Hindawi
Journal of Sensors
Volume 2021, Article ID 8864906, 15 pages
https://doi.org/10.1155/2021/8864906

Research Article

A Nonlinear Gradient Domain-Guided Filter Optimized by Fractional-Order Gradient Descent with Momentum RBF Neural Network for Ship Image Dehazing

Qionglin Fang and X. U. E. Han

School of Navigation, Jimei University, Xiamen 361021 Fujian, China

Correspondence should be addressed to X. U. E. Han; imlmd@163.com

Received 3 August 2020; Accepted 17 December 2020; Published 4 January 2021

Academic Editor: Giovanni Diraco

To avoid the blurred edges, noise, and halos caused by guided image filtering algorithm, this paper proposed a nonlinear gradient domain-guided image filtering algorithm for image dehazing. To dynamically adjust the edge preservation and smoothness of dehazed images, this paper proposed a fractional-order gradient descent with momentum RBF neural network to optimize the nonlinear gradient domain-guided filtering (NGDGIF-FOGDMRBF). Its convergence is proved. In order to speed up the convergence process, an adaptive learning rate is used to adjust the training process reasonably. The results verify the theoretical results of the proposed algorithm such as its monotonicity and convergence. The descending curve of error values by FOGDM is smoother than gradient descent and gradient descent with momentum method. The influence of regularization parameter is analyzed and compared. Compared with dark channel prior, histogram equalization, homomorphic filtering, and multiple exposure fusion, the halo and noise generated are significantly reduced with higher peak signal-to-noise ratio and structural similarity index.

1. Introduction

Marine accidents caused by visibility account for a considerable proportion of marine meteorological accidents. Effective image defogging methods can protect the safety of ship navigation [1]. Based on the theory of dark channel, Dong proposed an improved algorithm to remove fog outside the ship cabin [2]. It used the method of guided filtering instead of soft matting to optimize the transmittance and improved the estimation method of atmospheric light intensity. However, when the fog is very thick or the picture itself is white and there is no shadow coverage, the dark channel theory cannot be applied. Wu et al. used the dark channel prior defogging algorithm to reduce the influence of cloud cover on target recognition and improve the recognition effect of marine ships under complex weather conditions [3]. However, it has no obvious effect on large targets. Li et al. proposed a method based on the prior fusion of retinex and dark channel to enhance the defogging of sea cucumber images [4]. However, the disadvantage of this method is the increased processing time.

Among image dehazing methods, guided image filtering (GIF) is a kind of image filtering technology. It filters the input target image based on a guided graph. However, GIF also has some disadvantages. For example, its regularization coefficient in the objective function is fixed and does not change with the spatial position of the image. The detail layer adopts the fixed gain method and may cause additional noise in the background. Therefore, gradient identification has been used for improving the dada filter method [5]. Ding proposed maximum likelihood recursive identification for the multivariate equation-error autoregressive moving average systems using the data filtering [5]. Liu et al. proposed a gradient domain-guided image filtering, which contained a

clear first-order edge perception constraint for better preserving edges [6]. Khou et al. proposed a probability pansharpening method based on gradient domain-guided image filtering to fuse panchromatic and multispectral images [7]. Zhuang et al. proposed a new MRI reconstruction algorithm based on edge-preserving filtering [8]. Zhuang et al. introduced side window guided filtering [9]. Yin et al. proposed a novel edge-preserving smoothing algorithm for multiscale exposure fusion to preserve the detail information of the brightest or darkest region [10].

Therefore, taking into account the above analysis, this paper proposed a fractional-order gradient descent with momentum RBF neural network to optimize gradient domain-guided filtering. Neural network itself has some shortcomings, such as slow learning speed, easy to fall into local extremum, and learning and memory instability. The weights of the neural network are obtained by training the network. Gradient descent (GD) is a basic method for updating and optimizing the weights of neural networks [11, 12]. Standard GD has two main shortcomings: slow training speed and easy to fall into local optimal solution. On the basis of GD, stochastic gradient descent (SGD) divides the total data into several small batches according to the data distribution and updates the parameters with small batches of data [13]. The calculation time of each update step does not depend on the number of training samples. It can converge with very large number of training samples. However, it is difficult to choose a suitable learning rate. To avoid the high variance oscillation of SGD, momentum method was proposed [14]. It simulated the inertia of moving objects, by considering the former related training directions and weakening the irrelevant directions. Momentum can accelerate the learning in the related direction, restrain the oscillation, and accelerate the convergence, especially when dealing with high curvature, small but consistent gradient or gradient with noise. However, it is difficult to choose a better learning rate.

To solve this problem, the gradient descent with momentum (GDM) was proposed [15]. By accumulating the past gradient values, the fluctuation on the path to the optimal value is reduced and the convergence is accelerated. GDM can accelerate learning when the direction of the current and past gradients is uniform, while restrain oscillation when the direction of current and past gradients is inconsistent.

Fractional-order calculus, proposed in 1965, is an important branch of mathematics and has successfully demonstrated to have more advantages than integer-order calculus in the field of neural network. It has faster response speed and smaller buffeting effect [16]. Kinh et al. proposed a fractional gradient descent method for training BP neural networks [17]. Wang et al. proposed a new fractional gradient descent learning algorithm for radial basis function neural networks [18]. Khan et al. proposed a fractional gradient descent method for training BP neural networks [19].

