, SAR-BM3D [9] is one of the most widely known filtering-based methods, which integrates nonlocal principles with wavelet representation. Particularly, it uses a similarity measure to collect 3-dimensional (3D) groups of similar image patches and applies wavelet decomposition to compute 3D blocks to achieve denoised images. This algorithm performs well, but is computationally very expensive and produces undesirable artifacts in cases of heavy multiplicative noise. Within a variational framework, Aubert and Aujol (AA) [10] proposed a new model combining a data-fidelity term derived from the gamma distribution and the total variation (TV) regularization. TV [25] has been widely used due to its edge preserving capacity and convexity. Nevertheless, the data-fidelity term of this model is not convex, thus the resulting computed solution may not be optimal from a global perspective. Moreover, this model highly depends on the initialization. To resolve these problems, previous studies [11, 16, 19] have suggested convex data-fidelity terms, by using a logarithmic transformation or adding a penalty term. In particular, Lu et al. [19] developed a convex data-fidelity term that is suitable for removing severe multiplicative noise, which integrates the ideas included in the works of [11, 16]. All the above-mentioned models are proposed for grayscale image restoration. Recently, several studies [23, 24] have been proposed to eliminate multiplicative noise in color images by utilizing the data-fidelity terms introduced in [10, 11]. This work makes use of the data-fidelity term presented in [19] to better deal with heavy multiplicative noise in color images.

For color image restoration, TV regularization has been extended and studied in various works [24, 26–38]. Especially, based on the view of color edge detection, Jia et al. [38] introduced a saturation-value-based TV (SVTV) regularization in the HSV color space. They reported in [38] that SVTV is superior to the existing RGB-based color TV methods in several image processing problems by considerably reducing artificial color values and producing well-preserved details and edges. Wang et al. [23] also adopted SVTV for the removal of multiplicative noise and blur in color images. However, SVTV creates a staircase effect; it produces some staircase shapes in smooth transition regions similar to other TV-based regularizations. To alleviate this shortcoming, higher-order derivatives-based SV regularizers were suggested in [24], which were utilized to remove multiplicative noise in color images. These regularizers benefited from both SV-based regularization and higher-order regularization. Despite the excellent performance of SV-based regularization in color image

restoration, local derivatives-based regularization is limited due to the smearing of details and repeated patterns.

To overcome this drawback of local regularizers, various methods exploiting the nonlocal properties of images have been developed for Gaussian noise problems. First, based on the nonlocal self-similarity property that expresses the global repetitiveness of textures and structures in images, a nonlocal means filter was proposed in [39], which enhanced the local smoothing filters. As an extension, Dabov et al. [40] proposed a denoising algorithm, called BM3D, using block matching and 3D transform domain collaborative filtering, through which SAR-BM3D was developed. In addition, a series of nonlocal regularizations have been developed for inverse problems [41–45]. These nonlocal methods provide superior results in comparison to local methods, but many image details and structures cannot be restored due to inaccurate nonlocal weights. The patch-based sparse representation methods have also attracted much attention. Elad and Aharon [46] suggested a denoising algorithm, called K-SVD, by assuming that each patch of an image is accurately represented by a few elements from a dictionary that is learned from natural images. This also led to a variational model in [15] for removing multiplicative noise, which combines K-SVD and TV regularization. Regardless of the phenomenal results of K-SVD, it ignores self-similarity as each patch is considered independently in dictionary learning and sparse coding. In addition, the dictionary learning process is computationally expensive. To achieve better performance, sparsity and self-similarity were combined in several works [47–49], while low-rank-based approaches were also proposed in [22, 50–52]. The aforementioned sparse representation methods consider an image patch as the basic unit of sparse representation.

Unlike these sparse representation methods, Zhang et al. [53] proposed a group-based sparse representation (GSR) model that uses an image group as its basic processing unit. The GSR model characterizes the local sparsity of patches and nonlocal self-similarity of repeated patterns between patches. This also involves an efficient self-adaptive group dictionary learning method, rather than dictionary learning from images. The model demonstrated excellent performance in image deblurring, inpainting, and compressive sensing. It was also utilized in diverse applications such as image deblocking [54], low-light image enhancement [55], SAR image despeckling [56], and Cauchy noise removal [57]. In several works [58–60], the GSR framework was extended, by integrating it with the low-rank approach or patch-based sparse representation. Despite the outstanding performance of these GSR-based models, for color image processing, they convert the image from the RGB space to the YCbCr space and only extract the Y (luminance) channel. Although the Y channel includes most of the important features of the image, such as edges and textures, the Y channel alone might not be sufficient for image processing when all RGB channels are independently corrupted by noise. Recently, in [61], a GSR model utilizing all channel information was proposed and applied to the color image inpainting problem. In this model, an image group corresponding to each RGB channel is constructed, and the final image group is formed from these three image groups. That is, this approach calculates the patch distance between two grayscale patches in each channel, rather than using the patch distance between two color patches. In our work, we exploit a multi-channel-based GSR approach for the removal of multiplicative noise in color images, but unlike the approach in [61], we compute the patch distance between two color image patches to form a color image group. Moreover, we also adopt the SVTV regularization to mitigate certain artifacts caused by patches.

The main contribution of our work is to propose a GSR-based model for restoring color images deteriorated by multiplicative gamma noise. The proposed GSR approach is based on multi-color

channels, which is different from existing approaches. The model also involves a convex data-fidelity term and SVTV regularization. The proposed model sufficiently denoises color images in smooth regions while maintaining textures, details, and edges, even in the presence of severe multiplicative noise. Moreover, unlike existing methods, it does not generate artificial color values or artifacts. To numerically solve the proposed model, an efficient iterative algorithm is offered based on the alternating direction method of multiplies.

The outline of the paper is as follows. Section 2 reviews several variational models for removing multiplicative gamma noise in grayscale images, as well as SVTV regularization and a GSR approach. In Section 3, we present the proposed model and optimization algorithm. Section 4 presents the experimental results of the proposed model, with comparisons to existing methods. Finally, in Section 5, we summarize and conclude our work with some discussions.

2. Background

In this section, we review several multiplicative noise removal models for grayscale images, in addition to SVTV regularization and a GSR model.

2.1. Variational multiplicative noise removal models

First, based on maximum a posterior estimation and Bayes' rule, Aubert and Aujol (AA) [10] proposed a TV model for denoising images in the presence of multiplicative gamma noise, with the assumption $f > 0$:

$$\min_{u>0} \int_{\Omega} \left(\log u + \frac{f}{u} \right) dx + \mu \int_{\Omega} |\nabla u| \, dx, \tag{2.1}$$

where $\mu > 0$ is a parameter that manages the smoothness of the image u, and $|\nabla u| = \sqrt{(\partial_{x_1} u)^2 + (\partial_{x_2} u)^2}$ with $x = (x_1, x_2)$. TV regularization [25] has been widely utilized owing to its convexity and edge conservation ability. Nevertheless, since the data-fidelity term in model (2.1) is not convex, classical optimization algorithms may not furnish a global optimal solution and are highly reliant on initialization.

To rectify these problems, Shi and Osher (SO) [11] proposed the following convex model using the logarithmic transformation, $w = \log u$:

$$\min_{w} \int_{\Omega} \left(w + f e^{-w} \right) dx + \mu \int_{\Omega} |\nabla w| \, dx. \tag{2.2}$$

Thus, after solving this minimization problem to obtain w, the denoised image u is achieved as $u = e^w$. The denoising performance of this model surpasses that of the AA model (2.1) and its denoising results do not depend on initial guesses.

As another approach to build up a convex data-fidelity term, Dong and Zeng [16] inserted a penalty term to the data-fidelity term in model (2.1):

$$\min_{u>0} \int_{\Omega} \left[\log u + \frac{f}{u} + \alpha \left(\sqrt{\frac{u}{f}} - 1 \right)^2 \right] dx + \mu \int_{\Omega} |\nabla u| \, dx, \tag{2.3}$$

where $\alpha > 0$ is a parameter. The objective function in (2.3) is convex if $\alpha \geq \frac{2\sqrt{6}}{9}$. This model is based on the observation that the value of $\mathbb{E}((\sqrt{u/f} - 1)^2)$ is close to 0 with a large L, and its value is still

rather small even when L is small. However, $\mathbb{E}(\sqrt{u/f})$ is always larger than 1 and is close to 1 when L is large enough, i.e., the noise level is relatively small.

To deal with heavy multiplicative noise, Lu et al. [19] substituted 1 in the penalty term with a parameter $\beta \geq 1$. In addition, they adopted the logarithmic transformation, $w = \log(u)$, since exponent-like models provide better denoising performance than logarithm-like models:

$$\min_{w} \int_{\Omega} \left[w + fe^{-w} + \alpha \left(\sqrt{\frac{e^w}{f}} - \beta \right)^2 \right] dx + \mu \int_{\Omega} |\nabla w| \, dx, \tag{2.4}$$

where β depends on the noise level L. The objective function in (2.4) is strictly convex if $\alpha\beta^4 \leq \frac{4096}{27}$. The numerical results in [19] validated that the data-fitting term in (2.4) is suitable for eliminating heavy multiplicative noise.

2.2. Saturation-value total variation

The HSV color model is a color system similar to human perception. Hue refers to the color portion, saturation refers to the gray level of the color, and value refers to the brightness of the color and varies depending on saturation. The important contours and textures of an image lie primarily in its saturation and brightness components.

For a color image $\boldsymbol{u} = (\boldsymbol{u}_r, \boldsymbol{u}_g, \boldsymbol{u}_b)$, the saturation (S) and value (V) components can be expressed in $\boldsymbol{u}_r, \boldsymbol{u}_g$, and $\boldsymbol{u}_b$ as

$$S(x) = \frac{1}{3}\|\mathbf{C}\boldsymbol{u}(x)^T\|_2, \quad V(x) = \frac{1}{\sqrt{3}}|\boldsymbol{u}_r(x) + \boldsymbol{u}_g(x) + \boldsymbol{u}_b(x)|,$$

$$\text{with} \quad \mathbf{C} = \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}, \quad \boldsymbol{u}(x)^T = \begin{bmatrix} \boldsymbol{u}_r(x) \\ \boldsymbol{u}_g(x) \\ \boldsymbol{u}_b(x) \end{bmatrix}, \tag{2.5}$$

where $\|\cdot\|_2$ is the vector Euclidean norm.

Based on the observation that the saturation and value components contain most of the important edges and textures of an image, Jia et al. [38] proposed a saturation-value total variation (SVTV) for color images as follows:

$$\begin{aligned} \text{SV-TV}(\boldsymbol{u}) \quad = \quad \sup_{(\psi_1,\psi_2)\in\mathcal{K}^2, \psi_3\in\mathcal{K}^1} \Bigg\{ \int_{\Omega} &\frac{1}{\sqrt{2}}(\boldsymbol{u}_r(x) - \boldsymbol{u}_g(x)) \operatorname{div}(\psi_1(x)) \\ &+ \frac{1}{\sqrt{6}}(\boldsymbol{u}_r(x) + \boldsymbol{u}_g(x) - 2\boldsymbol{u}_b(x)) \operatorname{div}(\psi_2(x)) \\ &+ \frac{\gamma}{\sqrt{3}}(\boldsymbol{u}_r(x) + \boldsymbol{u}_g(x) + \boldsymbol{u}_b(x)) \operatorname{div}(\psi_3(x)) \Bigg\} dx, \end{aligned} \tag{2.6}$$

where $\mathcal{K}^n = C^1(\Omega, \mathbb{B}^{2n})$ stands for the set of continuously differentiable and bounded functions from the compact support in Ω to $\mathbb{B}^{2n}$, with $\mathbb{B}^n$ representing the closed united ball with a positive integer n, and the parameter $\gamma > 0$ is the weight of the value component in the SVTV term. The convexity, lower semi-continuity, and compactness properties of SVTV were proven. Assuming $\boldsymbol{u} \in W^{1,2}(\Omega)$, SVTV can be written as

$$\text{SV-TV}(\boldsymbol{u}) = \int_{\Omega} \sqrt{\sum_{j=1,2} |\partial_{x_j}\boldsymbol{u}(x)|_s^2} + \gamma \sqrt{\sum_{j=1,2} |\partial_{x_j}\boldsymbol{u}(x)|_v^2} \, dx, \tag{2.7}$$

where $|\partial_{x_j}\boldsymbol{u}(x)|_s$ and $|\partial_{x_j}\boldsymbol{u}(x)|_v$ $(j = 1, 2)$ are defined as

$$|\partial_{x_j}\boldsymbol{u}(x)|_s = \frac{1}{3}\|\mathbf{C}\partial_{x_j}\boldsymbol{u}(x)^T\|_2, \quad |\partial_{x_j}\boldsymbol{u}(x)|_v = \frac{1}{\sqrt{3}}|\partial_{x_j}\boldsymbol{u}_r(x) + \partial_{x_j}\boldsymbol{u}_g(x) + \partial_{x_j}\boldsymbol{u}_b(x)|, \tag{2.8}$$

with $\partial_{x_j}\boldsymbol{u}(x)^T = (\partial_{x_j}\boldsymbol{u}_r(x), \partial_{x_j}\boldsymbol{u}_g(x), \partial_{x_j}\boldsymbol{u}_b(x))^T$. The experimental results in [38] demonstrate the superior restoration performance of SVTV over existing TV-based methods, by reducing the color artifacts induced by noise and better conserving the details.

2.3. Group-based sparse representation

This subsection discusses the GSR model in [53] proposed for grayscale images.

Given a grayscale image $\boldsymbol{x} \in \mathbb{R}^N$, it is divided into n overlapping patches $\boldsymbol{x}_k$ of size $\sqrt{P} \times \sqrt{P}$ $(k = 1, 2, ..., n)$, with stride s. For each patch $\boldsymbol{x}_k$, c most similar patches are searched in a local window of size $L \times L$, and the selected patches are stacked into a matrix $\boldsymbol{x}_{G_k} = [\boldsymbol{x}_{G_k,1}, \boldsymbol{x}_{G_k,2}, ..., \boldsymbol{x}_{G_k,c}] \in \mathbb{R}^{P \times c}$ as columns. The matrix $\boldsymbol{x}_{G_k}$ is called a group and can be defined by a linear operator $\boldsymbol{R}_{G_k}$ as

$$\boldsymbol{x}_{G_k} = \boldsymbol{R}_{G_k}(\boldsymbol{x}). \tag{2.9}$$

The transpose $\boldsymbol{R}_{G_k}^T$ can convert a group into the k-th position in the image padded with zeros elsewhere.

