

Research Article

A Cartoon-Texture Decomposition Based Multiplicative Noise Removal Method

Chenping Zhao,^{1,2} Xiangchu Feng,¹ Weiwei Wang,¹ and Huazhu Chen¹

¹School of Mathematics and Statistics, Xidian University, Xi'an 710026, China

²School of Mathematical Science, Henan Institute of Science and Technology, Xinxiang 453003, China

Correspondence should be addressed to Chenping Zhao; zcp0378@163.com and Xiangchu Feng; xcfeng@mail.xidian.edu.cn

Received 23 April 2016; Revised 29 June 2016; Accepted 26 July 2016

Academic Editor: Maria L. Gandarias

Copyright © 2016 Chenping Zhao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We propose a new frame for multiplicative noise removal. To improve the multiplicative denoising performance, we add the regularization of texture component in the denoising model, designing a multiscale multiplicative noise removal model. The proposed model is jointly convex and can be easily solved by optimization algorithms. We introduce Douglas-Rachford splitting method to solve the proposed model. In the algorithm, we make full use of some important proximity operators, which have closed expression or can be executed in one time iteration. In particular, the proximity of H^{-1} norm is deduced, which is just the Fourier domain filtering. In the process of simulation experiments, we first analyze and select the needed parameters and then test the experiments on several images using the designed algorithm and the given parameters. Finally, we compare the denoising performance of the proposed model with the existing models, in which the signal to noise ratio (SNR) and the peak signal to noise ratios (PSNRs) are applied to evaluate the noise suppressing effects. Experimental results demonstrate that the designed algorithms can solve the model perfectly and the recovery images of the proposed model have higher SNRs/PSNRs and better visual quality.

1. Introduction

Image denoising is a basic and important task in image processing. The relatively mature developed denoising model is the additive noise model in which case the noise is assumed to obey a Gaussian distribution, that is,

$$g = f + n, \quad n \sim N(0, \sigma^2). \quad (1)$$

However, the noises involved in many applications do not conform to the characteristic of the additive one; they may corrupt an image in other forms. In this paper, we are concerned with the denoising problem under the assumption that the original image has been corrupted by some multiplicative noise. The corresponding examples include the uneven phenomena during the magnetic resonance imaging (MRI) and the speckle noise in the ultrasonic and in synthetic aperture radar (SAR) image. The degradation model of the multiplicative noise can be expressed as

$$g = f \cdot n, \quad (2)$$

where \cdot refers to the componentwise multiplication. To model the actual problems, we assume that the multiplicative noise

obeys some random distribution; for example, the noise in SAR images is assumed to obey the Gamma distribution and the one in ultrasonic image obeys the Rayleigh distribution.

1.1. Multiplicative Noise Removal Models. There are many multiplicative noise removal models; one of the most classic models based on TV regularization is called RLO model proposed by Rudin et al. [1]:

$$\begin{aligned} \min_f \quad & \int_{\Omega} |\nabla f| dx \\ \text{s.t.} \quad & \int_{\Omega} \frac{g(x)}{f(x)} dx = 1, \\ & \int_{\Omega} \left(\frac{g(x)}{f(x)} - 1 \right)^2 dx = \sigma^2, \end{aligned} \quad (3)$$

where the mean of noise is assumed to be 1 and the variance is assumed to be σ^2 . Since (3) is a nonconvex problem, it is very difficult to be solved.

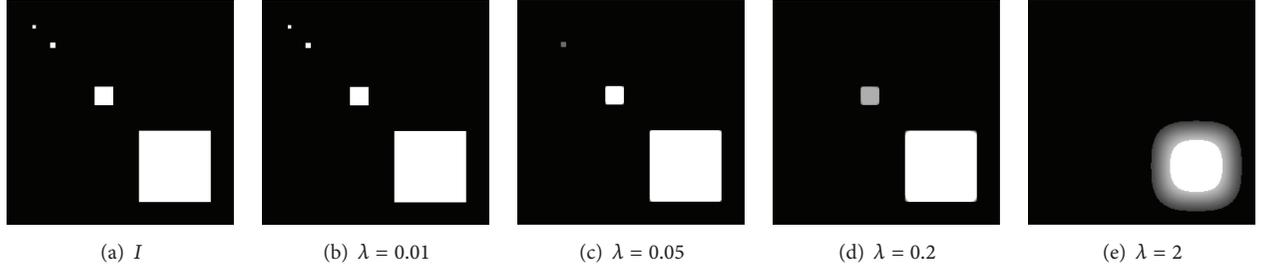


FIGURE 1: Numerical experiment results of (10) with different λ : (a) is the original image I ; (b)–(e) are the minimizer f with $\lambda = 0.01, 0.05, 0.2, \text{ and } 2$, respectively.