However, there is still no fractional gradient descent with momentum algorithm for training neural network. This paper proposed fractional-order gradient descent with momentum method for training RBF neural network. The aim of the paper is to improve the guided filter for defogging ship images. The algorithm is design for avoiding the blurred edges, noise, and halos caused by the original guided image filtering algorithm. The proposed method estimates the optimal parameters of nonlinear gradient domain-guided image filter. The main contributions of this paper are as follows:

(1) A nonlinear gradient domain-guided image filtering algorithm was proposed

(2) The optimal value of model parameters is proved

(3) Fractional-order calculus is applied to gradient descent with momentum algorithm for training neural network. The new algorithm is used to adjust the weights of the neural network to improve its learning speed and performance for optimizing gradient domain-guided filtering

(4) The convergence of the FOGDM-RBF is proved

2. Related Work

2.1. Atmospheric Scattering Model. The atmospheric scattering model can be defined as [1]

$$I(x) = J(x)t(x) + A(1 - t(x)), \tag{1}$$

where $I(x)$ denotes the observed hazy image, $J(x)$ denotes the clean image to be recovered, $t(x)$ denotes the the transmittance, A denotes the global atmospheric light, and $t(x)$ is defined as [1]

$$t(x) = e^{-\beta d(x)}, \tag{2}$$

where $d(x)$ denotes the distance between the camera to the image scene and β denotes the scattering coefficient.

2.2. Gradient Domain-Guided Filtering. In the gradient domain-guided filtering algorithm, the filtered image q is assumed to be a linear transform of the guidance image I in the window Ω [2]:

$$q = aI + b, \tag{3}$$

where$a$, b are two constants in the window. The cost function is defined as [2]

$$E = \sum_{i \in \Omega} \left[(aI_i + b - p_i)^2 + \frac{\lambda}{\Gamma} (a - \gamma)^2 \right], \tag{4}$$

where p denotes the input image to be filtered. Denote the regularization parameter as $\varepsilon = \lambda / \Lambda$. Λ is the edge-aware weighing parameter, λ is the regularization parameter, and γ is a coefficient for distinguishing edges from smooth areas defined as [2]

$$\gamma = 1 - \frac{1}{1 + e^{\eta(p-\mu)}}, \tag{5}$$

where μ is the mean value of all p pixels and η is calculated as [2]

$$\eta = \frac{4}{\mu - \min(p)}, \tag{6}$$

2.3. RBF Neural Network. In 1985, Powell proposed a radial basis function for multivariate interpolation. RBF neural network is a three-layer neural network, which includes an input layer, hidden layer, and output layer. A radial basis function is used to form the hidden layer space, so that the input vector can be directly mapped to the hidden space without weight connection. When the center point of RBF is determined, the mapping relationship is determined. The mapping from the hidden layer space to the output space is linear. The output of the network is the linear-weighted sum of the output of the hidden unit, and the weight is the adjustable parameter of the network.

The mapping from input to output of the network is nonlinear, while the output of the network is linear to the adjustable parameters. The weight of the network can be directly solved by the linear equations, thus greatly speeding up the learning speed and avoiding the local minimum problem.

The neural network is used to determine the optimal parameters of the nonlinear-guided filter, so as to minimize the difference between the input image p and the output image I. The output image retains the overall characteristics of the input image, and the filtering results can fully obtain the change details of the guide image.

The most frequently used radial basis function is Gauss function:

$$\varphi_i(x) = e^{-\left(\|x-\mu_i\|^2\right)/2\sigma^2}, \tag{7}$$

$$\varphi = [\varphi_1, \varphi_2, \cdots, \varphi_P], \tag{8}$$

where x is the input vector, $\|x\|$ denotes the Euclidean Norm of x, φ_i denotes the radial basis function, x_i denotes the central vector of the function, σ_j denotes the width of the radial basis function, μ_i denotes the threshold vector, P denotes the number of hidden layer nodes, N denotes the number of input training samples, and y denotes the output of the neural network.

$$y = W(n)\varphi(x(n)), \tag{9}$$

$$W(n) = [w_1(n), w_2(n), \cdots, wp(n)]. \tag{10}$$

2.4. Fractional-Order Calculus. The Riemann-Liouville definition of fractional calculus is as follows:

Definition 1. For the absolute integrable function $x(t)$ in the interval $[t_0, t]$, its Riemann-Liouville integral is as follows:

$${}_{t_0}D_t^{\alpha}x(t) = \frac{1}{\Gamma(\alpha)}\int_{t_0}^{t}(t-\tau)^{\alpha-1}x(\tau)d\tau, \tag{11}$$

where the real part of α is positive and $\Gamma(x)$ is the gamma function.

Definition 2. For the absolute integrable function $x(t)$ in the interval $[t_0, t]$, its Riemann-Liouville differential is defined as

$${}_{t_0}D_t^{\alpha}x(t) = \frac{d^m}{dt^n}\left[\frac{1}{\Gamma(m-\alpha)}\int_{t_0}^{t}(t-\tau)^{m-\alpha-1}x(\tau)d\tau\right], \tag{12}$$

where $\alpha \in [m-1, m)$ and m is a positive integer.

Lemma 3. *For the function $f(x) = (x - x_0)^v$, when $0 \le m \le \alpha < m + 1$, the following formula holds:*

$${}_{x_0}D_x^{\alpha}f(x) = \frac{\Gamma(v+1)}{\Gamma(v-\alpha+1)}(x-x_0)^{v-\alpha}. \tag{13}$$

3. Main Results

3.1. Nonlinear Gradient Domain-Guided Filter. We suppose the filtered image q is a nonlinear transform of the guidance image I:

$$q = aI^{\alpha} + b, \tag{14}$$

where α is the exponent. In order to avoid gradient reversion, we also set the following restrictive conditions:

$$1 \le \alpha \le 2, \tag{15}$$

when $\alpha = 1$, (14) will degenerate into (3). Therefore, the gradient domain-guided filtering is a special case of the newly proposed nonlinear gradient domain-guided filtering algorithm. The optimization of guided filter is to search out optimal factors a and b, so as to minimize the difference between the input image p and the output image I.

Theorem 4. *The optimal values of a and b are computed as*

$$a_k = \frac{(1/|w_k|)\sum_{i\in w_k}p_iI_i^{\alpha} - (1/|w_k|)\sum_{i\in w_k}p_i\sum_{i\in w_k}I_i^{\alpha} + (\lambda\gamma/\Lambda)}{(1/|w_k|)\left[\sum_{i\in w_k}I_i^{2\alpha} - \left(\sum_{i\in w_k}I_i^{\alpha}\right)^2\right] + (\lambda/\Lambda)}, \tag{16}$$

$$b_k = \sum_{i\in w_k}p_i - a_k\sum_{i\in w_k}I_i^{\alpha}, \tag{17}$$

where $|w_k|$ is the number of pixels in the window.