The GSR model assumes that each group $\boldsymbol{x}_{G_k}$ can be accurately represented by a few atoms of a self-adaptive learning dictionary $\boldsymbol{D}_{G_k} = [\boldsymbol{d}_{G_k,1}, \boldsymbol{d}_{G_k,2}, ..., \boldsymbol{d}_{G_k,m}]$, where each atom $\boldsymbol{d}_{G_k,i} \in \mathbb{R}^{P \times c}$ is a matrix of the same size as the group $\boldsymbol{x}_{G_k}$ and $m = \min(P, c)$. The sparse coding process of each group $\boldsymbol{x}_{G_k}$ over $\boldsymbol{D}_{G_k}$ is to find a sparse vector $\boldsymbol{\alpha}_{G_k} = [\alpha_{G_k,1}, \alpha_{G_k,2}, ..., \alpha_{G_k,m}]$ such that

$$\boldsymbol{x}_{G_k} = \boldsymbol{D}_{G_k}\boldsymbol{\alpha}_{G_k} := \sum_{i=1}^{m} \alpha_{G_k,i}\boldsymbol{d}_{G_k,i}. \tag{2.10}$$

Then, the image $\boldsymbol{x}$ is reconstructed from the sparse codes $\{\boldsymbol{\alpha}_{G_k}\}$ as

$$\boldsymbol{x} = \boldsymbol{D}_G \circ \boldsymbol{\alpha}_G := \sum_{k=1}^{n} \boldsymbol{R}_{G_k}^T(\boldsymbol{D}_{G_k}\boldsymbol{\alpha}_{G_k})./ \sum_{k=1}^{n} \boldsymbol{R}_{G_k}^T(\mathbf{1}_{P \times c}), \tag{2.11}$$

where $\boldsymbol{D}_G$ and $\boldsymbol{\alpha}_G$ denote the concatenation of all $\boldsymbol{D}_{G_k}$ and $\boldsymbol{\alpha}_{G_k}$, respectively, $./$ is the element-wise division of vectors, and $\mathbf{1}_{P \times c} \in \mathbb{R}^{P \times c}$ is a matrix of ones.

Assuming that a degraded image is given by $\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n}$, where $\boldsymbol{H}$ is a linear degradation operator and $\boldsymbol{n}$ is noise, the GSR model for image restoration can be formulated as

$$\hat{\boldsymbol{\alpha}}_G := \arg\min_{\boldsymbol{\alpha}_G} \frac{1}{2}\|\boldsymbol{H}\boldsymbol{D}_G \circ \boldsymbol{\alpha}_G - \boldsymbol{y}\|_2^2 + \lambda\|\boldsymbol{\alpha}_G\|_0, \tag{2.12}$$

where $\lambda > 0$ is a regularization parameter, and $\|\boldsymbol{\alpha}_G\|_0$ indicates the number of non-zero elements in $\boldsymbol{\alpha}_G$. By selecting an appropriate self-adaptive group dictionary, the minimization problem (2.12) can be explicitly solved. Then, the reconstructed image is obtained as $\hat{\boldsymbol{x}} = \boldsymbol{D}_G \circ \hat{\boldsymbol{\alpha}}_G$.

3. Proposed model and algorithm

In this section, we present a GSR model to restore color images degraded by multiplicative gamma noise and an optimization algorithm for solving the proposed model.

3.1. Proposed model

Let us assume that a noisy color image $\boldsymbol{f} \in \mathbb{R}^{3N}$ is given by

$$\boldsymbol{f}_{ch} = \boldsymbol{u}_{ch} \cdot \eta_{ch} > 0, \quad ch = r, g, b, \tag{3.1}$$

where $\boldsymbol{u} \in \mathbb{R}^{3N}$ is the clean image, and $\eta_{ch} \in \mathbb{R}^N$ represents the gamma noise in the ch-channel that follows the distribution in (1.2).

The problem (3.1) is an ill-posed inverse problem, so we formulate a minimization problem to reconstruct $\boldsymbol{u}$ from the given data $\boldsymbol{f}$. First, following the process in (2.1)–(2.4) and by letting $\boldsymbol{w} = \log(\boldsymbol{u})$, we consider the following convex data-fidelity term to handle heavy multiplicative gamma noise:

$$F(\boldsymbol{w}) = \left\langle \boldsymbol{w} + \boldsymbol{f}e^{-\boldsymbol{w}} + \alpha \left(\sqrt{\frac{e^{\boldsymbol{w}}}{\boldsymbol{f}}} - \beta \right)^2, \mathbf{1} \right\rangle, \tag{3.2}$$

where $\langle \cdot, \cdot \rangle$ is the inner product, $\mathbf{1} \in \mathbb{R}^{3N}$, and $\alpha, \beta > 0$ are parameters satisfying $\alpha\beta^4 \leq \frac{4096}{27}$.

Next, we introduce our GSR approach to reconstruct color images corrupted by multiplicative gamma noise. For a color image $\boldsymbol{x} \in \mathbb{R}^{3N}$, we extract a color image patch $\boldsymbol{x}_k = [\boldsymbol{x}_k^1, \boldsymbol{x}_k^2, \boldsymbol{x}_k^3]^T \in \mathbb{R}^{3P}$, where $\boldsymbol{x}_k^{ch}$ is a patch of size $\sqrt{P} \times \sqrt{P}$ in the ch-channel ($ch = 1, 2, 3$). Then the group $\boldsymbol{x}_{G_k}$ consisting of c most similar patches is given by

$$\boldsymbol{x}_{G_k} = \begin{bmatrix} \boldsymbol{x}_{G_{k,1}}^1 & \boldsymbol{x}_{G_{k,2}}^1 & \cdots & \boldsymbol{x}_{G_{k,c}}^1 \\ \boldsymbol{x}_{G_{k,1}}^2 & \boldsymbol{x}_{G_{k,2}}^2 & \cdots & \boldsymbol{x}_{G_{k,c}}^2 \\ \boldsymbol{x}_{G_{k,1}}^3 & \boldsymbol{x}_{G_{k,2}}^3 & \cdots & \boldsymbol{x}_{G_{k,c}}^3 \end{bmatrix} \in \mathbb{R}^{3P \times c}, \tag{3.3}$$

and the self-adaptive learning dictionary $\boldsymbol{D}_{G_k}$ is as follows

$$\boldsymbol{D}_{G_k} = \begin{bmatrix} \boldsymbol{d}_{G_{k,1}} & \boldsymbol{d}_{G_{k,2}} & \cdots & \boldsymbol{d}_{G_{k,m}} \end{bmatrix} \in \mathbb{R}^{(3P \times c) \times m}, \tag{3.4}$$

where $\boldsymbol{d}_{G_{k,i}} = [\boldsymbol{d}_{G_{k,i}}^1, \boldsymbol{d}_{G_{k,i}}^2, \boldsymbol{d}_{G_{k,i}}^3]^T \in \mathbb{R}^{3P \times c}$ ($i = 1, \cdots, m$) with $m = \min(3P, c)$.

To search for similar patches in the image $\boldsymbol{w}$, we first take the logarithmic transformation in (3.1), which yields

$$\hat{\boldsymbol{f}} = \boldsymbol{w} + \hat{\eta}, \tag{3.5}$$

where $\hat{\boldsymbol{f}} = \log(\boldsymbol{f})$, and $\hat{\eta} = \log(\eta)$. In Figure 1, we plot the histograms of noise $\hat{\eta}$ in the RGB or HSV color spaces when the noise level $L = 10, 5$, and 3. The figure shows that the histograms of $\hat{\eta}$ in both color spaces at all noise levels are similar to normal distributions. Especially, the noise distribution in the HSV color space fits better with a normal distribution. Based on this observation, we extract the color patches in the HSV color space and compute the Euclidean distance between them. Specifically, after transforming the RGB-based color image $\boldsymbol{w}$ to the HSV-based image $\tilde{\boldsymbol{w}}$, we extract two color image patches, $\tilde{\boldsymbol{w}}_{k_1}$ and $\tilde{\boldsymbol{w}}_{k_2} \in \mathbb{R}^{3P}$, from the transformed image $\tilde{\boldsymbol{w}}$ and compute the similarity measure between the two patches using the following distance function:

$$dist(\tilde{\boldsymbol{w}}_{k_1}, \tilde{\boldsymbol{w}}_{k_2}) = \|\tilde{\boldsymbol{w}}_{k_1} - \tilde{\boldsymbol{w}}_{k_2}\|_2^2. \tag{3.6}$$

Thus, the group $\tilde{w}_{G_k}$ and dictionary $\tilde{D}_{G_k}$ will be attained in the HSV color space, which is discussed in the subsequent subsection.
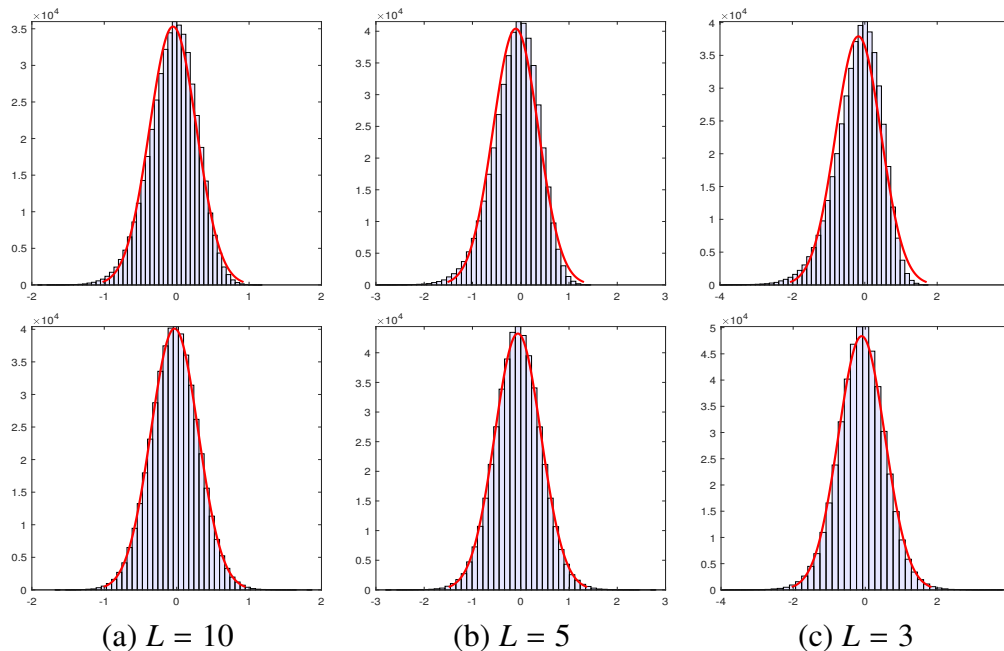

Figure 1. Plots of the histograms of $\hat{\eta} = \hat{f} - w$ in the RGB or HSV color spaces and fitted normal density functions (red curve), when noise levels $L = 10, 5, 3$. Top: $\hat{\eta}$ in the RGB space, Bottom: $\hat{\eta}$ in the HSV space.

Lastly, despite the outstanding restoration performance of the GSR-based approaches, these patch-based methods also cause some ringing or block artifacts in the restored images. To alleviate these artifacts, we also utilize the local regularizer, SVTV, defined in (2.7). Consequently, by combining the data-fidelity term in (3.2) with GSR and SVTV regularization, we propose the following minimization problem for the elimination of multiplicative gamma noise in color images:

$$\min_{w, \alpha_G} F(w) + \lambda \|\alpha_G\|_0 + \mu \, \mathrm{SV\text{-}TV}(w), \quad \text{subject to:} \quad w = D_G \circ \alpha_G, \tag{3.7}$$

where λ and $\mu > 0$ are regularization parameters.

3.2. *Optimization algorithm*

Since problem (3.7) is nonconvex and nonsmooth, it is challenging to solve. Therefore, we utilize the alternating direction method of multipliers (ADMM) [62, 63] that are proposed for solving convex or nonconvex linearly constrained problems.

First, we utilize the variable splitting technique. That is, we introduce an auxiliary variable z to replace w in the SVTV term. Then, problem (3.7) can be transformed into the following constrained problem:

$$\min_{w, \alpha_G, z} F(w) + \lambda \|\alpha_G\|_0 + \mu \, \mathrm{SV\text{-}TV}(z), \quad \text{subject to:} \quad w = D_G \circ \alpha_G, \quad z = w. \tag{3.8}$$

The augmented Lagrangian function (ALF) for problem (3.8) is given by

$$\mathcal{L}_\tau(\boldsymbol{w}, \boldsymbol{\alpha}_G, \boldsymbol{z}; \boldsymbol{p}, \boldsymbol{q}) = F(\boldsymbol{w}) + \lambda\|\boldsymbol{\alpha}_G\|_0 + \mu\, \text{SV-TV}(\boldsymbol{z}) - \langle \boldsymbol{p}, \boldsymbol{w} - \boldsymbol{D}_G \circ \boldsymbol{\alpha}_G \rangle$$
$$+ \frac{\tau}{2}\|\boldsymbol{w} - \boldsymbol{D}_G \circ \boldsymbol{\alpha}_G\|_2^2 - \langle \boldsymbol{q}, \boldsymbol{z} - \boldsymbol{w} \rangle + \frac{\tau}{2}\|\boldsymbol{z} - \boldsymbol{w}\|_2^2, \qquad (3.9)$$

where $\boldsymbol{p} \in \mathbb{R}^{3N}$ and $\boldsymbol{q} \in \mathbb{R}^{3N}$ are the Lagrangian multipliers, and $\tau > 0$ is a penalty parameter.