The second classic model is called AA model [2] which is given by Aubert and Aujol under the hypothesis of Gamma distribution:

$$\min_f \int_{\Omega} \left(\log f + \frac{g(x)}{f(x)} \right) dx + \lambda \|\nabla f\|_1. \quad (4)$$

The objective function in (4) is also nonconvex. However, Aubert and Aujol showed the existence of minimizers of the objective function and employed a gradient method to solve (4) numerically. There are several improved works based on (4) developed in recent years.

Recently, Shi and Osher [3] proposed considering a logarithm transformation on the noisy observation, $\log g = \log + \log n$, and derived the TV minimization model for multiplicative noise removal problems. Huang et al. also proposed the log-domain denoising model using the transformation $f = \exp(z)$ in (4). Consider

$$\min_z \int_{\Omega} (z(x) + g(x) e^{-z(x)}) dx + \lambda \|z\|_{\text{TV}}, \quad (5)$$

which is known as EXP model [4]. The objective function in (5) is strictly convex in z and the authors promoted an alternating minimization algorithm to solve the model and showed the convergence of the method simultaneously. Nevertheless, model (5) is convex in z rather than in f in the original image domain. At the same time, Durand et al. [5] proposed a method composed of several stages. They also used the log-image data and applied reasonable suboptimal hard thresholding on its curvelet transform; then they applied a variational method by minimizing a specialized hybrid criterion composed of an l_1 data-fidelity term of the thresholded curvelet coefficients and a TV regularization term in the log-image domain. The restored image can be obtained by using an exponential of the minimizer, weighted in such a way that the mean of the original image is preserved. Their restored images combine the advantages of shrinkage and variational methods. Besides the above approaches, dictionary learning methods and nonlocal mean methods have also been proposed and developed for multiplicative denoising [6–9].

In [10], Zhao et al. developed a convex optimization model for multiplicative noise removal. The main idea is to rewrite a multiplicative noise equation such that both the image variable and the noise variable are decoupled. That is to say, rewrite problem (2) as

$$g = Nf, \quad (6)$$

where $N = \text{diag}(n)$ is a diagonal matrix of which the main diagonal entries are given by $[n]_i$. According to (2), when there is no noise in the observed image, we obtain that $n = e$, a vector of all ones. When there is a multiplicative noise in the observed image, we expect that $\{[n]_{i_0} \neq 0, \forall i\}$, and moreover they are greater than zeros. So we can say that N is invertible, and (6) is equivalent to

$$Gw = f, \quad (7)$$

where $G = \text{diag}(g)$ is the diagonalizable matrix of vector g and $[w]_i = 1/[n]_i$. The convex model to solve (2) is proposed as

$$\min_{w,f} \frac{1}{2} \|w - \mu e\|_2^2 + \alpha_1 \|Gw - f\|_1 + \alpha_2 \|f\|_{\text{TV}}. \quad (8)$$

In (8), the first term is to measure the variance of w , the second term is the data term, and the third term is the TV regularization term. If the first term is absent and w can be arbitrarily assigned, then minimizing

$$\min_{w,f} \alpha_1 \|Gw - f\|_1 + \alpha_2 \|f\|_{\text{TV}} \quad (9)$$

may lead to the trivial solution $f \equiv C$, $w \equiv CG^{-1}$, where C is a constant. So the existence of variance term $(1/2)\|w - \mu e\|_2^2$ can promise a proper minimizer of (8) [10]. Furthermore, It is worth noting that the image variable and the noise variable are decoupled, and the model is jointly convex for (f, w) , which make the model able to be easily solved by optimizing methods.

1.2. The Contribution. We firstly analyze the performance of (8). It is observed that, if we fix w and set $I = Gw$, the minimizing problem (8) is just a TV-L1 problem, denoted as

$$\min_f \alpha_1 \|I - f\|_1 + \alpha_2 \|f\|_{\text{TV}}. \quad (10)$$

In [11], it was pointed out that minimizing (10) can extract the large scale component and leave out the small scale contents of image I . In other words, we can extract cartoon component with different scales based on the selection of parameter $\lambda = \alpha_2/\alpha_1$. We verify the above conclusion through the following experiment. The example is shown in Figure 1(a), an image I with four squares with size of $3 \times 3, 5 \times 5, 20 \times 20$, and 80×80

pixels, respectively. The experiment results are the following: when $\lambda = 0.01$, all the four squares can be extracted, as shown in Figure 1(b); when $\lambda = 0.05$, three squares can be extracted while the one of 3×3 is lost, as shown in Figure 1(c); when $\lambda = 0.2$, the two big squares appear, and another two are lost, as shown in Figure 1(d); when $\lambda = 2$, only the biggest square 80×80 remains, while all other ones are lost, as shown in Figure 1(e). The above results are consistent with the theoretical analysis in [11]; the bigger λ is, the bigger the scale of extraction is. As a result, we have sound reasons to believe that the minimizer f of model (10) is almost the cartoon component of the whole restored image we expect.