Proof. The noise is defined as

$$n = q - p. \tag{18}$$

Substituting (14) into (18) yields

$$n = aI^{\alpha} + b - p. \tag{19}$$

The ultimate goal is to minimize this noise. Therefore, the

cost function can be written as

$$\arg\min E_k = \arg\min \sum_{i\in w_k}\left[(a_k I_i^\alpha + b_k - p_i)^2 + \frac{\lambda}{\Lambda}(a_k-\gamma)^2\right], \tag{20}$$

where w_k is the window centered at pixel k.

Seeking partial derivative of network parameters can obtain

$$\frac{\partial E_k}{\partial a_k} = \sum_{i\in w_k}\left[2(a_k I_i^\alpha + b_k - p_i)I_i^\alpha + \frac{2\lambda}{\Lambda}(a_k-\gamma)\right] = 0,$$
$$\frac{\partial E_k}{\partial b_k} = \sum_{i\in w_k} 2(a_k I_i^\alpha + b_k - p_i) = 0. \tag{21}$$

Therefore, the following results hold:

$$a_k = \frac{\sum_{i\in w_k} p_i I_i^\alpha - b_k\sum_{i\in w_k} I_i^\alpha + |w_k|(\lambda\gamma/\Lambda)}{\sum_{i\in w_k}\left(I_i^{2\alpha}+\lambda/\Lambda\right)},$$
$$b_k = \sum_{i\in w_k} p_i - a_k\sum_{i\in w_k} I_i^\alpha. \tag{22}$$

The proof of this theorem relays in the same principles as ordinary least squares. Thus, this completes the proof of Theorem 4. Next, the above model will be applied to the entire image filtering window. But each pixel is contained in multiple windows. For example, if a 3*3 window filter is used, every point except the edge area will be included in nine windows. Therefore, for different windows, we will get $|w_k|$ numbers of q_i value.

Denote

$$\bar{a}_i = \frac{1}{|w|}\sum_{k:i\in w_k} a_k, \tag{23}$$

$$\bar{b}_i = \frac{1}{|w|}\sum_{k:i\in w_k} b_k. \tag{24}$$

The final result is obtained by averaging all the q_i values.

$$q_i = \frac{1}{|w|}\sum_{k:i\in w_k}(a_k I_i^\alpha + b_k) = \bar{a}_i I_i^\alpha + \bar{b}_i. \tag{25}$$

Thus, the mapping from I to q for each pixel has been established.

3.2. Proposed FOGDM-RBF Algorithm. Define $d(n)$ as the expected output response of the n times of iterative of the neural network. The error signal is defined as

$$e(n) = d(n) - y(n) = d(n) - W(n)\varphi(x(n)). \tag{26}$$

Define the objective function as

$$J(n) = \frac{1}{2}e^2(n) = \frac{1}{2}[d(n) - W(n)\varphi(x(n))]^2. \tag{27}$$

Denote

$$\Delta w(n+1) = w(n+1) - \Delta w(n). \tag{28}$$

According to the gradient descent with momentum algorithm, one can obtain

$$\Delta w(n+1) = \gamma_n \Delta w(n) - \eta_c D^\alpha_{w(n)} J(n), \tag{29}$$

$$0 < \alpha < 1, \tag{30}$$

where is $\eta > 0$ the learning rate and γ_n is the momentum coefficient designed as below:

$$\gamma_n = \begin{cases} \dfrac{\gamma\left\|{}_cD^\alpha_{w(n)}J(n)\right\|}{\|\Delta w(n)\|}, & \|\Delta w(n)\| \neq 0, \\ 0, & \text{else}, \end{cases} \tag{31}$$

where $\gamma \in (0,\eta)$ is the momentum factor and $\|\cdot\|$ is the Euclidian norm. According to the definition of Caputo fractional derivative, one can obtain

$${}_cD^\alpha_{w_i(n)} J(n) = \sum_{j=1}^{N} J_j'(w\varphi(xj(n)))\varphi(xj(n))\frac{(w_i(n)-c)^{1-\alpha}}{(1-\alpha)\Gamma(1-\alpha)}. \tag{32}$$

3.3. Convergence Analysis of FOGAM-RBF. The following assumptions are given:

(A1) $|\varphi|, |\varphi'|, |\varphi''|, |y|, |y'|, |y''|$ are uniformly bounded:

$$\lambda_1 = \max\left\{|\varphi|, |\varphi'|, |\varphi''|, |y|, |y'|, |y''|\right\} \tag{33}$$

(A2) w are uniformly bounded:

$$\lambda_2 = \max_{n>0}\|w(n)\| \tag{34}$$

This condition can be easily satisfied since the most common Gauss function is uniformly bounded and differentiable.

Theorem 5. *Assume (A1) and (A2) are valid, and*

$$(\eta-\gamma)(1-\alpha)\Gamma(1-\alpha)(\lambda_2 - c)^{\alpha-1} \geq NP\lambda_1^3(\eta^2+\gamma^2). \tag{35}$$