The ADMM applied to problem (3.8) results in the following iterative algorithm:

$$\begin{cases} (\boldsymbol{\alpha}_G^{\ell+1}, \boldsymbol{z}^{\ell+1}) := \arg\min_{\boldsymbol{\alpha}_G, \boldsymbol{z}} \mathcal{L}_\tau(\boldsymbol{w}^\ell, \boldsymbol{\alpha}_G, \boldsymbol{z}; \boldsymbol{p}^\ell, \boldsymbol{q}^\ell), \\[2mm] \boldsymbol{w}^{\ell+1} := \arg\min_{\boldsymbol{w}} \mathcal{L}_\tau(\boldsymbol{w}, \boldsymbol{\alpha}_G^{\ell+1}, \boldsymbol{z}^{\ell+1}; \boldsymbol{p}^\ell, \boldsymbol{q}^\ell), \\[2mm] \boldsymbol{p}^{\ell+1} = \boldsymbol{p}^\ell - \tau(\boldsymbol{w}^{\ell+1} - \boldsymbol{D}_G \circ \boldsymbol{\alpha}_G^{\ell+1}), \\[2mm] \boldsymbol{q}^{\ell+1} = \boldsymbol{q}^\ell - \tau(\boldsymbol{z}^{\ell+1} - \boldsymbol{w}^{\ell+1}). \end{cases} \qquad (3.10)$$

In the subsequent paragraphs, we solve two subproblems in (3.10).

3.2.1. Solving the $(\boldsymbol{\alpha}_G, \boldsymbol{z})$-subproblem in (3.10)

First, we solve the $(\boldsymbol{\alpha}_G, \boldsymbol{z})$-subproblem in (3.10), which is formulated as

$$\min_{\boldsymbol{\alpha}_G, \boldsymbol{z}} \lambda\|\boldsymbol{\alpha}_G\|_0 + \frac{\tau}{2}\|\boldsymbol{D}_G \circ \boldsymbol{\alpha}_G - \boldsymbol{r}\|_2^2 + \frac{\tau}{2}\|\boldsymbol{z} - \boldsymbol{s}\|_2^2 + \mu\, \text{SV-TV}(\boldsymbol{z}), \qquad (3.11)$$

where $\boldsymbol{r} = \boldsymbol{w}^\ell - \boldsymbol{p}^\ell/\tau$, and $\boldsymbol{s} = \boldsymbol{w}^\ell + \boldsymbol{q}^\ell/\tau$.

As perviously mentioned, we measure the Euclidean distance in the HSV color space to find similar patches. Moreover, to handle the SVTV term defined in the RGB color space, we convert the RGB-based model (3.11) to a HSV-based model using the following orthogonal matrix $\bar{\mathbf{P}}$, that is derived the dual-form of SVTV in (2.6) and has a relation with $\mathbf{C}$ in (2.5) as follows:

$$\bar{\mathbf{P}} = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{6}} & \frac{1}{\sqrt{6}} & -\frac{2}{\sqrt{6}} \\ \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \end{bmatrix}, \qquad \mathbf{C} = \bar{\mathbf{P}}^T A \bar{\mathbf{P}} \quad \text{with} \quad \mathbf{A} = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \qquad (3.12)$$

If we define $\tilde{\boldsymbol{u}}(x) = \bar{\mathbf{P}}\boldsymbol{u}(x)^T$ for $x \in \Omega$, then the S and V components can also be expressed as $S(x) = \|(\tilde{\boldsymbol{u}}_1(x), \tilde{\boldsymbol{u}}_2(x))^T\|_2$ and $V(x) = |\tilde{\boldsymbol{u}}_3(x)|$ respectively. In addition, $|\partial_{x_j}\boldsymbol{u}(x)|_s$ and $|\partial_{x_j}\boldsymbol{u}(x)|_v$ in (2.8) can be expressed as

$$|\partial_{x_j}\boldsymbol{u}(x)|_s = \left\|\frac{1}{3}\bar{\mathbf{P}}^T \mathbf{A}\bar{\mathbf{P}}\bar{\mathbf{P}}^T \partial_{x_j}\tilde{\boldsymbol{u}}(x)^T\right\|_2 = \|(\partial_{x_j}\tilde{\boldsymbol{u}}_1(x), \partial_{x_j}\tilde{\boldsymbol{u}}_2(x))^T\|_2, \quad |\partial_{x_j}\boldsymbol{u}(x)|_v = |\partial_{x_j}\tilde{\boldsymbol{u}}_3(x)|, \qquad (3.13)$$

where the second equality of the first term is because the transpose of an orthogonal matrix is also orthogonal, $\bar{\mathbf{P}}\bar{\mathbf{P}}^T$ is the identity matrix, and the norm of a vector is invariant under multiplication by an orthogonal matrix.

Now, using the following orthogonal matrix

$$\mathbf{P} = \begin{bmatrix} \frac{1}{\sqrt{2}}\mathbf{I} & -\frac{1}{\sqrt{2}}\mathbf{I} & \mathbf{0} \\ \frac{1}{\sqrt{6}}\mathbf{I} & \frac{1}{\sqrt{6}}\mathbf{I} & -\frac{2}{\sqrt{6}}\mathbf{I} \\ \frac{1}{\sqrt{3}}\mathbf{I} & \frac{1}{\sqrt{3}}\mathbf{I} & \frac{1}{\sqrt{3}}\mathbf{I} \end{bmatrix}, \tag{3.14}$$

where $\mathbf{I} \in \mathbb{R}^N$ is the identity matrix, we define $\tilde{\boldsymbol{D}}_G = \mathbf{P}D_G$, $\tilde{\boldsymbol{r}} = \mathbf{P}r$, $\tilde{\boldsymbol{s}} = \mathbf{P}s$, and $\boldsymbol{v} = (\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3)^T = \mathbf{P}z$. Therefore, problem (3.11) can be converted to

$$\min_{\boldsymbol{\alpha}_G, \boldsymbol{v}} \lambda\|\boldsymbol{\alpha}_G\|_0 + \frac{\tau}{2}\|\tilde{\boldsymbol{D}}_G \circ \boldsymbol{\alpha}_G - \tilde{\boldsymbol{r}}\|_2^2 + \frac{\tau}{2}\|\boldsymbol{v} - \tilde{\boldsymbol{s}}\|_2^2 + \mu\left(\|\nabla\bar{\boldsymbol{v}}\|_1 + \gamma\|\nabla\boldsymbol{v}_3\|_1\right), \tag{3.15}$$

where $\bar{\boldsymbol{v}} = (\boldsymbol{v}_1, \boldsymbol{v}_2)^T$ and $\nabla\bar{\boldsymbol{v}} = (\nabla\boldsymbol{v}_1, \nabla\boldsymbol{v}_2)^T$. In this problem, the variables $\boldsymbol{\alpha}_G$ and $\boldsymbol{v}$ are independent of each other, so problem (3.15) can be separated into two problems as

$$\begin{cases} \boldsymbol{\alpha}_G^{\ell+1} := \arg\min_{\boldsymbol{\alpha}_G} \ \frac{\tau}{2}\|\tilde{\boldsymbol{D}}_G \circ \boldsymbol{\alpha}_G - \tilde{\boldsymbol{r}}\|_2^2 + \lambda\|\boldsymbol{\alpha}_G\|_0, \\ \\ \boldsymbol{v}^{\ell+1} := \arg\min_{\boldsymbol{v}} \ \frac{\tau}{2}\|\boldsymbol{v} - \tilde{\boldsymbol{s}}\|_2^2 + \mu\left(\|\nabla\bar{\boldsymbol{v}}\|_1 + \gamma\|\nabla\boldsymbol{v}_3\|_1\right). \end{cases} \tag{3.16}$$

Solving the $\boldsymbol{\alpha}_G$-subproblem in (3.16). The $\boldsymbol{\alpha}_G$-subproblem in (3.16) can be reformulated as

$$\min_{\boldsymbol{\alpha}_G} \ \frac{1}{2}\|\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{r}}\|_2^2 + \frac{\lambda}{\tau}\|\boldsymbol{\alpha}_G\|_0, \tag{3.17}$$

where $\tilde{\boldsymbol{x}} = \tilde{\boldsymbol{D}}_G \circ \boldsymbol{\alpha}_G$. The following proposition indicates that the error term in (3.17) can be well approximated by the sum of the error terms of groups under certain assumptions:

Proposition 1. *Let* $\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{r}} \in \mathbb{R}^{3N}$, $\tilde{\boldsymbol{x}}_{G_k}, \tilde{\boldsymbol{r}}_{G_k} \in \mathbb{R}^{3P\times c}$, *and assume that each element of* $\boldsymbol{e} = \tilde{\boldsymbol{x}} - \tilde{\boldsymbol{r}}$ *is independent and follows a distribution with zero mean and variance* σ^2. *Then, for any* $\varepsilon > 0$, *a relationship between* $\|\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{r}}\|_2^2$ *and* $\sum_{k=1}^n \|\tilde{\boldsymbol{x}}_{G_k} - \tilde{\boldsymbol{r}}_{G_k}\|_2^2$ *holds as follows:*

$$\lim_{\substack{N\to\infty \\ K\to\infty}} \boldsymbol{Pr}\left\{\left|\frac{1}{3N}\|\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{r}}\|_2^2 - \frac{1}{K}\sum_{k=1}^n \|\tilde{\boldsymbol{x}}_{G_k} - \tilde{\boldsymbol{r}}_{G_k}\|_2^2\right| < \varepsilon\right\} = 1,$$

where $\boldsymbol{Pr}(\cdot)$ *represents the probability, and* $K = 3P \times c \times n$.

Proof. It follows from the law of large numbers, see [53]. □

According to Proposition 1 and using $\|\boldsymbol{\alpha}_G\|_0 = \sum_{k=1}^n \|\boldsymbol{\alpha}_{G_k}\|_0$, problem (3.17) is equivalent to

$$\arg\min_{\boldsymbol{\alpha}_G} \sum_{k=1}^n \left(\frac{1}{2}\|\tilde{\boldsymbol{x}}_{G_k} - \tilde{\boldsymbol{r}}_{G_k}\|_2^2 + \tilde{\tau}\|\boldsymbol{\alpha}_{G_k}\|_0\right), \tag{3.18}$$

where $\tilde{\tau} = \frac{\lambda K}{3\tau N}$. This problem can be efficiently minimized by solving n $\boldsymbol{\alpha}_{G_k}$-subproblems, one of which is formulated as

$$\arg\min_{\boldsymbol{\alpha}_{G_k}} \frac{1}{2}\|\tilde{\boldsymbol{x}}_{G_k} - \tilde{\boldsymbol{r}}_{G_k}\|_2^2 + \tilde{\tau}\|\boldsymbol{\alpha}_{G_k}\|_0 = \frac{1}{2}\|\tilde{\boldsymbol{D}}_{G_k}\boldsymbol{\alpha}_{G_k} - \tilde{\boldsymbol{r}}_{G_k}\|_2^2 + \tilde{\tau}\|\boldsymbol{\alpha}_{G_k}\|_0, \tag{3.19}$$

where $\tilde{\boldsymbol{D}}_{G_k}$ is the self-adaptive learned dictionary from $\tilde{\boldsymbol{r}}_{G_k}$ as follows.

First, the singular value decomposition is applied to $\tilde{\boldsymbol{r}}_{G_k}$:

$$\tilde{\boldsymbol{r}}_{G_k} = \boldsymbol{U}_{G_k}\boldsymbol{\Sigma}_{G_k}\boldsymbol{V}_{G_k}^T = \sum_{i=1}^{m} \boldsymbol{\gamma}_{\tilde{r}_{G_k},i}(\boldsymbol{u}_{G_k,i}\boldsymbol{v}_{G_k,i}^T), \tag{3.20}$$

where $\boldsymbol{\Sigma}_{G_k} = \mathrm{diag}(\boldsymbol{\gamma}_{\tilde{r}_{G_k}})$ is a diagonal matrix with the elements of $\boldsymbol{\gamma}_{\tilde{r}_{G_k}} = [\boldsymbol{\gamma}_{\tilde{r}_{G_k},1}, \boldsymbol{\gamma}_{\tilde{r}_{G_k},2}, ..., \boldsymbol{\gamma}_{\tilde{r}_{G_k},m}]$ on its main diagonal, and $\boldsymbol{u}_{G_k,i}$ and $\boldsymbol{v}_{G_k,i}$ are the columns of $\boldsymbol{U}_{G_k}$ and $\boldsymbol{V}_{G_k}$, respectively. Then, the atoms in $\tilde{\boldsymbol{D}}_{G_k}$ is defined as $\tilde{\boldsymbol{d}}_{G_k,i} = \boldsymbol{u}_{G_k,i}\boldsymbol{v}_{G_k,i}^T$, thus the dictionary $\tilde{\boldsymbol{D}}_{G_k}$ for $\tilde{\boldsymbol{x}}_{G_k}$ is given by

$$\tilde{\boldsymbol{D}}_{G_k} = [\tilde{\boldsymbol{d}}_{G_k,1}, \tilde{\boldsymbol{d}}_{G_k,2}, ..., \tilde{\boldsymbol{d}}_{G_k,m}]. \tag{3.21}$$

Using $\tilde{\boldsymbol{r}}_{G_k} = \tilde{\boldsymbol{D}}_{G_k}\boldsymbol{\gamma}_{\tilde{r}_{G_k}}$ and due to the unitary property of $\tilde{\boldsymbol{D}}_{G_k}$, the $\boldsymbol{\alpha}_{G_k}$-subproblem in (3.19) is equivalent to

$$\arg\min_{\boldsymbol{\alpha}_{G_k}} \frac{1}{2}\|\boldsymbol{\alpha}_{G_k} - \boldsymbol{\gamma}_{\tilde{r}_{G_k}}\|_2^2 + \tilde{\tau}\|\boldsymbol{\alpha}_{G_k}\|_0. \tag{3.22}$$

This problem has a closed-form solution as

$$\hat{\boldsymbol{\alpha}}_{G_k} = hard(\boldsymbol{\gamma}_{\tilde{r}_{G_k}}, \sqrt{2\tilde{\tau}}) = \boldsymbol{\gamma}_{\tilde{r}_{G_k}} \odot \mathbf{1}(|\boldsymbol{\gamma}_{\tilde{r}_{G_k}}| - \sqrt{2\tilde{\tau}}), \tag{3.23}$$

where $hard(\cdot)$ represents the hard thresholding operator, $\odot$ represents the element-wise multiplication, and $\mathbf{1}(y)$ is the indicator function that has a value of 1 if $y > 0$ and 0 otherwise.