The above analysis shows that, if we want to recover more contents from an image I , we should choose a small λ , which bring much more small scales and details. However, for a noisy image, the small scale component may contain much more noise, which may hinder the denoising quality. For this reason, we will improve the denoising model through adding the prior information of the texture component and design a cartoon-texture decomposition based denoising model. Then, we design the numerical algorithm based on the Douglas-Rachford splitting algorithm and three useful proximity operators. Before conducting the experiment, we analyze the selection of the corresponding parameters. Finally, we verify the performance of our proposed model through comparing the SNRs/PSNRs indexes with some existing models.

The rest of the paper is organized as follows. We propose our new model in next section. In Section 3, the numerical method is presented, in which the Douglas-Rachford (DR) splitting algorithm and some proximal operators used in this paper are firstly reviewed, and the algorithm is then designed and described in detail. In Section 4, the denoising experiments on two types of multiplicative noise are implemented, and the experiment results and performance analyses are unfolded. Finally, we conclude our work in Section 5.

2. The Proposed Model

Based on the analysis in Section 1.2, the recovery image obtained by solving (8) mainly contains the cartoon component of what we expect, while most of the significant texture components are lost. To optimize the denoising performance, we improve model (8) by adding the texture information and alter the regularization term $\|f\|_{TV}$ to be $\|u\|_{TV}$, where u denotes the cartoon component of the restored image f . Taking into consideration the prior information of texture $v = f - u$, we select $\|v\|_{H^{-1}}^2$ as the regularization term about v , for that minimizing $\|\cdot\|_{H^{-1}}^2$ can extract the texture component very well. In a word, the new model is

$$\begin{aligned} \{w^*, u^*, v^*\} = \arg \min_{w, u, v} & \frac{1}{2} \|w - \mu e\|_2^2 \\ & + \alpha_1 \|Gw - (u + v)\|_1 \\ & + \alpha_2 (\|u\|_{TV} + \rho \|v\|_{H^{-1}}^2), \end{aligned} \quad (11)$$

and the restored image is $f^* = u^* + v^*$. In (11), α_1 , α_2 , and ρ are the positive regularization parameters to control the balance among the terms in the objective function.

The proposed model possesses the following superiorities. Firstly, we add the term $\rho \|v\|_{H^{-1}}^2$ in the regularization part, which can help to recover more image information and improve the quality of the restored image. Secondly, α_2/α_1 and ρ can be adjusted according to the noise level, which makes the model have multiscale property. Thirdly, when $\rho \rightarrow +\infty$, then $v \rightarrow 0$, and (11) degrades to model (8); that is, (11) is the modified and improved version of (8). Finally, the proposed model keeps the structure of (8), is jointly convex for (w, u, v) , and can be easily solved by optimal methods. In the following section, we use alternative iteration method and Douglas-Rachford method to deal with it, which combine the application of several proximity operators.

3. The Numerical Method

3.1. Douglas-Rachford Algorithm and Some Proximity Operators. Before proposing the numerical scheme of (11), we will review some basic algorithms and operators. The first one is the Douglas-Rachford splitting algorithm.

Definition 1. Let $\Phi(x)$ and $\Psi(x)$ be the proper, l.s.c., and convex functions such that

$$(\text{ridom } \Psi) \cap (\text{ridom } \Phi) \neq \emptyset, \quad (12)$$

and $\Psi(x) + \Phi(x) \rightarrow +\infty$ as $\|x\| \rightarrow +\infty$; then the problem

$$\min_{x \in R^N} \Psi(x) + \Phi(x) \quad (13)$$

admits at least one solution and, for any $\gamma \in (0, +\infty)$, its solutions are characterized by Algorithm 2.

Algorithm 2 (Douglas-Rachford algorithm). Consider the following:

Fixing $\epsilon \in (0, 1)$, $\gamma > 0$, $y_0 \in R^N$,

For $n = 0, 1, \dots$, do

$$x_n = \text{prox}_{\gamma\Phi} y_n$$

$$\lambda_n \in [\epsilon, 2 - \epsilon]$$

$$y_{n+1} = y_n + \lambda_n (\text{prox}_{\gamma\Psi}(2x_n - y) - x_n),$$

end for

where the operator prox_Φ will be defined in the following.

Secondly, we review the definition of proximity operator and present several special examples that will be used in the numerical scheme.

Definition 3. Let Φ be a proper, l.s.c., and convex function; for every $x \in R^N$, the minimization problem

$$\min_u \Phi(u) + \frac{1}{2} \|u - x\|_2^2 \quad (14)$$

admits a unique solution, which is denoted by $\text{prox}_\Phi(x)$, and the operator $\text{prox}_\Phi(x) : R^N \rightarrow R^N$ thus defined is called the proximity operator of Φ .

In the following, we list three examples of proximity operator which will be used in this paper:

(1) $\Phi(x) = \|\lambda^T x\|_1$, $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_N)^T \in R^N$. Set $y = \text{prox}_{\gamma\Phi}(x)$; then y can be componentwise expressed in the close form:

$$y_i = \begin{cases} \text{sign}(x_i) (|x_i| - \lambda_i \gamma), & |x_i| \geq \lambda_i \gamma \\ 0, & \text{else} \end{cases} \quad (15)$$

$i = 1, 2, \dots, N$,

which is known as the soft thresholding operator, and we denote it in short by $y = \text{ST}_{\lambda\gamma}(x)$.