Then, the following results hold:

$$J(w(n+1)) \leq J(w(n)), \tag{36}$$

$$\lim_{n\to+\infty} J(w(n)) = J*, \tag{37}$$

$$\lim_{n\to+\infty}\left\|{}_cD^\alpha_{w(n)}J(w(n))\right\| = 0. \tag{38}$$

Proof. The first step is to prove (36) which means $J(w(n))$ is decreasing. The second step is to prove (37) which means $J(w(n))$ is convergent. The third step is to prove (38) which means $\|{}_cD^{\alpha}_{w(n)}J(w(n))\|$ approaches zero. By using the Taylor mean value theorem with Lagrange remainder, one can obtain:

$$\begin{aligned}
J(w(n+1)) - J(w(n)) &= \sum_{i=1}^{N}[J_i(w(n+1)\varphi(x_i)) - J_i(w(n)\varphi(x_i))] \\
&= \sum_{i=1}^{N} J_i'(w(n)\varphi(x_i))[w(n+1)\varphi(x_i) - w(n)\varphi(x_i)] \\
&\quad \times \frac{1}{2}\sum_{i=1}^{N} J_i'(w(n)\varphi(x_i))[w(n+1)\varphi(x_i) - w(n)\varphi(x_i)]^2 \\
&= \sum_{i=1}^{N} J_i'(w(n)\varphi(x_i))w(n+1)\varphi(x_i) \\
&\quad + \frac{1}{2}\sum_{i=1}^{N} J_i'(w(n)\varphi(x_i))\Delta w^2(n+1)\varphi^2(x_i).
\end{aligned} \tag{39}$$

From (32), one can obtain that if $\forall 1 \le j \le P$, $(w_j(n) - c)^{1-\alpha} \ne 0$, the following equation holds:

$$\begin{aligned}
&\sum_{j=1}^{N} J_j'\left(w\varphi\left(x_j(n)\right)\right)\varphi\left(x_j(n)\right) \\
&\quad = (1-\alpha)\Gamma(1-\alpha)(w(n)-c)^{\alpha-1}\,{}_cD^{\alpha}_{w(n)}J(n).
\end{aligned} \tag{40}$$

Substituting (40) into the first item of (39) yields

$$\begin{aligned}
&\sum_{j=1}^{N} J_j'\left(w\varphi\left(x_j(n)\right)\right)\Delta w(n+1)\varphi\left(x_j(n)\right) \\
&\quad = (1-\alpha)\Gamma(1-\alpha)(w(n)-c)^{\alpha-1}.\Delta w(n+1)\,{}_cD^{\alpha}_{w(n)}J(n).
\end{aligned} \tag{41}$$

From (29), one can obtain

$$\begin{aligned}
\Delta w(n+1)\,{}_cD^{\alpha}_{w(n)}J(n) &= \left[\gamma_n\Delta w(n) - \eta_c D^{\alpha}_{w(n)}J(n)\right]{}_cD^{\alpha}_{w(n)}J(n) \\
&= \gamma_n\Delta w(n)\,{}_cD^{\alpha}_{w(n)}J(n) - \eta\left\|{}_cD^{\alpha}_{w(n)}J(n)\right\|^2 \\
&\le \|\gamma_n\| \cdot \|\Delta w(n)\| \cdot \left\|{}_cD^{\alpha}_{w(n)}J(n)\right\| - \eta\left\|{}_cD^{\alpha}_{w(n)}J(n)\right\|^2.
\end{aligned} \tag{42}$$

Substituting (31) into (42) yields

$$\begin{aligned}
\Delta w(n+1)\,{}_cD^{\alpha}_{w(n)}J(n) &\le \frac{\gamma\left\|{}_cD^{\alpha}_{w(n)}J(n)\right\|}{\|\Delta w(n)\|} \cdot \|\Delta w(n)\| \\
&\quad \cdot \left\|{}_cD^{\alpha}_{w(n)}J(n)\right\| - \eta\left\|{}_cD^{\alpha}_{w(n)}J(n)\right\|^2 = \gamma\left\|{}_cD^{\alpha}_{w(n)}J(n)\right\|^2 \\
&\quad - \eta\left\|{}_cD^{\alpha}_{w(n)}J(n)\right\|^2 - (\eta-\gamma)\left\|{}_cD^{\alpha}_{w(n)}J(n)\right\|^2.
\end{aligned} \tag{43}$$

Substituting (43) into (41) yields

$$\begin{aligned}
&\sum_{j=1}^{N} J_j'\left(w\phi\left(x_j(n)\right)\right)\Delta w(n+1)\phi\left(x_j(n)\right) \\
&\quad \le -(\eta-\gamma)(1-\alpha)\Gamma(1-\alpha)(w(n)-c)^{\alpha-1} \cdot \left\|{}_cD^{\alpha}_{w(n)}J(n)\right\|^2.
\end{aligned} \tag{44}$$

From (34), one can obtain

$$-(w-c)^{\alpha-1} = -\frac{1}{(w-c)^{1-\alpha}} \le -\frac{1}{(\lambda_2-c)^{1-\alpha}} = -(\lambda_2-c)^{\alpha-1}. \tag{45}$$

Substituting (45) into (44) yields

$$\begin{aligned}
&\sum_{j=1}^{N} J_j'\left(w\phi\left(x_j(n)\right)\right)\Delta w(n+1)\phi\left(x_j(n)\right) \\
&\quad \le -(\eta-\gamma)(1-\alpha)\Gamma(1-\alpha)(\lambda_2-c)^{\alpha-1}\left\|{}_cD^{\alpha}_{w_i(n)}J(n)\right\|^2.
\end{aligned} \tag{46}$$

If $\exists 1 \le j \le P$, $(w_j(n) - c)^{1-\alpha} = 0$, without loss of generality, assume that $(w_1(n) - c)^{1-\alpha} = 0$.