Solving the $\boldsymbol{v}$-subproblem in (3.16). Next, we solve the $\boldsymbol{v}$-subproblem in (3.16). To solve this convex problem, we again adopt the ADMM in [62]. First, to handle the non-differentiable terms, we introduce two auxiliary variables $\boldsymbol{d}_1$ and $\boldsymbol{d}_2$ to substitute $\nabla\bar{\boldsymbol{v}}$ and $\nabla\boldsymbol{v}_3$, respectively. Then, the unconstrained $\boldsymbol{v}$-problem in (3.16) is converted into the following constrained minimization problem:

$$\min_{v,d_1,d_2} \frac{\tau}{2}\|\boldsymbol{v} - \tilde{\boldsymbol{s}}\|_2^2 + \mu\|\boldsymbol{d}_1\|_1 + \mu\gamma\|\boldsymbol{d}_2\|_1, \tag{3.24}$$
$$\text{subject to: } \boldsymbol{d}_1 = \nabla\bar{\boldsymbol{v}}, \quad \boldsymbol{d}_2 = \nabla\boldsymbol{v}_3.$$

The ALF of problem (3.24) is given by

$$\mathcal{L}_\xi(\boldsymbol{v}, \boldsymbol{d}_1, \boldsymbol{d}_2; \boldsymbol{b}_1, \boldsymbol{b}_2) = \frac{\tau}{2}\|\boldsymbol{v} - \tilde{\boldsymbol{s}}\|_2^2 + \mu\|\boldsymbol{d}_1\|_1 + \mu\gamma\|\boldsymbol{d}_2\|_1 - \langle\boldsymbol{b}_1, \boldsymbol{d}_1 - \nabla\bar{\boldsymbol{v}}\rangle$$
$$+ \frac{\xi}{2}\|\boldsymbol{d}_1 - \nabla\bar{\boldsymbol{v}}\|_2^2 - \langle\boldsymbol{b}_2, \boldsymbol{d}_2 - \nabla\boldsymbol{v}_3\rangle + \frac{\xi}{2}\|\boldsymbol{d}_2 - \nabla\boldsymbol{v}_3\|_2^2, \tag{3.25}$$

where $\boldsymbol{b}_1 \in \mathbb{R}^{3N\times4}$ and $\boldsymbol{b}_2 \in \mathbb{R}^{3N\times2}$ are the Lagrangian multipliers, and $\xi > 0$ is a penalty parameter.

The ADMM applied to problem (3.24) brings the following iterative algorithm:

$$\begin{cases} \boldsymbol{v}^{t+1} := \arg\min_v \ \mathcal{L}_\xi(\boldsymbol{v}, \boldsymbol{d}_1^t, \boldsymbol{d}_2^t, \boldsymbol{b}_1^t, \boldsymbol{b}_2^t), \\[2mm] (\boldsymbol{d}_1^{t+1}, \boldsymbol{d}_2^{t+1}) := \arg\min_{d_1,d_2} \ \mathcal{L}_\xi(\boldsymbol{v}^{t+1}, \boldsymbol{d}_1, \boldsymbol{d}_2, \boldsymbol{b}_1^t, \boldsymbol{b}_2^t), \\[2mm] \boldsymbol{b}_1^{t+1} = \boldsymbol{b}_1^t - \xi(\boldsymbol{d}_1^{t+1} - \nabla\bar{\boldsymbol{v}}^{t+1}), \\[2mm] \boldsymbol{b}_2^{t+1} = \boldsymbol{b}_2^t - \xi(\boldsymbol{d}_2^{t+1} - \nabla{\boldsymbol{v}_3}^{t+1}). \end{cases} \tag{3.26}$$

The constraint in (3.24) can be written as $(d_1, d_2)^T = \mathbf{B}(\bar{v}, v_3)^T$ with $\mathbf{B} = \begin{bmatrix} \nabla & 0 \\ 0 & \nabla \end{bmatrix}$. Since the matrix $\mathbf{B}$ has a full column-rank, we can obtain the following convergence results, according to Theorem 1 in [64]: *If the sequence $\{(v^t, d_1^t, d_2^t, b_1^t, b_2^t)\}$ is generated by ADMM in (3.26), then this sequence has a single limit point $(v^*, d_1^*, d_2^*, b_1^*, b_2^*)$, and this limit point is a saddle point of the augmented Lagrangian $\mathcal{L}_\xi$. That is, $\{(v^t, d_1^t, d_2^t)\}$ has a single limit point (v^*, d_1^*, d_2^*), and this limit point solves problem (3.24).*

The v-subproblem in (3.26) can be rewritten as

$$\min_v \frac{\tau}{2}\|v - \tilde{s}\|_2^2 + \frac{\xi}{2}\|d_1^t - \nabla\bar{v} - b_1^t/\xi\|_2^2 + \frac{\xi}{2}\|d_2^t - \nabla v_3 - b_2^t/\xi\|_2^2. \tag{3.27}$$

From the first-order optimality condition, v^{t+1} must satisfy the following normal equations:

$$\begin{aligned}
(\tau + \xi\nabla^T\nabla)v_i &= \tau s_i + \xi\nabla^T(d_{1,i}^t - b_{1,i}^t/\xi), \quad i = 1, 2, \\
(\tau + \xi\nabla^T\nabla)v_3 &= \tau s_3 + \xi\nabla^T(d_2^t - b_2^t/\xi),
\end{aligned} \tag{3.28}$$

where $\nabla^T = -\text{div}$ with div being the discrete divergence operator defined as $\text{div}(s_1, s_2) = \partial_{x_1}s_1 + \partial_{x_2}s_2$. Assuming the periodic boundary condition, the above equations can be explicitly solved using the 2-dimensional discrete Fourier transform (DFT).

The (d_1, d_2)-subproblem in (3.26) can be separated into two subproblems for each variable since the variables are independent of each other. Specifically, the d_i-subproblems can be written as

$$\begin{aligned}
d_1^{t+1} &:= \arg\min_{d_1} \mu\|d_1\|_1 + \frac{\xi}{2}\|d_1 - (\nabla\bar{v}^{t+1} + b_1^t/\xi)\|_2^2, \\
d_2^{t+1} &:= \arg\min_{d_2} \mu\gamma\|d_2\|_1 + \frac{\xi}{2}\|d_2 - (\nabla v_3^{t+1} + b_2^t/\xi)\|_2^2.
\end{aligned} \tag{3.29}$$

Then, the solutions d_i^{t+1} can be obtained as follows

$$\begin{aligned}
d_1^{t+1} &= shrink(\nabla\bar{v}^{t+1} + b_1^t/\xi, \mu/\xi), \\
d_2^{t+1} &= shrink(\nabla v_3^{t+1} + b_2^t/\xi, \mu\gamma/\xi),
\end{aligned} \tag{3.30}$$

where $shrink(\cdot)$ is the soft-thresholding operator defined as

$$shrink(\mathbf{a}, b) = \frac{\mathbf{a}}{|\mathbf{a}|} \cdot \max(|\mathbf{a}| - b, 0). \tag{3.31}$$

3.2.2. Solving the w-subproblem in (3.10)

Now, we solve the w-subproblem in (3.10), which can be written as

$$\min_w F(w) + \frac{\tau}{2}\|w - x^{\ell+1} - p^\ell/\tau\|_2^2 + \frac{\tau}{2}\|z^{\ell+1} - w - q^\ell/\tau\|_2^2, \tag{3.32}$$

where $x^{\ell+1} = \mathbf{P}^T\tilde{x}^{\ell+1}$ with $\tilde{x}^{\ell+1} = (\tilde{D}_G \circ \hat{\alpha}_G)^{\ell+1}$, and $z^{\ell+1} = \mathbf{P}^T v^{\ell+1}$. This minimization problem is convex, but it is difficult to obtain an explicit form of solution due to the complex data-fidelity term. Fortunately, the problem can be efficiently solved by using Newton's method as follows:

$$w^{t+1} = w^t - \frac{\hat{F}'(w^t) + \tau(w^t - x^{\ell+1} - p^\ell/\tau) + \tau(w^t + q^\ell/\tau - z^{\ell+1})}{\hat{F}''(w^t) + 2\tau}, \tag{3.33}$$

where $\hat{F}'(w)$ and $\hat{F}''(w)$ are given by

$$
\begin{aligned}
\hat{F}'(w) &= 1 - fe^{-w} + \alpha\left(\frac{e^w}{f} - \beta\sqrt{\frac{e^w}{f}}\right), \\
\hat{F}''(w) &= fe^{-w} + \alpha\left(\frac{e^w}{f} - \frac{\beta}{2}\sqrt{\frac{e^w}{f}}\right).
\end{aligned}
\tag{3.34}
$$

In practice, only a few iteration numbers are required for convergence.

Consequently, the algorithm for solving model (3.7) is summarized in Algorithm 1.

Algorithm 1 Solving the proposed model (3.7)

1: **Input**: choose the parameters $\alpha, \beta, \lambda, \mu, \gamma, \tau, \xi > 0$, patch size P, similar patch number c, window size L, stride s, maximum iteration numbers M_{out}, M_v, M_w, tolerance values tol_{out}, tol_v, tol_w.
2: **Initialization**: set $w^0 = \log(f)$, $p^0 = q^0 = \mathbf{0}$, $b_i^0 = \mathbf{0}$ $(i = 1, 2)$.
3: **repeat**
4:     **Let $\tilde{r} = P(w^\ell - p^\ell/\tau)$ and $\tilde{s} = P(w^\ell + q^\ell/\tau)$.**
5:     **Compute $\tilde{x}^{\ell+1} = (\tilde{D}_G \circ \hat{\alpha}_G)^{\ell+1}$ using $\tilde{D}_{G_k}$ in (3.21) and $\hat{\alpha}_{G_k}$ in (3.23).**
6:     **Compute $v^{\ell+1}$ by iterating for $t = 0, 1, 2, \cdots, M_v$:**
7:         compute v^{t+1} by solving (3.28) using DFT,
8:         compute d_i^{t+1} $(i = 1, 2)$ as in (3.30),
9:         compute b_i^{t+1} $(i = 1, 2)$ as in (3.26),
10:    **Let $x^{\ell+1} = P^T\tilde{x}^{\ell+1}$ and $z^{\ell+1} = P^T v^{\ell+1}$.**
11:    **Compute $w^{\ell+1}$ by iterating for $t = 0, 1, 2, \cdots, M_w$:**
12:         $w^{t+1} = w^t - \frac{\hat{F}'(w^t) + \tau(w^t - x^{\ell+1} - p^\ell/\tau) + \tau(w^t + q^\ell/\tau - z^{\ell+1})}{\hat{F}''(w^t) + 2\tau}$, with $\hat{F}'$ and $\hat{F}''$ in (3.34),
13:    $p^{\ell+1} = p^\ell - \tau(w^{\ell+1} - x^{\ell+1})$,
14:    $q^{\ell+1} = q^\ell - \tau(z^{\ell+1} - w^{\ell+1})$,
15: **until** a stopping criterion is satisfied.
16: **Output**: restored image $u = e^w$.

---

4. Experimental results

This section presents the numerical results of the proposed model, with comparisons to other existing models including L2-SVTV in [38] and AA-SVTV in [23], that combine SVTV with the L^2 data-fidelity term or the data-fidelity term in (2.1), in addition to SO-SVNTVTV2 in [24] that integrates the data-fidelity term in (2.2) with a SV-based nonconvex higher-order regularization. We also compare our model with SAR-BM3D which is applied to each RGB channel. For a fair comparison, we adopt the variable splitting approach and ADMM for solving the aforementioned existing variational models. All numerical experiments were implemented using MATLAB R2023a on a 64-bit Windows 10 desktop PC with an Intel CPU at 4.2GHz and 64GB RAM. All numerical results can be found in the material at the following link: `https://url.kr/svqunr`.

4.1. Implementation setting

We test each model on 16 natural images and five SAR images, provided in Figure 2. The range of intensity values in the original images is assumed to be [0, 255]. In the experiments, all test images are corrupted by multiplicative gamma noise with noise levels of $L = 10, 5$, or 3.

| (a) Airplane | (b) Baboon | (c) Barbara | (d) Bears | (e) Boat | (f) Building | (g) Butterfly |

| (h) Castle | (i) Cathedral | (j) Flower | (k) Hill | (l) House | (m) Lena | (n) Peppers |

| (o) Policemen | (p) Statues | (q) Remote1 | (r) Remote2 | (s) Remote3 | (t) Remote4 | (u) Remote5 |

Figure 2. Original test images. (a)-(c), (e), (l)-(n): $N = 256 \times 256$, (f), (h), (i), (o), (p): $N = 481 \times 321$, (d), (j): $N = 321 \times 481$, (g): 201×266, (k): 278×350, (q)-(u): $N = 512 \times 512$.

To evaluate the effectiveness of the models, we calculate the peak-signal-to-noise-ratio (PSNR) value defined as

$$\mathrm{PSNR}(\boldsymbol{u}, \boldsymbol{u}_*) = 10 \log_{10}\left(\frac{3N \times 255^2}{\|\boldsymbol{u} - \boldsymbol{u}_*\|_2^2}\right),$$

where $\boldsymbol{u}$ and $\boldsymbol{u}_*$ denote the restored and original images, respectively, and N is the image size. We also compute the structural similarity (SSIM) index [65], which is a perception-based measure that considers the structure of the objects in the visual scene. Specifically, we compute the average of the SSIM values calculated from each RGB channel. For all models, we adjust the parameters to achieve the optimal restored images, taking into account both the visual quality and the PSNR and SSIM values.

The stopping criterion for the proposed model is as follows:

$$\frac{\|\mathbf{h}^{iter} - \mathbf{h}^{iter-1}\|_2}{\|\mathbf{h}^{iter}\|_2} \leq tol \quad \text{or} \quad k > M,$$

where $\mathbf{h} = \boldsymbol{w}$ or $\boldsymbol{v}$, and $tol > 0$ and $M > 0$ are the given tolerance and maximum iteration number, respectively. We set $tol = 2 \times 10^{-3}$ and $M = 30$ for the outer iteration, and $tol = 10^{-3}$ and $M = 10$ for the $\boldsymbol{v}$-subproblem and Newton's method for $\boldsymbol{w}$.