(2) Consider $\Phi(x) = \|x\|_{\text{TV}}$. Computing $\text{prox}_{\gamma\Phi}(x)$ is equivalent to solving the TV-L2 problem

$$\min_u \frac{1}{2} \|u - x\|_2^2 + \gamma \|u\|_{\text{TV}}. \quad (16)$$

There are many strategies to solve it. One of the most classical methods is the semi-implicit gradient descent algorithm proposed by Chambolle in [12]; another is the Forward-Backward method developed in [5]. Both of the above methods need inner loop in the processing of numerical scheme. In this paper, we adopt the variable splitting method and express the problem in discrete form:

$$\min_u \frac{1}{2} \|u - x\|_2^2 + \gamma \|\nabla u\|_1, \quad (16')$$

where ∇ is the gradient operator given by

$$\nabla = (\nabla^{(1)}, \nabla^{(2)}),$$

$$[\nabla(u)]_i = ([\nabla^{(1)}(u)]_i, [\nabla^{(2)}(u)]_i), \quad i = 1, 2, \dots, N, \quad (17)$$

$$\|\nabla u\|_1 = \sum_{i=1}^N \sqrt{[\nabla^{(1)}(u)]_i^2 + [\nabla^{(2)}(u)]_i^2}.$$

We introduce an auxiliary variable $p \in R^{N \times 2}$ to transfer ∇u :

$$\begin{aligned} p_i &= (p_{i,1}, p_{i,2}) \\ &= [\nabla(u)]_i \\ &= ([\nabla^{(1)}(u)]_i, [\nabla^{(2)}(u)]_i), \quad i = 1, 2, \dots, N; \end{aligned} \quad (18)$$

then we get the following approximation model to (16'):

$$\min_{u,p} \frac{1}{2} \|u - x\|_2^2 + \gamma \|p\|_1 + \frac{\beta}{2} \|p - \nabla u\|_2^2 \quad (19)$$

with a sufficiently large penalty parameter β . We solve (19) by an alternating minimization scheme given in Algorithm 4.

Algorithm 4 (proximity of TV). Consider the following:

(1) Fixing u , compute p componentwise

$$\begin{aligned} &\min_{p_i} \{\gamma \|p_i\|_2 + (\beta/2) \|p_i - [\nabla u]_i\|_2^2\} \\ &\Rightarrow p_i = \max\{\|[\nabla u]_i\|_2 - \gamma/\beta, 0\} ([\nabla u]_i / \|[\nabla u]_i\|_2), \\ &i = 1, \dots, N \end{aligned}$$

(2) Fixing p , compute u

$$\begin{aligned} &\min_u \{(1/2) \|u - x\|_2^2 + (1/\beta) \|p - \nabla u\|_2^2\} \\ &\Rightarrow (\nabla^T \nabla + (1/\beta) I) u = (\nabla^T p + (1/\beta) x) \\ &\Rightarrow u = (\nabla^T \nabla + (1/\beta) I)^{-1} (\nabla^T p + (1/\beta) x). \end{aligned}$$

It is worth mentioning that the above alternating minimization scheme is embedded into the whole algorithm with just one time iteration, which makes the algorithm have no inner iteration.

(3) Consider $\Phi(x) = \|x\|_{H^{-1}}$. We will deduce $\text{prox}_{\gamma\Phi}(x)$ in the continuous situation, with the purpose of describing the derivation process clearly; this does not affect the processing of the numerical implementation. The proximity of $\Phi(x)$ is expressed as

$$\text{prox}_{\gamma\Phi}(x) = \arg \min_v \frac{1}{2} \|v - x\|_2^2 + \gamma \|v\|_{H^{-1}}^2. \quad (20)$$

Denote the objective function in (20) by

$$H(v) = \frac{1}{2} \|v - x\|_2^2 + \gamma \|v\|_{H^{-1}}^2, \quad (21)$$

while minimizing $H(v)$ is equivalent to minimizing $\widehat{H}(\widehat{v})$ in Fourier domain:

$$\widehat{H}(\widehat{v}) = \frac{1}{2} \|\widehat{v} - \widehat{x}\|_2^2 + \gamma \int \frac{|\widehat{v}(\xi)|^2}{(2\pi|\xi|)^2} d\xi; \quad (22)$$

minimizing $\widehat{H}(\widehat{v})$ in \widehat{v} yields the unique solution $\widehat{v} = \widehat{L}\widehat{x}$, where

$$\widehat{L}(\xi) = \frac{2(\pi|\xi|)^2}{2(\pi|\xi|)^2 + \gamma}. \quad (23)$$

Taking inverse Fourier transform, we have

$$\begin{aligned} v^* &= \text{prox}_{\gamma\Phi_1}(z) \\ &= L * z. \end{aligned} \quad (24)$$

It is indicated from (24) that the proximity of H^{-1} norm is just the Fourier domain filtering.