From (32), one can obtain

$${}_cD^{\alpha}_{w_1(n)}J(n) = 0. \tag{47}$$

Substituting (37) into (44) yields

$$\begin{aligned}
&\sum_{j=1}^{N} J_j'\left(w\phi\left(x_j(n)\right)\right)\Delta w(n+1)\phi\left(x_j(n)\right) \\
&\quad \le 0 - (\eta-\gamma)(1-\alpha)\Gamma(1-\alpha)(w_2(n)-c)^{\alpha-1} \\
&\qquad \cdot \left\|{}_cD^{\alpha}_{w_2(n)}J(n)\right\|^2 - \cdots - (\eta-\gamma)(1-\alpha)\Gamma(1-\alpha) \\
&\qquad \cdot (w_P(n)-c)^{\alpha-1}\left\|{}_cD^{\alpha}_{w_P(n)}J(n)\right\|^2 \\
&\quad \le -(\eta-\gamma)(1-\alpha)\Gamma(1-\alpha)(\lambda_2-c)^{\alpha-1}\left\|{}_cD^{\alpha}_{w(n)}J(n)\right\|^2.
\end{aligned} \tag{48}$$

From (46) and (48), we have

$$\begin{aligned}
&\sum_{j=1}^{N} J_j'\left(w\phi\left(x_j(n)\right)\right)\Delta w(n+1)\phi\left(x_j(n)\right) \\
&\quad \le -(\eta-\gamma)(1-\alpha)\Gamma(1-\alpha)(\lambda_2-c)^{\alpha-1}\left\|{}_cD^{\alpha}_{w_i(n)}J(n)\right\|^2.
\end{aligned} \tag{49}$$

Input: Guidance image I and the input image p.
Output: Filtered image q.
1: Initialization of parameters, such as window size and regularization parameter
2: Initialization of the neural network model
3: Initialization of neural networks with weights
4: it = 0
5: while it < maxIT & training error does not meet the learning requirements
6: Calculate the activation function according to (7)
7: Calculate the output of neural network according to (9)
8: Calculate the output error of the neural network according to (29)
9: Calculate the objective function according to (27)
10: Calculate Caputo fractional derivative of objective function according to (32)
11: Update the momentum coefficient according to (31)
12: Update the weight of the neural network according to (29)
13: it = it + 1
14: end while
15:Use the obtained neural network weights and the input testing data set to calculate the optimal parameters
16: for window w_k
17: Calculate coefficient η according to equation (6)
18: Calculate coefficient γ according to equation (5)
19: Calculate a_k according to equation (16)
20: Calculate b_k according to equation (17)
21: end k
22: for each pixel i
23: Calculate the mean values of a_k according to equation (23)
24: Calculate the mean values of b_k according to equation (24)
25: Calculate the filtered image pixel q_i according to equation (25)
26: end i
27: return filtered image q.

Algorithm 1. Algorithm process. FOGDM-RBF for optimizing gradient domain-guided filter

From (29), one can obtain

$$\|\Delta w(n+1)\|^2 = \left\|\gamma_n \Delta w(n) - \eta_c D^{\alpha}_{w(n)} J(n)\right\|^2 \le 2\gamma_n^2 \|\Delta w(n)\|^2 + 2\eta^2 \left\|{}_c D^{\alpha}_{w(n)} J(n)\right\|^2. \tag{50}$$

Substituting (31) into (50) yields

$$\begin{aligned}\|\Delta w(n+1)\|^2 &\le 2\frac{\gamma^2{}_c|\,|D^{\alpha}_{w(n)} J(n)\right\|^2}{\|\Delta w(n)\|^2}\|\Delta w(n)\|^2 \\ &\quad + 2\eta^2 \left\|{}_c D^{\alpha}_{w(n)} J(n)\right\|^2 \\ &= 2\left(\gamma^2+\eta^2\right)\left\|{}_c D^{\alpha}_{w(n)} J(n)\right\|^2.\end{aligned} \tag{51}$$

Substituting (51) into the second item of (39) yields

$$\begin{aligned}&\frac{1}{2}\sum_{i=1}^{N} J_i'(w(n)\phi(x_i))\Delta w^2(n+1)\phi^2(x_i) \\ &\le \sum_{i=1}^{N} J_i'(w(n)\phi(x_i))\left(\gamma^2+\eta^2\right)\left\|{}_c D^{\alpha}_{w(n)} J(n)\right\|^2 \phi^2(x_i).\end{aligned} \tag{52}$$

From (33), one can obtain

$$\|\varphi(x_i)\| \le \sqrt{P\lambda_1}. \tag{53}$$

Substituting (53) into (52) yields

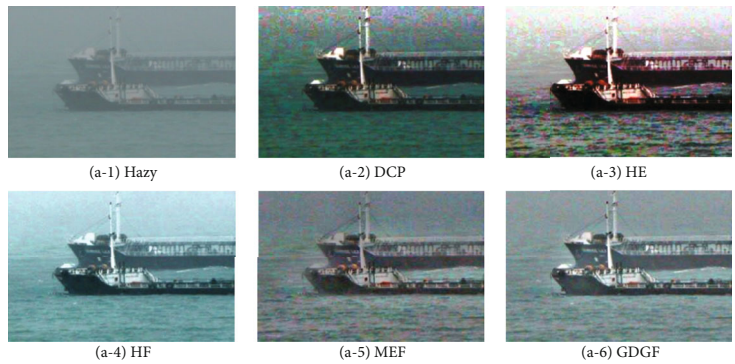
$$\begin{aligned}&\frac{1}{2}\sum_{i=1}^{N} J_i'(w(n)\phi(x_i))\Delta w^2(n+1)\phi^2(x_i) \\ &\le P\lambda_1^2 \sum_{i=1}^{N} J_i'(w(n)\phi(x_i))\left(\gamma^2+\eta^2\right)\left\|{}_c D^{\alpha}_{w(n)} J(n)\right\|^2.\end{aligned} \tag{54}$$

Substituting (33) into (54) yields

$$\begin{aligned}&\frac{1}{2}\sum_{i=1}^{N} J_i'(w(n)\varphi(x_i))\Delta w^2(n+1)\varphi^2(x_i) \\ &\le NP\lambda_1^3\left(\gamma^2+\eta^2\right)\left\|{}_c D^{\alpha}_{w(n)} J(n)\right\|^2.\end{aligned} \tag{55}$$