The parameter settings of the proposed model are as follows. First, in the data-fidelity term, parameter α, which controls the effect of the penalty term, is fixed at 0.1. The value of β inside the penalty term is closely related to the noise level L and is chosen as $\frac{1}{0.1113+0.1109L^2} + 1$ as suggested in [19]. The parameter γ in the SVTV term, which controls the weight of the value component, is set to 0.6, while parameter μ, which controls the effect of SVTV regularization, is fixed at 0.05 throughout

the experiment. For the GSR representation, the image patch size, $\sqrt{P} \times \sqrt{P}$, is set as 6×6 for natural images and 5×5 for SAR images. The number of similar patches, c, is set as 80, so the size of a group is 108×80 for natural images and 75×80 for SAR images. The size of the search window is 20×20. The overlapping width between adjacent patches is $s = 4$ pixels. The parameters τ and λ also affect the GSR representation; the larger the τ or λ, the smoother the restored image becomes. However, the larger the τ, the longer the ADMM algorithm takes, so we fix parameter τ as 3. The parameter λ mainly affects the quality of restored images, so the optimal values of λ are chosen differently depending on the noise levels and images. For instance, $\lambda \in \{0.08, 0.1, 0.11\}$ when $L = 10$, $\lambda \in \{0.13, 0.15, 0.17\}$ when $L = 5$, and $\lambda \in \{0.22, 0.24, 0.26\}$ when $L = 3$. The specific value of λ is given in each figure. The ADMM parameter ξ is fixed at 1. For other variational models, we set $\gamma = 0.4$ for L2-SVTV and $\gamma = 0.6$ for AA-SVTV and SO-SVNTVTV2. Specific values of the regularization parameters for all variational models are provided in each figure in the supplementary file at the above link.

4.2. Denoising results for natural color images

In Figures 3–6, we present the denoising results of the proposed model tested on natural color images, with comparisons to other models.

First, Figure 3 presents the denoising results tested on the Hill image with different noise levels including $L = 10$, 5, and 3. The results show that L2-SVTV does not sufficiently remove the noise in some bright areas, such as the sky and white wall areas. This results in erroneous color values in those areas and makes the entire image dark. It also smooths out dark regions such as the roof and right tree areas, which can be clearly seen in the zoomed-in image in Figure 4. Contrarily, AA-SVTV adequately smooths bright regions while retaining noise in the dark regions. In addition, it induces certain artifacts such as black dots, especially in the dark regions. This becomes worse as the noise level increases, as shown in the zoomed-in image in Figure 4. Moreover, as the noise level increases, the entire color tone of the restored image of AA-SVTV is inclined to be distorted. For instance, the color tone of the red roof is degraded compared to those of SO-SVNTVTV2 and our model. SO-SVNTVTV2 suitably eliminates noise in both bright and dark regions, with fewer faulty color values or artifacts than L2-SVTV and AA-SVTV. These indicate the advantages of the SO data-fidelity term when removing multiplicative gamma noise. Additionally, SO-SVNTVTV2 alleviates the staircase effect caused by TV-based models. However, our model improves SO-SVNTVTV2 by preserving more details and generating more naturally restored images with no staircase artifacts and fewer artificial colors. This also leads to the higher PSNR and SSIM values of our model. In contrast, SAR-BM3D also sufficiently conserves edges and fine features, but it produces artificial color values and undesirable artifacts near edges, which deteriorates the denoising performance of SAR-BM3D on color images. These findings demonstrate the effectiveness of the proposed model for restoring color images polluted by multiplicative gamma noise.

(a) 17.06/0.4076   (b) 25.64/0.6292   (c) 26.83/0.7389   (d) 27.05/0.7323   (e) 26.96/0.7249   (f) 27.62/0.7401

(a) 14.04/0.2879   (b) 24.43/0.5644   (c) 25.31/0.6597   (d) 25.76/0.6658   (e) 25.63/0.6557   (f) 26.67/0.7059

(a) 11.83/0.2107   (b) 23.90/0.5335   (c) 24.08/0.5894   (d) 24.91/0.6182   (e) 24.65/0.5982   (f) 25.54/0.6374

Figure 3. Denoising results with different noise levels: $L = 10$ (top), 5 (middle), 3 (bottom). (a) Data f, (b) L2-SVTV, (c) AA-SVTV, (d) SO-SVNTVTV2, (e) SAR-BM3D, (f) Proposed. Parameter of (f): (top to bottom) $\lambda = 0.1, 0.13, 0.24$. PSNR/SSIM values are presented.

L2-SVTV   AA-SVTV   SO-SVNTVTV2   SAR-BM3D   Proposed

Figure 4. Zoomed-in portions from the images in Figure 3 when $L = 5$.

In Figure 5, we present the denoising results of all models when $L = 5$ and 3. Similar to the previous results, L2-SVTV oversmooths the dark regions, such as the hair and textured background in the Barbara image, the stems of flowers in the Building image and the body of statue in the Statues image. It also yields improper color values in some bright regions. This can be observed in the scarf in the Barbara image, the wall in the Building image and the sky area in the Statues image. AA-SVTV retains some noise or black dots in the aforementioned dark regions. This can also be seen in the sky area in the Building image, as shown in the zoomed-in image in Figure 6. As the noise level increases, the denoising performance of L2-SVTV and AA-SVTV significantly deteriorates. L2-SVTV generates the degraded color tones of restored images, while AA-SVTV supplies the black dots in the entire image region, as observed in the Peppers image. SO-SVNTVTV2 not only properly denoises the image in both the dark and bright regions, but also mitigates the staircase artifacts that appeared in the restored images from L2-SVTV and AA-SVTV. However, SO-SVNTVTV2 fails to preserve fine textures and details, as seen in the scarf in the Barbara image and the window areas in the Building image. In contrast, these textural regions are well conserved in SAR-BM3D and our model. This leads

to the higher PSNR values of SAR-BM3D and our model than SO-SVNTVTV2 in the Barbara and Building images. Nevertheless, SAR-BM3D produces faulty color values, especially near edges, while SO-SVNTVTV2 and our model do not. Especially when $L = 3$, SAR-BM3D produces artificial color values in the entire image region and certain artifacts induced by patches. These can be more obviously seen in the entire image of Peppers and the zoomed-in image of Statues in Figure 6. Consequently, our model provides the cleanest and clearest restored images among the models, while retaining textures, details, and sharp edges. This also corresponds to the highest PSNR and SSIM values. These examples also confirm the superior denoising performance of the proposed model over existing models, even in the presence of heavy multiplicative gamma noise.
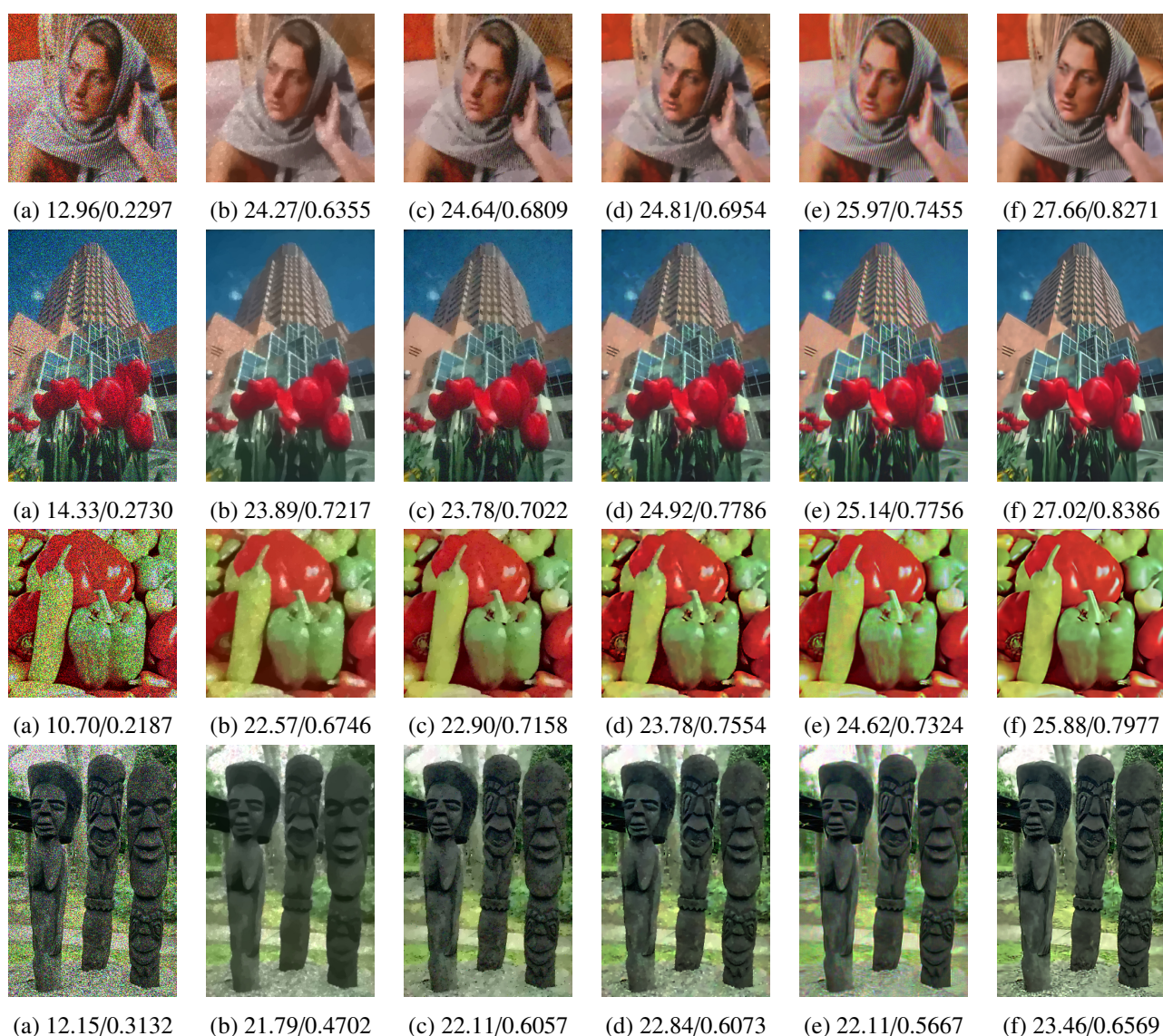

(a) 12.96/0.2297	(b) 24.27/0.6355	(c) 24.64/0.6809	(d) 24.81/0.6954	(e) 25.97/0.7455	(f) 27.66/0.8271
(a) 14.33/0.2730	(b) 23.89/0.7217	(c) 23.78/0.7022	(d) 24.92/0.7786	(e) 25.14/0.7756	(f) 27.02/0.8386
(a) 10.70/0.2187	(b) 22.57/0.6746	(c) 22.90/0.7158	(d) 23.78/0.7554	(e) 24.62/0.7324	(f) 25.88/0.7977
(a) 12.15/0.3132	(b) 21.79/0.4702	(c) 22.11/0.6057	(d) 22.84/0.6073	(e) 22.11/0.5667	(f) 23.46/0.6569

Figure 5. Denoising results when $L = 5$ (1st and 2nd rows) and $L = 3$ (3rd and 4th rows). (a) Data f, (b) L2-SVTV, (c) AA-SVTV, (d) SO-SVNTVTV2, (e) SAR-BM3D, (f) Proposed. Parameter of (f): (top to bottom) $\lambda = 0.15, 0.15, 0.26, 0.22$. PSNR/SSIM values are presented.
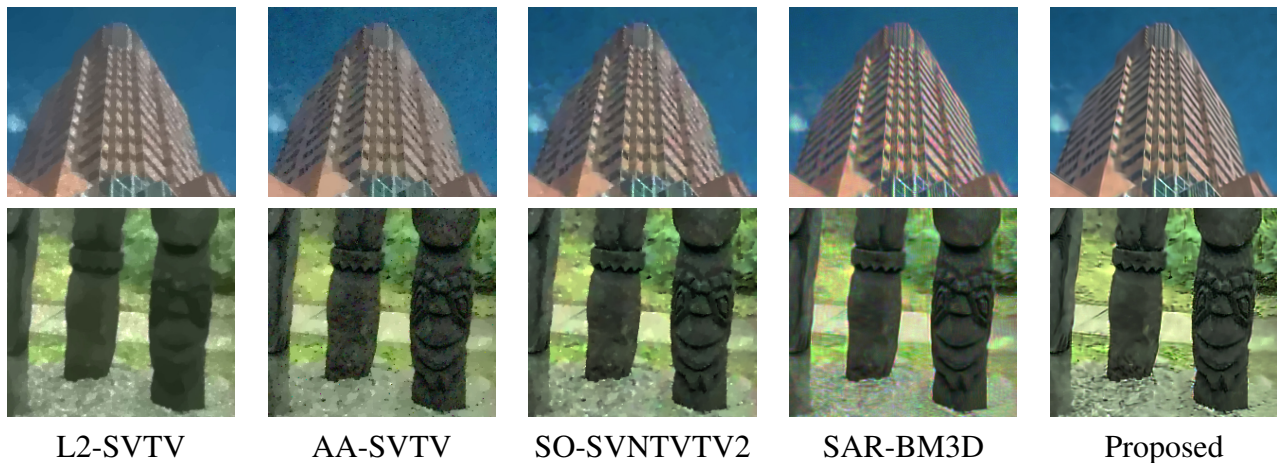
Figure 6. Zoomed-in portions from the images in Figure 5.

In Figure 7, we show the effect of our multi-channel-based GSR approach by using the proposed model (3.7) with two different GSR approaches. Specifically, our GSR approach selects similar color image patches by computing the distance between two color patches and constructing an image group using these selected color patches. The other GSR approach chooses similar image patches in each color channel by measuring the patch distance in each channel. For comparable results, we use $P = 8 \times 8$ and $c = 80$ for this decoupled GSR approach. In the Barbara image, the decoupled GSR approach slightly better preserves textures such as the scarf parts, which leads to higher SSIM values than our coupled GSR approach. However, the decoupled GSR approach furnishes artificial color values in all the examples, unlike our GSR approach. Thus, our GSR approach provides higher PSNR and SSIM values in most examples. These demonstrate the efficacy of our GSR approach for the removal of multiplicative noise in color images.