3.2. Solving the Proposed Model Numerically. We solve (11) by alternatively updating w, u, v in turn. The framework is given as in Algorithm 1.

In Algorithm 1, updating w is to compute the proximity of $\|\cdot\|_1$, which is just the soft thresholding operator and can be realized by (15). Updating u is to solve a TV-L1 problem, and there are many methods to deal with it; to improve the algorithm efficiency, we realize the updating using Douglas-Rachford splitting algorithm, combining the computing of $\text{prox}_{\|\cdot\|_1}$ and $\text{prox}_{\|\cdot\|_{\text{TV}}}$. Updating v is to solve an $L1-H^{-1}$ problem, which is also dealt with by Douglas-Rachford splitting algorithm, combining the soft thresholding and Fourier domain filtering. A detailed description of the whole process is shown as in Algorithm 2.

There are some issues that need to be illustrated.

TABLE 1: PSNR and SNR results for the Gamma noise removal.

Image	k	λ	ρ	Method	PSNR	SNR
Cameraman	$k = 15$	—	—	Noisy image	17.3027	5.1705
		0.18	—	AA	24.9257	12.7825
		0.8	—	Convex	25.5732	13.4701
		0.8	20	Ours	25.7435	13.5874
Lena	$k = 15$	—	—	Noisy image	17.0867	2.9056
		0.18	—	AA	27.5628	11.5464
		0.6	—	Convex	27.4350	11.0819
		0.6	20	Ours	27.7890	11.6649
Aerial	$k = 15$	—	—	Noisy image	24.2567	4.5260
		0.18	—	AA	26.8085	6.0724
		0.85	—	Convex	28.1713	6.2617
		0.85	20	Ours	28.5635	7.5391
Cameraman	$k = 10$	—	—	Noisy image	15.5091	3.3768
		0.18	—	AA	24.1401	11.9376
		0.8	—	Convex	24.3942	11.5807
		0.8	50	Ours	24.7390	11.7593
Lena	$k = 10$	—	—	Noisy image	15.2946	1.1407
		0.18	—	AA	26.6863	12.3616
		0.6	—	Convex	26.7026	12.4127
		0.6	50	Ours	27.0777	12.3698
Aerial	$k = 10$	—	—	Noisy image	22.5160	2.7469
		0.18	—	AA	25.4918	5.7500
		0.85	—	Convex	27.3225	6.1550
		0.85	50	Ours	27.7026	7.2175
Cameraman	$k = 4$	—	—	Noisy image	11.4822	-0.6501
		0.18	—	AA	22.2638	9.8301
		0.6	—	Convex	22.4161	10.6513
		0.6	80	Ours	22.6186	10.4654
Lena	$k = 4$	—	—	Noisy image	11.3537	-2.8313
		0.18	—	AA	24.4908	9.7435
		0.6	—	Convex	24.4912	9.0932
		0.6	80	Ours	24.5983	10.2984
Aerial	$k = 4$	—	—	Noisy image	18.5211	-1.2087
		0.18	—	AA	24.4714	4.6684
		0.85	—	Convex	24.5657	4.7386
		0.85	80	Ours	25.0209	5.0016

Initialization: $u^1 = g, v^1 = 0, f^1 = u^1 + v^1$,
(1) for $k = 1, 2, \dots$, do
(2) $w^{k+1} = \arg \min_w (1/2) \|w - \mu e\|_2^2 + \alpha_1 \|Gw - f^k\|_1$
(3) $u^{k+1} = \arg \min_u \alpha_1 \|Gw^{k+1} - (u + v^k)\|_1 + \alpha_2 \|u\|_{TV}$
(4) $v^{k+1} = \arg \min_v \alpha_1 \|Gw^{k+1} - (u^{k+1} + v)\|_1 + \alpha_2 \rho \|v\|_{H^{-1}}^2$
(5) $f^{k+1} = u^{k+1} + v^{k+1}$.
(6) if stopping criterion is satisfied, then
(7) stop the loop and output f^{k+1} .
(8) end for

ALGORITHM 1

Remark 5. For step (3) in Algorithm 1, when we update u as $u^{k+1} = \text{prox}_{\gamma_1 \alpha_2 \|\cdot\|_{TV}}(y^{k+1})$, we realize it by Algorithm 4 in one