Substituting (55) into (39) yields

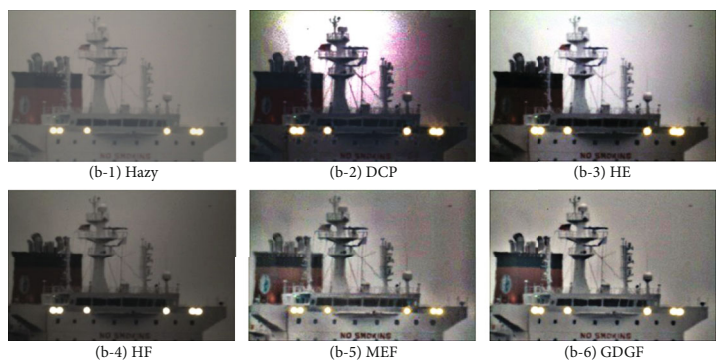
$$\begin{aligned}J(w(n+1)) - J(w(n)) &\le \sum_{i=1}^{N} J_i'(w(n)\varphi(x_i))\Delta w(n+1)\varphi(x_i) \\ &\quad + NP\lambda_1^3\left(\gamma^2+\eta^2\right)\left\|{}_c D^{\alpha}_{w(n)} J(n)\right\|^2.\end{aligned} \tag{56}$$



(a-1) Hazy (a-2) DCP (a-3) HE

(a-4) HF (a-5) MEF (a-6) GDGF

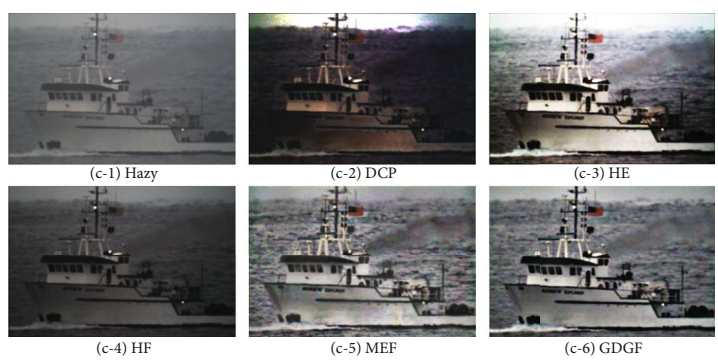
(a)



(b-1) Hazy (b-2) DCP (b-3) HE

(b-4) HF (b-5) MEF (b-6) GDGF

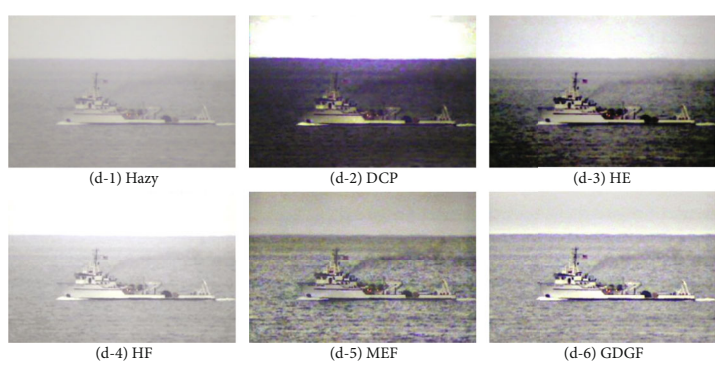
(b)



(c-1) Hazy (c-2) DCP (c-3) HE

(c-4) HF (c-5) MEF (c-6) GDGF

(c)

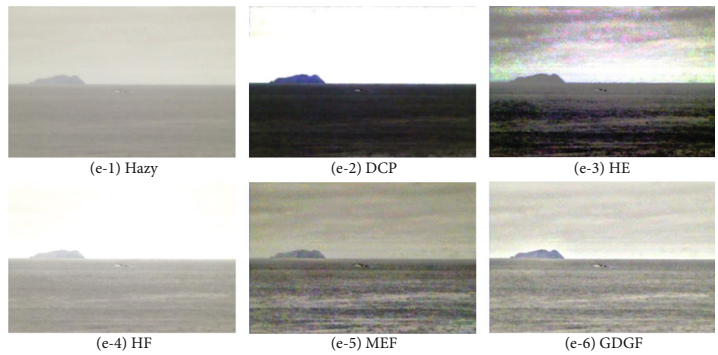


(d-1) Hazy (d-2) DCP (d-3) HE

(d-4) HF (d-5) MEF (d-6) GDGF

(d)

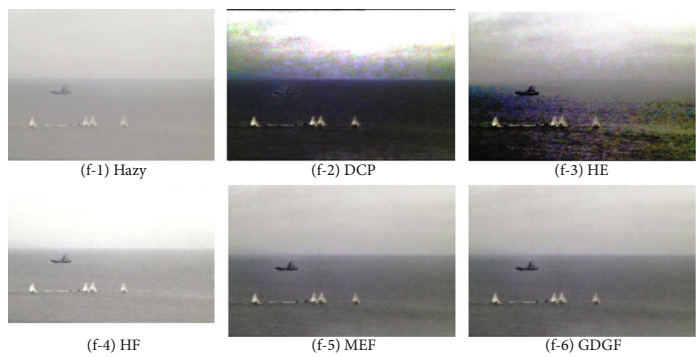
Figure 1: Continued.