Figure 8 displays the effect of the data-fidelity term in the proposed model. In particular, we show the effect of the penalty term, $\left(\sqrt{e^w/f} - \beta \right)^2$, included in the data-fidelity term, since the effectiveness of the SO data-fidelity term has already been shown in the previous examples compared to the L^2 or AA data-fidelity terms. That is, we compare our model with the model (3.7) without this penalty term. Although the restored images look almost identical, the use of the penalty term produces reconstructed images with higher PSNR and SSIM values. Therefore, these results validate the superiority of our data-fidelity term, compared to the SO data-fidelity term. For more examples, please refer to the supplementary file at the above link.

Figure 9 illustrates the effect of the SVTV term in the proposed model. Specifically, we compare our model with the model (3.7) without the SVTV term. The GSR model (3.7) without the SVTV term generates certain ringing artifacts near edges and block artifacts in homogeneous regions, which are usually induced by patch-based methods. However, the SVTV term alleviates or diminishes these artifacts by sufficiently smoothing these areas, as shown in the images in the bottom row. These show the effectiveness of the SVTV term in the proposed model.

Figure 7. Effect of our multi-channel GSR approach ($L = 3$). (a) Model (3.7) with GSR applied to each HSV channel, (b) Proposed model. PSNR/SSIM values are presented.

Figure 8. Effect of the quadratic penalty term, $\left(\sqrt{e^w / f} - \beta \right)^2$, of the data-fidelity term in the proposed model ($L = 3$). (a) Model (3.7) without the quadratic penalty term, (b) Proposed model. PSNR/SSIM are presented.
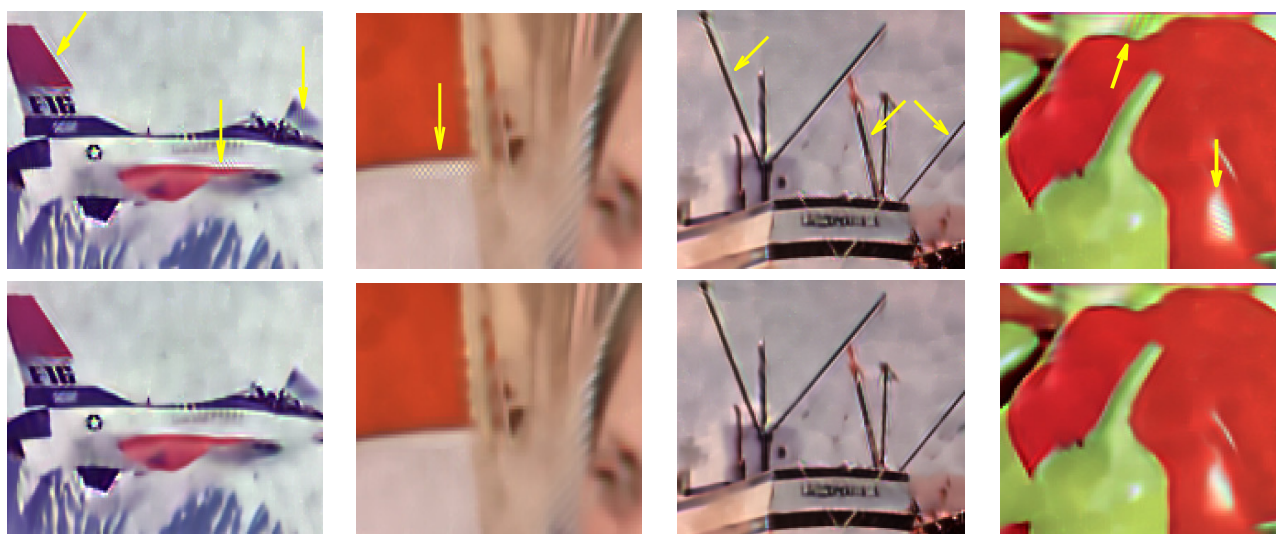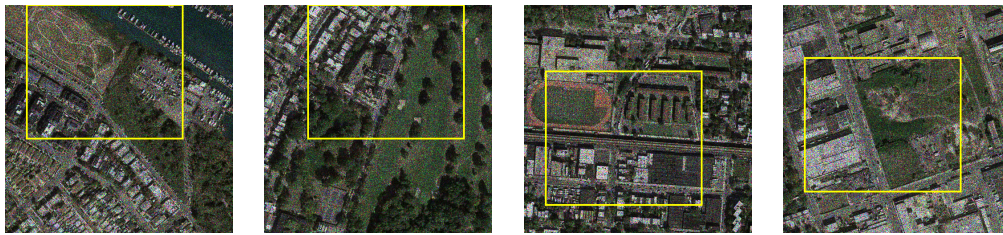
Figure 9. Effect of the SVTV term in the proposed model. (Top) Model (3.7) without SVTV term, (Bottom) Proposed model.

4.3. Denoising results for color SAR images

Figure 10 presents the denoising results of all models tested on real SAR images, when $L = 3$. The SAR images consist of many dark regions such as trees and shadows, while including some bright regions such as buildings and ships. Similar to the natural color image cases, L2-SVTV smooths out dark areas but does not adequately remove noise in bright areas, leading to inappropriate color values. In addition, the color tone of the restored image of L2-SVTV is significantly degraded compared to other models. In contrast, AA-SVTV sufficiently denoises the bright regions but supplies some black artifacts in the homogeneous regions including the grass or ground, and dark regions such as building shadows and tree areas. On the other hand, the denoising performance of SAR-BM3D on color SAR images is not as good as on natural color images. SAR-BM3D performed somewhat well on natural color images including large repeated textures or various color values, such as the Barbara, Building, and Peppers images, as shown in Figure 5. However, color SAR images do not usually contain large repeated textures or various color values. Instead, they include many small objects and edges. In color SAR images, SAR-BM3D sufficiently eliminates noise in both dark and bright regions and preserves details well, but it produces many erroneous color values in these small objects and edges. Thus, its PSNR and SSIM values are much lower than those of SO-SVNTVTV2 and our model at all noise levels, unlike the case of natural color images. SO-SVNTVTV2 conducts well on color SAR images, by appropriately denoising both dark and bright areas in addition to retaining fine details and edges. Additionally, it does not generate faulty color values or artifacts compared to SAR-BM3D or AA-SVTV. The restored images of SO-SVNTVTV2 and our model are visually very similar, but our model further improves SO-SVNTVTV2 by conserving more details and edges, thereby yielding cleaner and sharper restored images. In all cases, this results in higher PSNR and SSIM values for our model. Therefore, these examples establish the considerable denoising performance of the proposed model, even on real SAR images.
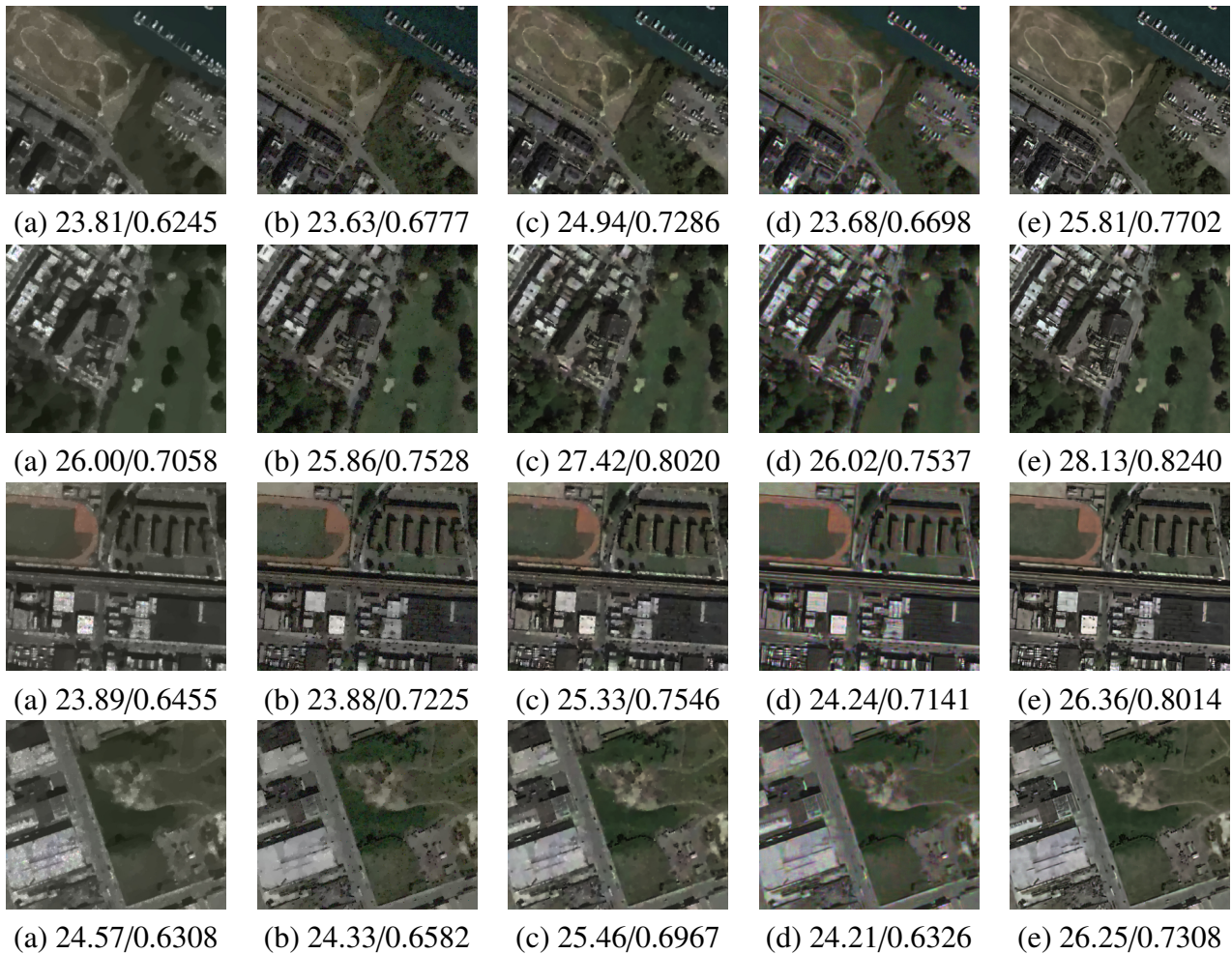
Data images with zoomed-in areas displayed

(a) 23.81/0.6245   (b) 23.63/0.6777   (c) 24.94/0.7286   (d) 23.68/0.6698   (e) 25.81/0.7702

(a) 26.00/0.7058   (b) 25.86/0.7528   (c) 27.42/0.8020   (d) 26.02/0.7537   (e) 28.13/0.8240

(a) 23.89/0.6455   (b) 23.88/0.7225   (c) 25.33/0.7546   (d) 24.24/0.7141   (e) 26.36/0.8014

(a) 24.57/0.6308   (b) 24.33/0.6582   (c) 25.46/0.6967   (d) 24.21/0.6326   (e) 26.25/0.7308

Figure 10. Zoomed-in portions of the restored images when $L = 3$. (a) L2-SVTV, (b) AA-SVTV, (c) SO-SVNTVTV2, (d) SAR-BM3D, (e) Proposed. Parameter of (e): (top to bottom) $\lambda = 0.24, 0.26, 0.24, 0.22$. PSNR/SSIM values are presented.

Figure 11 displays plots of the relative errors of w^ℓ and PSNR values of $u^\ell = e^{w^\ell}$ via the outer iteration number ℓ for the proposed model, when $L = 10, 5$, and 3. The figure shows that as ℓ increases, the relative error ultimately decreases, while the PSNR value increases and converges to a certain value. In addition, as the noise level L increases, a higher number of outer iterations is required for convergence. These plots verify the numerical convergence of the proposed algorithm, even though the theoretical convergence of the algorithm is still a remaining issue.
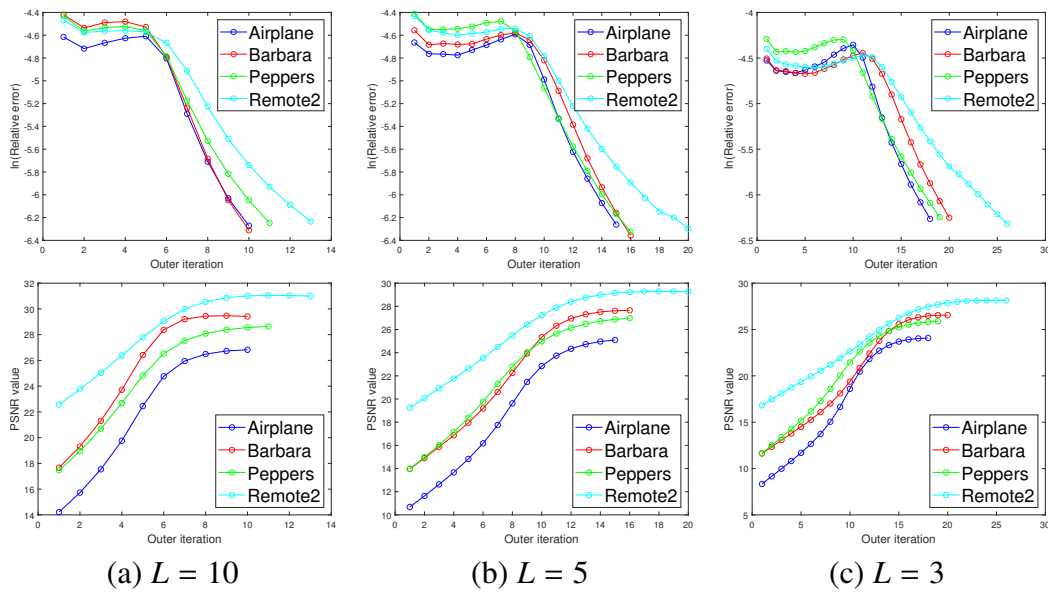
Figure 11. Plots of the relative error and PSNR values via the outer iteration ℓ when $L = 10$, 5, 3. (Top) Relative errors of w^ℓ, $\log\left(\frac{\|w^{\ell+1} - w^\ell\|_2}{\|w^{\ell+1}\|_2}\right)$, (Bottom) PSNR values of $u^\ell = e^{w^\ell}$.