Initialization: $u^{(1)} = g, v^{(1)} = 0, f = u + v, y = u, z = 0$
(1) For $k = 1, 2, \dots$, do
(2) update w using soft-thresholding method
 $w^{k+1} = G^{-1}f + \text{ST}_{\alpha_1 G}(\mu e - G^{-1}f^k)$
(3) update u
 $y^{k+1} = y^k + \lambda_1 (\text{prox}_{\gamma_1 \alpha_1 \|\cdot\|_{TV}}(2u^k - y^k) - u^k)$
 $u^{k+1} = \text{prox}_{\gamma_1 \alpha_2 \|\cdot\|_{TV}}(y^{k+1})$
(4) update v
 $z^{k+1} = z^k + \mu_1 (\text{prox}_{\gamma_2 \alpha_1 \|\cdot\|_{H^{-1}}}((Gw^{k+1} - u^{k+1})) - v^k)$
 $v^{k+1} = \text{prox}_{\gamma_2 \alpha_2 \rho \|\cdot\|_{H^{-1}}^2}(z^{k+1})$
(5) $f^{k+1} = u^{k+1} + v^{k+1}$
(6) end For, if stopping criterion is satisfied
(7) output $f = u + v$.

ALGORITHM 2



FIGURE 2: The original images: (a) Cameraman, (b) Lena, and (c) Aerial image of some city.



(a) Original image



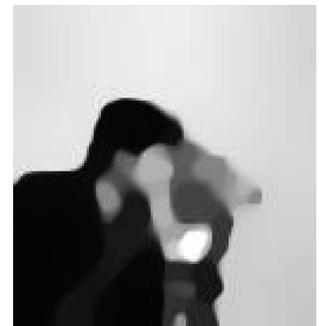
(b) Cartoon: $\rho = 100$



(c) Cartoon: $\rho = 10$



(d) Cartoon: $\rho = 1$



(e) Cartoon: $\rho = 0.1$



(f) Texture: $\rho = 100$



(g) Texture: $\rho = 10$



(h) Texture: $\rho = 1$



(i) Texture: $\rho = 0.1$

FIGURE 3: The influence of ρ on decomposition. (a) is image f . (b–e) are the cartoon u 's with different selection of ρ , which correspond to $\rho = 0.1, 1, 10$, and 100 , respectively. (f–i) are the texture v 's, which correspond to the u 's on (b–e).


 FIGURE 4: Experiment results under the corruption of Gamma noise, $K = 15$.

time iteration, which can avoid the inner loop and can improve the efficiency of the algorithm.

Remark 6. In this paper, the stopping criterion is used as follows:

$$\frac{\|f^{k+1} - f^k\|_2}{\|f^k\|_2} \leq 1e - 4. \quad (25)$$

4. Implementation Details and Experimental Results

This section is mainly devoted to numerical simulation of image restoration in the presence of multiplicative noise. We test the performance of our model under the corruption of two types of multiplicative noise: Gamma and Rayleigh.

Gamma Noise. The probability density function of Gamma noise n is given by

$$p(n) = \frac{n^{k-1}}{\Gamma(k)\theta^k} e^{-n/\theta}, \quad (26)$$

where $\theta > 0$, $k > 1$. The mean and the variance of n are

$$E(n) = k\theta, \quad (27)$$

$$V(n) = k\theta^2.$$

As multiplicative noise of an image, the mean of n is set to be 1, so $\theta = 1/k$, and the variance of n is $k\theta^2 = 1/k$.

In model (10), $w = \{1/n(i)\}_{i=1}^N$, and the value of its mean is estimated as [10]

$$\mu = E\left(\frac{1}{n}\right) = \frac{1}{\theta(k-1)} = \frac{k}{k-1}. \quad (28)$$

Rayleigh Noise. The probability density function of Rayleigh noise n is given by

$$p(n) = \frac{n}{\sigma^2} \exp\left(-\frac{n^2}{2\sigma^2}\right), \quad (29)$$

where σ is a positive parameter. The mean value of n is equal to $\sqrt{\pi/2}\sigma$, the variance of n is equal to $(2 - \pi/2)\sigma^2$, and the



FIGURE 5: Experiment results under the corruption of Gamma noise, $K = 10$.

mean value of the inverse of Rayleigh noise can be estimated as [10]

$$\mu = E\left(\frac{1}{n}\right) = \sqrt{\frac{\pi}{2\sigma^2}}. \quad (30)$$

We test experiments on three images: Cameraman (size of 256×256), Lena (size of 512×512), and Aerial image of some city (size of 512×512); see Figure 2.

In this paper, we use the signal to noise ratio (SNR) and the peak signal to noise ratio (PSNR) as the evaluation indicators. These indicators are measured between clean and restored images. Let I_{new} represent the image restored from the noisy image I_{noisy} , and let $\sigma_{I_0}^2$ represent the average variance of the clean image I_0 . Then, we define the SNR indicator by the formula as follows:

$$\text{SNR} = 10 \log_{10} \left(\frac{d\sigma_{I_0}^2}{\|I_{\text{new}} - I_0\|_2^2} \right), \quad (31)$$

and the PSNR is defined as

$$\text{PSNR} = 10 \log_{10} \left(\frac{d \|I_0\|_{\infty}^2}{\|I_{\text{new}} - I_0\|_2^2} \right), \quad (32)$$

where d denotes the length of the images.