(e-1) Hazy (e-2) DCP (e-3) HE

(e-4) HF (e-5) MEF (e-6) GDGF

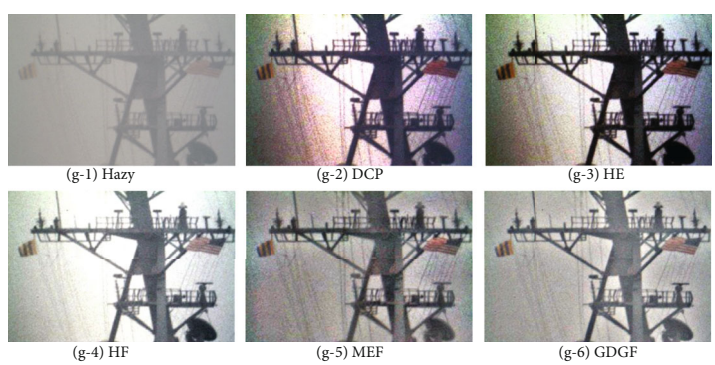
(e)



(f-1) Hazy (f-2) DCP (f-3) HE

(f-4) HF (f-5) MEF (f-6) GDGF

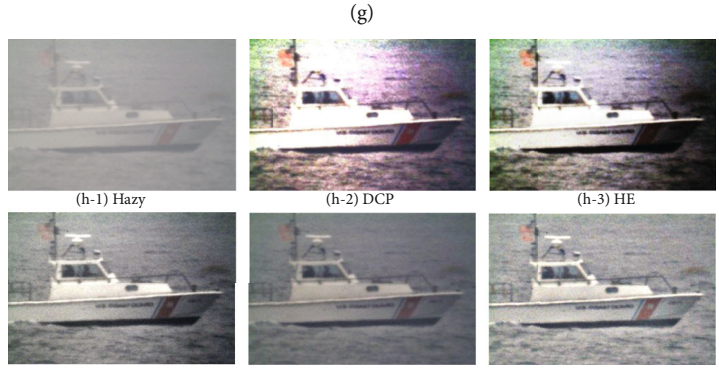
(f)



(g-1) Hazy (g-2) DCP (g-3) HE

(g-4) HF (g-5) MEF (g-6) GDGF

(g)

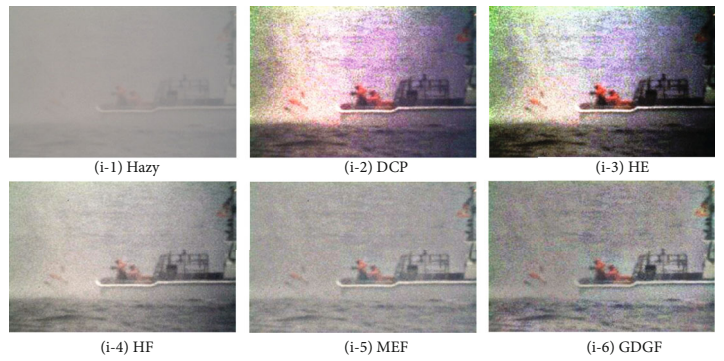


(h-1) Hazy (h-2) DCP (h-3) HE

(h-4) HF (h-5) MEF (h-6) GDGF

(h)

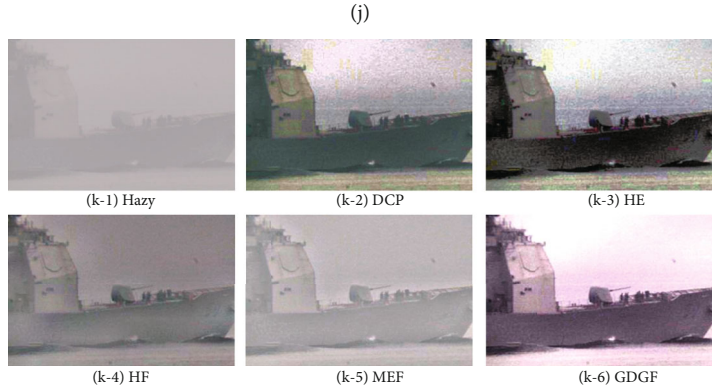
Figure 1: Continued.


(i-1) Hazy (i-2) DCP (i-3) HE
(i-4) HF (i-5) MEF (i-6) GDGF

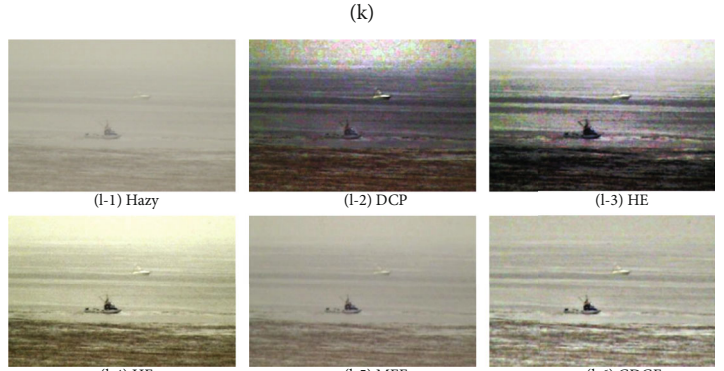
(i)


(j-1) Hazy (j-2) DCP (j-3) HE
(j-4) HF (j-5) MEF (j-6) GDGF

(j)


(k-1) Hazy (k-2) DCP (k-3) HE
(k-4) HF (k-5) MEF (k-6) GDGF

(k)


(l-1) Hazy (l-2) DCP (l-3) HE
(l-4) HF (l-5) MEF (l-6) GDGF

(l)

Figure 1: Continued.

Figure 1: Dehazing results of different algorithms.

Substituting (49) into (56) yields

$$J(w(n+1)) - J(w(n)) \le -(\eta-\gamma)(1-\alpha) \cdot \Gamma(1-\alpha)(\lambda_2 - c)^{\alpha-1} \left\| {}_cD^{\alpha}_{w(n)} J(n) \right\|^2 + NP\lambda_1^3(\gamma^2+\eta^2)] \left\| {}_cD^{\alpha}_{w(n)} J(n) \right\|^2 = -[(\eta-\gamma)(1-\alpha) \cdot \Gamma(1-\alpha)(\lambda_2 - c)^{\alpha-1} - NP\lambda_1^3(\gamma^2+\eta^2)] \left\| {}_cD^{\alpha}_{w(n)} J(n) \right\|^2. \quad (57)$$

When formula (35) holds, one can obtain

$$J(w(n+1)) - J(w(n)) \le 0. \quad (58)$$

This completes the proof of statement (36) of Theorem 5. From (38), the monotone decreasing sequence $J(w(n))$ is bound below $J(w(n)) \ge 0$. Since all bounded monotonic sequences of real numbers are convergent, $J(w(n))$ is convergent. Therefore, there exists $J* \ge 0$such that

$$\lim_{n \to +\infty} J(w(n)) = J*. \quad (59)$$

This completes the proof of statement (37) of Theorem 5.