Lastly, in Tables 1–3, we present the PSNR and SSIM values of all models, when $L = 10$, 5, and 3, respectively. The results show that the PSNR and SSIM values of AA-SVTV decrease significantly at noise level $L = 3$. Furthermore, SAR-BM3D provides similar PSNR and SSIM values as SO-SVNTVTV2 for natural color images, but its PSNR and SSIM values are considerably diminished for color SAR images. In all cases, the proposed model yields the highest PSNR and SSIM values. These measurements also ensure the outstanding denoising performance of the proposed model compared to other models.

Table 1. PSNR and SSIM values for all models at noise level $L = 10$.

Image	Data f	L2-SVTV	AA-SVTV	SO-SVNTVTV2	SAR-BM3D	Proposed
Airplane	12.69/0.1970	23.93/0.7176	24.83/0.7965	25.06/0.7985	25.34/0.7419	**26.83/0.8246**
Baboon	15.30/0.4449	21.55/0.5299	22.39/0.6325	22.47/0.6135	22.83/0.6569	**23.54/0.6702**
Barbara	16.00/0.3362	25.16/0.6897	26.25/0.7587	26.48/0.7784	27.66/0.8110	**29.41/0.8699**
Bears	18.12/0.3207	27.83/0.6846	28.38/0.7433	28.85/0.7496	28.18/0.7242	**29.34/0.7690**
Boat	15.16/0.3794	24.42/0.7217	26.16/0.8042	26.44/0.8047	25.80/0.7766	**27.52/0.8279**
Building	17.30/0.3882	25.76/0.7974	25.55/0.7840	26.68/0.8357	26.88/0.8337	**28.68/0.8737**
Butterfly	16.55/0.4424	23.24/0.8235	24.07/0.8508	25.31/0.8841	24.09/0.8558	**26.85/0.9149**
Castle	15.74/0.2609	25.85/0.7728	26.89/0.7946	27.50/0.8421	27.04/0.8179	**28.72/0.8665**
Cathedral	15.37/0.2984	23.69/0.7055	24.74/0.7798	25.21/0.7744	25.16/0.7547	**26.79/0.7999**
Flower	19.31/0.3042	30.82/0.8958	30.92/0.8462	32.12/0.9150	31.53/0.9004	**32.85/0.9253**
Hill	17.06/0.4076	25.64/0.6292	26.83/0.7389	27.05/0.7323	26.96/0.7249	**27.62/0.7401**
House	14.62/0.1839	26.97/0.7481	28.03/0.7816	28.41/0.7904	29.76/0.7776	**30.64/0.8087**
Lena	15.10/0.2795	25.09/0.7358	26.96/0.8072	27.05/0.8109	27.41/0.7968	**29.26/0.8606**
Peppers	15.88/0.3609	25.04/0.7566	25.75/0.8066	26.51/0.8182	27.44/0.8176	**28.64/0.8526**
Policemen	16.76/0.4615	23.41/0.7372	24.92/0.8372	25.70/0.8576	25.09/0.8469	**27.48/0.9086**
Statues	17.39/0.5465	23.84/0.6094	24.88/0.7734	25.23/0.7525	24.18/0.7156	**26.10/0.7933**
Average	16.15/0.3139	25.14/0.6209	26.10/0.6813	26.63/0.6895	26.58/0.6789	**28.14/0.7202**
Remote1	20.05/0.5335	26.79/0.7756	27.03/0.8134	27.83/0.8411	26.19/0.8004	**28.81/0.8689**
Remote2	21.31/0.5074	28.81/0.8124	29.07/0.8497	30.23/0.8822	28.47/0.8481	**31.00/0.8984**
Remote3	19.89/0.6262	26.55/0.7933	27.06/0.8624	27.92/0.8690	26.05/0.8295	**28.83/0.8949**
Remote4	19.68/0.5646	26.47/0.7679	27.28/0.8453	28.32/0.8583	26.80/0.8252	**29.31/0.8797**
Remote5	18.50/0.4210	26.98/0.7480	27.46/0.7952	28.04/0.8052	26.73/0.7587	**28.82/0.8255**
Average	19.87/0.5305	27.12/0.7794	27.58/0.8332	28.47/0.8512	26.85/0.8124	**29.35/0.8735**

[*] Bold values indicate the best denoising performance.

Table 2. PSNR and SSIM values for all models at noise level $L = 5$.

Image	Data f	L2-SVTV	AA-SVTV	SO-SVNTVTV2	SAR-BM3D	Proposed
Airplane	9.66/0.1375	22.54/0.6631	23.35/0.7415	23.59/0.7518	23.70/0.6709	**25.10/0.7781**
Baboon	12.34/0.3214	20.69/0.4526	21.15/0.5497	21.18/0.5063	21.46/0.5419	**22.48/0.6147**
Barbara	12.96/0.2297	24.27/0.6355	24.64/0.6809	24.81/0.6954	25.97/0.7455	**27.66/0.8271**
Bears	15.12/0.2077	26.71/0.6307	26.74/0.6634	27.54/0.6836	26.92/0.6572	**27.98/0.7070**
Boat	12.13/0.3016	23.08/0.6592	24.53/0.7570	24.73/0.7573	24.35/0.7196	**25.98/0.7875**
Building	14.33/0.2730	23.89/0.7217	23.78/0.7022	24.92/0.7786	25.14/0.7756	**27.02/0.8386**
Butterfly	13.61/0.3445	21.47/0.7615	22.02/0.7914	23.42/0.8385	22.63/0.8071	**24.83/0.8792**
Castle	12.73/0.1732	24.55/0.7319	25.12/0.7526	25.93/0.8017	25.49/0.7594	**27.19/0.8347**
Cathedral	12.37/0.2070	22.74/0.6591	23.06/0.7142	23.48/0.7161	23.56/0.6864	**25.19/0.7587**
Flower	16.31/0.2081	29.17/0.8699	29.46/0.8416	30.76/0.8954	30.10/0.8699	**31.50/0.9085**
Hill	14.04/0.2879	24.43/0.5644	25.31/0.6597	25.76/0.6658	25.63/0.6557	**26.67/0.7059**
House	11.60/0.1177	25.48/0.7153	26.28/0.7451	26.89/0.7645	27.85/0.7279	**28.82/0.7811**
Lena	12.14/0.1993	24.17/0.6917	25.33/0.7709	25.62/0.7711	25.84/0.7363	**27.67/0.8234**
Peppers	12.84/0.2734	23.70/0.7107	24.28/0.7513	24.66/0.7839	25.83/0.7727	**26.98/0.8239**
Policemen	13.72/0.3619	21.82/0.6658	23.09/0.7760	23.78/0.7982	23.60/0.7856	**25.60/0.8615**
Statues	14.35/0.4083	22.54/0.5237	23.24/0.6823	23.55/0.6469	22.92/0.6234	**24.52/0.7188**
Average	13.14/0.2147	23.83/0.5733	24.46/0.6304	25.04/0.6399	25.06/0.6220	**26.57/0.6832**
Remote1	17.05/0.3980	24.92/0.6857	25.04/0.7425	26.11/0.7800	24.71/0.7295	**26.99/0.8147**
Remote2	18.29/0.3773	27.05/0.7483	27.14/0.8030	28.50/0.8380	27.02/0.7956	**29.27/0.8577**
Remote3	16.89/0.4868	24.60/0.7005	25.00/0.7987	26.11/0.8114	24.64/0.7644	**27.06/0.8507**
Remote4	16.67/0.4314	25.03/0.7070	25.25/0.7809	26.52/0.8024	25.32/0.7649	**27.60/0.8397**
Remote5	15.49/0.2986	25.66/0.6906	25.66/0.7244	26.51/0.7471	25.29/0.6906	**27.25/0.7688**
Average	16.88/0.3984	25.45/0.7064	25.62/0.7699	26.75/0.7958	25.40/0.7490	**27.63/0.8263**

* Bold values indicate the best denoising performance.

Table 3. PSNR and SSIM values for all models at noise level $L = 3$.

Image	Data f	L2-SVTV	AA-SVTV	SO-SVNTVTV2	SAR-BM3D	Proposed
Airplane	7.41/0.1020	21.86/0.5995	22.11/0.6777	22.73/0.7236	22.71/0.6098	**24.08/0.7308**
Baboon	10.12/0.2379	20.06/0.4003	20.03/0.4381	20.33/0.4231	20.60/0.4604	**21.13/0.4823**
Barbara	10.80/0.1654	23.51/0.5944	23.44/0.6154	24.08/0.6503	24.75/0.6823	**26.55/0.7818**
Bears	12.87/0.1438	26.02/0.5990	25.72/0.6054	26.72/0.6331	26.05/0.6092	**27.06/0.6449**
Boat	9.91/0.2481	22.37/0.6059	22.99/0.7008	23.72/0.7197	23.39/0.6704	**24.95/0.7456**
Building	12.07/0.1987	22.91/0.6753	22.62/0.6418	23.65/0.7241	23.86/0.7200	**25.86/0.7960**
Butterfly	11.39/0.2756	20.46/0.7176	20.76/0.7504	22.17/0.8021	21.51/0.7607	**23.49/0.8444**
Castle	10.46/0.1229	23.81/0.6931	23.84/0.7015	24.69/0.7628	24.25/0.7081	**25.94/0.7863**
Cathedral	10.12/0.1505	21.88/0.6228	22.02/0.6584	22.52/0.6740	22.54/0.6341	**24.10/0.7122**
Flower	14.10/0.1508	28.35/0.8524	28.32/0.8059	29.80/0.8817	28.95/0.8430	**30.52/0.8927**
Hill	11.83/0.2107	23.90/0.5335	24.08/0.5894	24.91/0.6182	24.65/0.5982	**25.54/0.6374**
House	9.37/0.0813	24.51/0.6767	24.76/0.7051	25.65/0.7388	26.30/0.6781	**27.56/0.7530**
Lena	9.92/0.1492	23.36/0.6428	23.91/0.7232	24.52/0.7344	24.57/0.6810	**26.52/0.7863**
Peppers	10.70/0.2187	22.57/0.6746	22.90/0.7158	23.78/0.7554	24.62/0.7324	**25.88/0.7977**
Policemen	11.52/0.2917	21.29/0.6291	22.01/0.7199	22.62/0.7486	22.58/0.7318	**24.39/0.8188**
Statues	12.15/0.3132	21.79/0.4702	22.11/0.6057	22.84/0.6073	22.11/0.5667	**23.46/0.6569**
Average	10.92/0.1616	23.04/0.5371	23.23/0.5789	24.05/0.6044	23.97/0.5753	**25.44/0.6398**
Remote1	14.86/0.3045	23.81/0.6245	23.63/0.6777	24.94/0.7286	23.68/0.6698	**25.81/0.7702**
Remote2	16.06/0.2884	26.00/0.7058	25.86/0.7528	27.42/0.8020	26.02/0.7537	**28.13/0.8240**
Remote3	14.68/0.3837	23.42/0.6331	23.56/0.7425	24.93/0.7627	23.63/0.7103	**25.84/0.8089**
Remote4	14.42/0.3371	23.89/0.6455	23.88/0.7225	25.33/0.7546	24.24/0.7141	**26.36/0.8014**
Remote5	13.27/0.2206	24.57/0.6308	24.33/0.6582	25.46/0.6967	24.21/0.6326	**26.25/0.7308**
Average	14.66/0.3069	24.34/0.6479	24.25/0.7107	25.62/0.7489	24.36/0.6961	**26.48/0.7870**

* Bold values indicate the best denoising performance.

5. Conclusions

This paper proposes a multi-color channel-based GSR model for the removal of multiplicative gamma noise in color images. The proposed GSR comprises an image group with similar color patches, by measuring the distance between two color patches. This prevented artificial color values produced by separate color channel-based approaches. Furthermore, GSR, which exploits the sparsity and nonlocal similarity of patches, aided in the conservation of fine structures and repeated textures. Additionally, it adequately denoised smooth regions without the staircase artifacts caused by TV-based models, which led to the natural visual quality of restored images. The model also includes a convex data-fidelity term and SVTV regularization. The data-fidelity term facilitated the removal of heavy multiplicative noise, while SVTV helped alleviate some ringing or block artifacts induced by patches, thus enhancing the denoising results. We adopted the ADMM to solve the proposed nonconvex model and its subproblem. These approaches yielded an efficacious iterative algorithm. The experimental results validated the superior denoising performance of the proposed model over other existing models, when it comes to both visual inspection and image quality measurements. However, despite the numerical convergence of the proposed algorithm, the theoretical convergence of the proposed algorithm still remains an issue. Furthermore, the GSR approach requires more computational time than local regularization-based methods due to searching for similar patches and the singular value decomposition of groups. Thus, faster implementation also needs to be investigated in future work.

Use of AI tools declaration

The author declares they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

Conflict of interest

The author declares no conflicts of interest in this paper.

References

1. C. J. Oliver, S. Quegan, *Understanding synthetic aperture radar images*, SciTech Publishing, Inc., Raleigh, NC, 2004.