4.1. Parameters Selection. Some necessary parameters α_1 , α_2 , ρ , λ_1 , μ_1 , γ_1 , and γ_2 are needed to be given to start up the new algorithm; α_1 , α_2 , and ρ are regularization parameters to keep balance among the terms in model (11). In particular, $\lambda = \alpha_2/\alpha_1$ is an important indicator to trade off between the data term and the regularization term. λ_1 and μ_1 are two relaxation parameters of the Douglas-Rachford splitting algorithms in Algorithm 2 and are empirically set to be 0.5. γ_1 and γ_2 are the parameters of the proximity operator of $\|\cdot\|_{\text{TV}}$ and $\|\cdot\|_{H^{-1}}$, respectively, and are empirically set to be 10.

According to the experimental analysis in [10], the ratio $\lambda = \alpha_2/\alpha_1$ in (8) is almost a constant depending on the type of noise and was empirically estimated to be 5/6, 5/8 for Gamma


 FIGURE 6: Experiment results under the corruption of Gamma noise, $K = 4$.

noise and Rayleigh noise, respectively. From a number of experiments, we find that the best value of α_2/α_1 can be chosen from the range $[0.6, 0.85]$, and the exact selection depends on the different noise levels and images. We give the values of α_2/α_1 in Table 1.

For ρ , it is a key parameter to promote the performance of model (11). Since a proper value of ρ is critical to the recovery of a satisfying image, it is necessary for us to emphatically discuss its impact on the denoising results. Here, we show its influences on image decomposition by using the image Cameraman. Fixing other parameters, we decompose the image into cartoon component and texture component with different ρ and put the experimental results in Figure 3. It is clear from Figure 3 that the different selections of ρ can bring completely different results. If ρ is selected too large ($\rho \geq 100$), there are very few contents in the texture v , and many details are found in u . With the decrease of ρ , more and more features get involved in the texture component v .

However, when ρ is selected too little (e.g., $\rho \leq 0.1$), the block effect in cartoon component u is very serious, and some major features are included in the texture component v . Moreover, in the process of denoising, the too little value of ρ may bring much more details or small scale parts appearing in v , including some noise we do not expect. For this reason, we experimentally select ρ in the interval $[20, 80]$ according to the different noise level, and the exact selection can be found in Table 1.

4.2. Experimental Results in the Assumption of Gamma Noise. In the assumption of Gamma noise corruption, we test denoising performances of AA model, convex model, and the proposed model on the images of Figure 2 and compare the SNRs/PSNRs of three methods. The AA model is formulated as

$$\min_f \int_{\Omega} \left(\log f + \frac{g(x)}{f(x)} \right) dx + \lambda \|\nabla f\|_1, \quad (33)$$



FIGURE 7: Experiment results under the corruption of Rayleigh noise with the mean 1 and the variance 0.2732.

which is solved by the gradient descent method

$$\frac{f_{i+1} - f_i}{\Delta t} = \operatorname{div} \left(\frac{\Delta f_i}{|\Delta f_i|} \right) + \lambda \frac{g - f_i}{f_i^2}, \quad i = 1, 2, \dots, \quad (34)$$

where Δt is step size and is set as 0.01 for all experiments and λ is the regularization parameter and is adjusted according to the images and noise level; the exact value can be found in Table 1. The convex model is solved by the ADMM method; see [10].

In the test, the original images are corrupted by Gamma noise with $k = 4$, $k = 10$, and $k = 15$, respectively. In Table 1, we list values of related parameters and obtained SNRs/PSNRs by taking average of ten noise cases under the same experimental setting. It is clear that the proposed model performs quite better in terms of PSNR values and SNR values than AA model and convex model.

We further show the denoised results obtained by three methods in Figures 4–6. It is observed that when $k = 15$ and $k = 10$, as shown in Figures 4 and 5, the restored images

from the new model possess more clear contents and much better visual effects than the AA model and the convex model. However, when $k = 4$, that is, the noise level is higher, we can observe from Figure 6 that the new model possesses little advantage over the convex model in the visual effect. We also observe from Table 1 that, when $k = 4$, the SNRs/PSNRs of our model and the convex model are neck to neck. The main reason for this phenomenon is that the larger the variance of the noise is, the more the texture components are seriously destroyed and regarded as noise part, and the bigger value ρ should be adjusted. As described in the analysis (c) of the new model, there will be a little content in the v parts, and denoising results of the new model will approximate that of the convex model.

4.3. Experimental Results in the Assumption of Rayleigh Noise. The Rayleigh noise is generated by $\sigma \sqrt{-2 \log(1 - b)}$, where b is a uniformly distributed random variable generated by “rand” in MATLAB. The mean of Rayleigh noise is set to be 1, and the variance is 0.2732.



FIGURE 8: Experiment results of the proposed model. From (a–l), (a, e, i) are the noisy images corrupted by Gamma noise ($K = 10$), (b, f, j) are the restored cartoon parts, (c, g, k) are the restored texture parts, and (d, h, l) are the whole recovery images.

Since AA model is deduced under the assumption of Gamma noise, we just test the denoising performance of the convex model and the proposed model. Table 2 shows the SNRs/PSNRs results of the two models. We notice that the proposed model gets higher SNRs and PSNRs than the convex model. The experiment results are shown as in Figure 7. It can be observed that the restored images from the new model possess more clear contents and much better visual effects.

4.4. Further Analysis of the Restored Images. In this section, we further analyze the cartoon part u and the texture part v of the restored image f of model (11). Take the case under Gamma noise as an example; we exhibit the experiment results when $k = 10$.

It can be observed from Figure 8 that there are rich contents in the texture parts v , which enhance the image quality in both the SNRs/PSNRs index and the visual effects. So,

TABLE 2: PSNR and SNR results for the Rayleigh noise removal.

Image	Method	PSNR	SNR
Cameraman	Noisy image	11.1192	-1.013
	Convex	22.4597	10.3275
	Ours	22.6852	10.5529
Lena	Noisy image	10.9307	-3.2543
	Convex	24.6532	10.4683
	Ours	25.2379	11.0529
Aerial	Noisy image	18.1219	-1.6079
	Convex	25.0408	5.3110
	Ours	25.0688	5.3390

the proposed model can also be regarded as a cartoon-texture decomposition method under the multiplicative noise corruption, on which we need further research.

5. Conclusions

A novel cartoon-texture decomposition based multiplicative noise removal model has been proposed in this paper. The main advantage of our work is embodied in the following two aspects: the one in which we take most of the a priori information of texture component into the denoising model, which makes the model possess more flexibility and efficiency. Another is that we dexterously solve the new convex model using optimal splitting algorithms and Fourier domain filtering method. The experiment results demonstrate that the proposed model can better remove the noise than the two other models, and the recovery images have higher SNRs/PSNRs and better visual effects.

Competing Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This paper is supported by the National Natural Science Foundation of China under Grant nos. 61472303, 61271294, 61362129, and 61379030 and by the Fundamental Research Funds for the Central Universities (no. NSIY21).

References

- [1] L. Rudin, P.-L. Lions, and S. Osher, "Multiplicative denoising and deblurring: theory and algorithms," in *Geometric Level Set Methods in Imaging, Vision, and Graphics*, pp. 103–119, Springer, New York, NY, USA, 2003.
- [2] G. Aubert and J.-F. Aujol, "A variational approach to removing multiplicative noise," *SIAM Journal on Applied Mathematics*, vol. 68, no. 4, pp. 925–946, 2008.
- [3] J. Shi and S. Osher, "A nonlinear inverse scale space method for a convex multiplicative noise model," *SIAM Journal on Imaging Sciences*, vol. 1, no. 3, pp. 294–321, 2008.
- [4] Y.-M. Huang, M. K. Ng, and Y.-W. Wen, "A new total variation method for multiplicative noise removal," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 20–40, 2009.
- [5] S. Durand, J. Fadili, and M. Nikolova, "Multiplicative noise removal using L1 fidelity on frame coefficients," *Journal of Mathematical Imaging and Vision*, vol. 36, no. 3, pp. 201–226, 2010.
- [6] Y. Hao, X. Feng, and J. Xu, "Multiplicative noise removal via sparse and redundant representations over learned dictionaries and total variation," *Signal Processing*, vol. 92, no. 6, pp. 1536–1549, 2012.
- [7] Y.-M. Huang, L. Moisan, M. K. Ng, and T. Zeng, "Multiplicative noise removal via a learned dictionary," *IEEE Transactions on Image Processing*, vol. 21, no. 11, pp. 4534–4543, 2012.
- [8] T. Teuber and A. Lang, "Nonlocal filters for removing multiplicative noise," in *Scale Space and Variational Methods in Computer Vision*, A. M. Bruckstein, B. M. ter Haar Romeny, A. M. Bronstein, and M. M. Bronstein, Eds., vol. 6667 of *Lecture Notes in Computer Science*, pp. 50–61, Springer, Berlin, Germany, 2012.
- [9] F. Dong, H. Zhang, and D.-X. Kong, "Nonlocal total variation models for multiplicative noise removal using split Bregman

iteration," *Mathematical and Computer Modelling*, vol. 55, no. 3-4, pp. 939–954, 2012.

- [10] X.-L. Zhao, F. Wang, and M. K. Ng, "A new convex optimization model for multiplicative noise and blur removal," *SIAM Journal on Imaging Sciences*, vol. 7, no. 1, pp. 456–475, 2014.
- [11] W. Yin, D. Goldfarb, and S. Osher, "The total variation regularized model for multiscale decomposition," *Multiscale Modeling and Simulation*, vol. 6, pp. 190–211, 2007.
- [12] A. Chambolle, "An algorithm for total variation minimization and applications," *Journal of Mathematical Imaging and Vision*, vol. 20, no. 1-2, pp. 89–97, 2004.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