2. R. F. Wagner, S. W. Smith, J. M. Sandrik, H. Lopez, Statistics of speckle in ultrasound B-scans, *IEEE Trans. Sonics Ultrason.,* **30** (1983), 156–163. https://doi.org/10.1109/T-SU.1983.31404

3. J. M. Schmitt, S. Xiang, K. M. Yung, Speckle in optical coherence tomography, *J. Biomed. Opt.,* **4** (1999), 95–105. https://doi.org/10.1117/1.429925

4. J. W. Goodman, Some fundamental properties of speckle, *J. Opt. Soc. Amer.,* **66** (1976), 1145–1150. https://doi.org/10.1364/JOSA.66.001145

5. J. S. Lee, Digital image enhancement and noise filtering by use of local statistics, *IEEE Trans. Pattern Anal. Mach. Intell.,* **2** (1980), 165–168. https://doi.org/10.1109/TPAMI.1980.4766994

6. V. S. Frost, J. A. Stiles, K. S. Shanmugan, J. C. Holtzman, A model for radar images and its application to adaptive digital filtering of multiplicative noise, *IEEE Trans. Pattern Anal. Mach. Intell.,* **PAMI-4** (1982), 157–166. https://doi.org/10.1109/TPAMI.1982.4767223

7. Y. Yu, S. T. Acton, Speckle reducing anisotropic diffusion, *IEEE Trans. Image Process.,* **11** (2002), 1260–1270. https://doi.org/10.1109/TIP.2002.804276

8. K. Krissian, C. F. Westin, R. Kikinis, K. G. Vosburgh, Oriented speckle reducing anisotropic diffusion, *IEEE Trans. Image Process.,* **16** (2007), 1412–1424. https://doi.org/10.1109/TIP.2007.891803

9. S. Parrilli, M. Poderico, C. V. Angelino, L. Verdoliva, A nonlocal SAR image denoising algorithm based on LLMMSE wavelet shrinkage, *IEEE Trans. Geosci. Remote Sens.,* **50** (2012), 606–616. https://doi.org/10.1109/TGRS.2011.2161586

10. G. Aubert, J. F. Aujol, A variational approach to removing multiplicative noise, *SIAM J. Appl. Math.,* **68** (2008), 925–946. https://doi.org/10.1137/060671814

11. J. Shi, S. Osher, A nonlinear inverse scale space method for a convex multiplicative noise model, *SIAM J. Imaging Sci.,* **1** (2008), 294–321. https://doi.org/10.1137/070689954

12. Y. M. Huang, M. K. Ng, Y. W. Wen, A new total variation method for multiplicative noise removal, *SIAM J. Imaging Sci.,* **2** (2009), 20–40. https://doi.org/10.1137/080712593

13. G. Steidl, T. Teuber, Removing multiplicative noise by Douglas-Rachford splitting methods, *J. Math. Imaging Vis.,* **36** (2010), 168–184. https://doi.org/10.1007/s10851-009-0179-5

14. S. Yun, H. Woo, A new multiplicative denoising variational model based on m-th root transformation, *IEEE Trans. Image Process.,* **21** (2012), 2523–2533. https://doi.org/10.1109/TIP.2012.2185942

15. Y. M. Huang, L. Moisan, M. K. Ng, T. Zeng, Multiplicative noise removal via a learned dictionary, *IEEE Trans. Image Process.,* **21** (2012), 4534–4543. https://doi.org/10.1016/j.dsp.2015.12.012

16. Y. Dong, T. Zeng, A convex variational model for restoring blurred images with multiplicative noise, *SIAM J. Imaging Sci.,* **6** (2013), 1598–1625. https://doi.org/10.1137/120870621

17. M. Kang, S. Yun, H. Woo, Two-level convex relaxed variational model for multiplicative denoising, *SIAM J. Imaging Sci.,* **6** (2013), 875–903. https://doi.org/10.1137/11086077X

18. Z. Li, Y. Lou, T. Zeng, Variational multiplicative noise removal by DC programming, *J. Sci. Comput.,* **68** (2016), 1200–1216. https://doi.org/10.1007/s10915-016-0175-z

19. J. Lu, L. Shen, C. Xu, Y. Xu, Multiplicative noise removal in imaging: An exp-model and its fixed-point proximity algorithm, *Appl. Comput. Harmon. Anal.,* **41** (2016), 518–539. https://doi.org/10.1016/j.acha.2015.10.003

20. H. Na, M. Kang, M. Jung, M. Kang, An exp model with spatially adaptive regularization parameters for multiplicative noise removal, *J. Sci. Comput.,* **75** (2018), 478–509. https://doi.org/10.1007/s10915-017-0550-4

21. H. Na, M. Kang, M. Jung, M. Kang, Nonconvex TGV regularization model for multiplicative noise removal with spatially varying parameters, *Inverse Probl. Imaging,* **13** (2019), 117–147. https://doi.org/10.3934/ipi.2019007

22. X. Liu, J. Lu, L. Shen, C. Xu, Y. Xu, Multiplicative noise removal: Nonlocal low-rank model and its proximal alternating reweighted minimization algorithm, *SIAM J. Imaging Sci.,* **13** (2020), 1595–1629. https://doi.org/10.1137/20M1313167

23. W. Wang, M. Yao, M. K. Ng, Color image multiplicative noise and blur removal by saturation-value total variation, *Appl. Math. Model.,* **90** (2021), 240–264. https://doi.org/10.1016/j.apm.2020.08.052

24. M. Jung, Saturation-value based higher-order regularization for color image restoration, *Multidim. Syst. Sign. P.,* **34** (2023), 365–394. https://doi.org/10.1007/s11045-023-00867-x

25. L. I. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, *Physica D,* **60** (1992), 259–268. https://doi.org/10.1016/0167-2789(92)90242-F

26. G. Sapiro, *Vector-valued active contours*, Proceedings of CVPR, 680–685, San Francisco, CA, USA, 1996. https://doi.org/10.1109/CVPR.1996.517146

27. P. Blomgren, T. F. Chan, Total variation methods for restoration of vector valued images, *IEEE T. Image Process.,* **7** (1998), 304–309. https://doi.org/10.1109/83.661180

28. R. Kimmel, R. Malladi, N. Sochen, Images as embedded maps and minimal surfaces: Movies, color, texture, and volumetric medical images, *Int. J. Comput. Vis.,* **39** (2000), 111–129. https://doi.org/10.1023/A:1008171026419

29. T. Chan, S. Kang, J. Shen, Total variation denoising and enhancement of color images based on the CB and HSV color models, *J. Vis. Commun. Image R.,* **12** (2001), 422–435. https://doi.org/10.1006/jvci.2001.0491

30. H. Attouch, G. Buttazzo, G. Michaille, *Variational analysis in Sobolev and BV spaces: Applications to pdes and optimization*, MPS-SIAM Ser. Optim., 6, SIAM, Philadelphia, 2006.

31. X. Bresson, T. F. Chan, Fast dual minimization of the vectorial total variation norm and applications to color image processing, *Inverse Probl. Imag.,* **2** (2008), 255–284. https://doi.org/10.3934/ipi.2008.2.455

32. Y. Wen, M. Ng, Y. Huang, Efficient total variation minimization methods for color image restoration, *IEEE T. Image Process.,* **17** (2008), 2081–2088. https://doi.org/10.1109/TIP.2008.2003406

33. P. Rodriguez, B. Wohlberg, *A generalized vector-valued total variation algorithm*, In: Image Processing (ICIP 2009), Cairo, Egypt, 2009, 1309–1312. https://doi.org/0.1109/ICIP.2009.5413587

34. C. Wu, X. C. Tai, Augmented lagrangian method, dual methods, and split bregman iteration for ROF, vectorial TV, and high order models, *SIAM J. Imaging Sci.,* **3** (2010), 300–339. https://doi.org/10.1137/090767558

35. S. Ono, I. Yamada, *Decorrelated vectorial total variation*, Proceedings of CVPR, Columbus, OH, USA, 2014, 4090–4097. https://doi.org/10.1109/CVPR.2014.521

36. R. Bergmann, R. H. Chan, R. Hielscher, J. Persch, G. Steidl, Restoration of manifold-valued images by half-quadratic minimization, *Inverse Probl. Imag.,* **10** (2016), 281–304. https://doi.org/10.3934/ipi.2016001

37. J. Duran, M. Moeller, C. Sbert, D. Cremers, Collaborative total variation: A general framework for vectorial TV models, *SIAM J. Imaging Sci.,* **9** (2016), 116–151. https://doi.org/10.1137/15M102873X

38. Z. Jia, M. K. Ng, W. Wang, Color image restoration by saturation-value total variation, *SIAM J. Imaging Sci.,* **12** (2019), 972–1000. https://doi.org/10.1137/18M1230451

39. A. Buades, B. Coll, J. M. Morel, *A non-local algorithm for image denoising*, Proceedings of CVPR, San Diego, CA, USA, 2005, 60–65. https://doi.org/10.1109/CVPR.2005.38

40. K. Dabov, A. Foi, V. Katkovnik, K. Egiazarian, Image denoising by sparse 3-d transform-domain collaborative filtering, *IEEE T. Image Process.,* **16** (2007), 2080–2095. https://doi.org/10.1109/TIP.2007.901238

41. S. Kindermann, S. Osher, P. W. Jones, Deblurring and denoising of images by nonlocal functionals, *Multiscale Model. Sim.,* **4** (2005), 1091–1115. https://doi.org/10.1137/050622249

42. A. Elmoataz, O. Lezoray, S. Bougleux, Nonlocal discrete regularization on weighted graphs: A framework for image and manifold processing, *IEEE T. Image Process.,* **17** (2008), 1047–1060. https://doi.org/10.1109/TIP.2008.924284

43. G. Peyré, Image processing with nonlocal spectral bases, *Multiscale Model. Sim.,* **7** (2008), 703–730. https://doi.org/10.1137/07068881X

44. X. Zhang, M. Burger, X. Bresson, S. Osher, Bregmanized nonlocal regularization for deconvolution and sparse reconstruction, *SIAM J. Imaging Sci.,* **3** (2010), 253–276. https://doi.org/10.1137/090746379

45. M. Jung, X. Bresson, T. F. Chan, L. A. Vese, Nonlocal Mumford-Shah regularizers for color image restoration, *IEEE T. Image Process.,* **20** (2011), 1583–1598. https://doi.org/10.1109/TIP.2010.2092433

46. M. Elad, M. Aharon, Image denoising via sparse and redundant representations over learned dictionaries, *IEEE T. Image Process.,* **15** (2006), 3736–3745. https://doi.org/10.1109/TIP.2008.2008065

47. J. Mairal, F. Bach, J. Ponce, G. Sapiro, A. Zisserman, *Non-local sparse models for image restoration*, Proceedings of ICCV, Tokyo, Japan, 2009, 2272–2279. https://doi.org/10.1109/ICCV.2009.5459452

48. W. Dong, L. Zhang, G. Shi, X. Wu, Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization, *IEEE T. Image Process.,* **20** (2011), 1838–1857. https://doi.org/10.1109/TIP.2011.2108306

49. W. Dong, L. Zhang, G. Shi, X. Li, Nonlocally centralized sparse representation for image restoration, *IEEE T. Image Process.,* **22** (2013), 1620–1630. https://doi.org/10.1109/TIP.2012.2235847

50. W. Dong, G. Shi, X. Li, Nonlocal image restoration with bilateral variance estimation: A low-rank approach, *IEEE T. Image Process.,* **22** (2013), 700–711. https://doi.org/10.1109/TIP.2012.2221729

51. S. Gu, L. Zhang, W. Zuo, X. Feng, *Weighted nuclear norm minimization with application to image denoising*, Proceedings of CVPR, Columbus, OH, USA, 2014, 2862–2869. https://doi.org/10.1109/CVPR.2014.366

52. T. Huang, W. Dong, X. Xie, G. Shi, X. Bai, Mixed noise removal via Laplacian scale mixture modeling and nonlocal low-rank approximation, *IEEE T. Image Process.,* **26** (2017), 3171–3186. https://doi.org/10.1109/TIP.2017.2676466

53. J. Zhang, D. Zhao, W. Gao, Group-based sparse representation for image restoration, *IEEE T. Image Process.,* **23** (2014), 3336–3351. https://doi.org/10.1109/TIP.2014.2323127

54. J. Zhang, S. Ma, Y. Zhang, W. Gao, *Image deblocking using group-based sparse representation and quantization constraint prior*, Proceedings of ICIP, Quebec City, QC, Canada, 2015, 306–310. https://doi.org/10.1109/ICIP.2015.7350809

55. W. Shi, C. Chen, F. Jiang, D. Zhao, W. Shen, *Group-based sparse representation for low lighting image enhancement*, Proceedings of ICIP, Phoenix, AZ, USA, 2016, 4082–4086. https://doi.org/10.1109/ICIP.2016.7533127

56. S. Liu, G. Zhang, Y. T. Soon, An over-complete dictionary design based on GSR for sar image despeckling, *IEEE Geosci. Remote Sens. Lett.,* **14** (2017), 2230–2234. https://doi.org/10.1109/LGRS.2017.2758397

57. S. Lee, M. Kang, Group sparse representation for restoring blurred images with Cauchy noise, *J. Sci. Comput.,* **83** (2020), 41. https://doi.org/10.1007/s10915-020-01227-8

58. Z. Zha, X. Yuan, B. Wen, J. Zhou, J. Zhang, C. Zhu, A Benchmark for sparse coding: When group sparsity meets rank minimization, *IEEE T. Image Process.,* **29** (2020), 5094–5109. https://doi.org/10.1109/TIP.2020.2972109

59. Z. Zha, X. Yuan, B. Wen, J. Zhang, J. Zhou, C. Zhu, Image restoration using joint patch-group based sparse representation, *IEEE T. Image Process.,* **29** (2020), 7735–7750. https://doi.org/10.1109/TIP.2020.3005515

60. Z. Zha, X. Yuan, B. Wen, J. Zhou, C. Zhu, Group sparsity residual constraint with non-local priors for image restoration, *IEEE T. Image Process.,* **29** (2020), 8960–8975. https://doi.org/10.1109/TIP.2020.3021291

61. Y. Kong, C. Zhou, C. Zhang, L. Sun, C. Zhou, Multi-color channels based group sparse model for image restoration, *Algorithms,* **15** (2022), 176. https://doi.org/10.3390/a15060176

62. S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, *Found. Trends Mach. Learn.,* **3** (2010), 1–122. https://doi.org/10.1561/2200000016

63. Y. Wang, W. Yin, J. Zeng, Global convergence of ADMM in nonconvex nonsmooth optimization, *J. Sci. Comput.,* **78** (2019), 1–2. https://doi.org/10.1007/s10915-018-0757-z

64. J. Mota, J. Xavier, P. Aguiar, M. Puschel, A proof of convergence for the alternating direction method of multipliers applied to polyhedral-constrained functions, *arXiv.1112.2295*, 2011.

65. Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, Image quality assessment: From error visibility to structural similarity, *IEEE Trans. Image Process.,* **13** (2004), 600–612. https://doi.org/10.1109/TIP.2003.819861