Denote

$$\xi = (n-\gamma)(1-\alpha)\Gamma(1-\alpha)(\lambda_2 - c)^{\alpha-1} - NP\lambda_1^3(\gamma^2+\eta^2). \quad (60)$$

Substituting (60) into (57) yields

$$J(w(n+1)) - J(w(n)) \le -\xi \left\| {}_cD^{\alpha}_{w(n)} J(n) \right\|^2. \quad (61)$$

Thus, we have

$$J(w(n+1)) \le J(w(1)) - \xi \sum_{i=1}^{n} \left\| {}_cD^{\alpha}_{w(i)} J(i) \right\|^2. \quad (62)$$

Since $J(w(n+1)) \ge 0$, we can get

$$\xi \sum_{i=1}^{n} \left\| {}_cD^{\alpha}_{w(i)} J(i) \right\|^2 \le J(w(1)) - J(w(n+1)) \le J(w(1)). \quad (63)$$

When $n \to +8$, it holds that

$$\sum_{i=1}^{+\infty} \left\| {}_cD^{\alpha}_{w(i)} J(i) \right\|^2 \le \frac{1}{\xi} J(w(1)) < +8. \quad (64)$$

Thus, we have

$$\lim_{n \to +\infty} \left\| {}_cD^{\alpha}_{w(n)} J(w(n)) \right\| = 0. \quad (65)$$

This completes the proof of Theorem 5.

Table 1: PSNR results of different algorithms.

Image	DCP	HE	GIF	GDGIF	NGDGIF
a20100628	12.0979	14.3046	17.4799	17.4808	17.4807
b20100628	17.6293	14.4323	17.1904	17.1530	17.1538
c20100630	11.6236	13.5189	16.3300	16.2748	16.2752
d20100708	15.5520	17.9859	22.5850	22.6246	22.6303
e20110815	11.4669	10.6328	12.4806	12.4940	12.4912
f20110815	13.0932	16.8343	21.9115	21.9457	21.9507
g20110815	16.5273	12.1737	18.0335	17.1542	18.0716

Table 2: SSIM results of different algorithms.

Image	DCP	HE	GIF	GDGIF	NGDGIF
a20100628	0.6509	0.6993	0.7823	0.7876	0.7880
b20100628	0.7968	0.5510	0.7292	0.7279	0.7280
c20100630	0.5065	0.6152	0.7717	0.7693	0.7695
d20100708	0.6140	0.6619	0.4043	0.4429	0.4435
e20110815	0.6156	0.3674	0.7963	0.8118	0.8122
f20110815	0.6639	0.6359	0.7924	0.8032	0.8036
g20110815	0.5282	0.2992	0.4087	0.4193	0.4197

3.4. Gradient Domain-Guided Filtering Optimized by FOGDM-RBF. NGDGIF-FOGDMRBF is used to tune the regularization parameter of gradient domain-guided filtering to obtain better dehazing results with higher PSNR and SSIM. Its process is shown in Algorithm 1.

4. Experimental Demonstration and Discussion

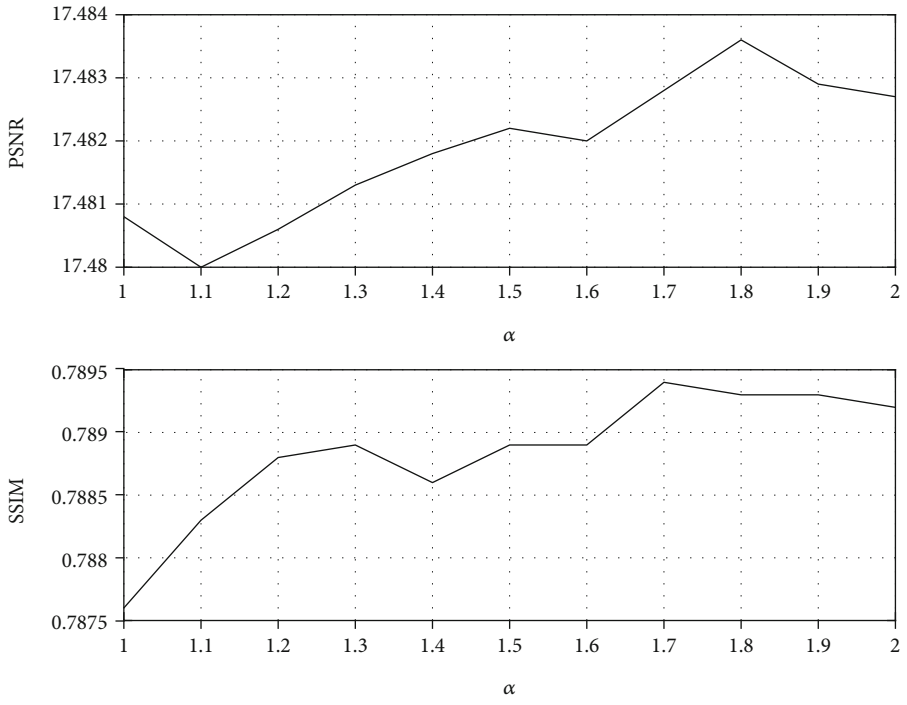
4.1. Dehazing Results. The ISCAS 2012 DATA are used in this experiment [20]. Dark channel prior (DCP) [21] and histogram equalization (HE) [22], homomorphic filtering (HF) [23] and multiple exposure fusion [24] are also used. Figure 1 shows 16 groups of images. Each group includes the observed foggy images, dehazing results by DCP, HE, HF, MEF, and NGDGIFFOGDMRBF. The local window radius is 16. The maximum iteration number is set as 200. The batch size is 256. The initial fractional order is 0.8. The learning rate is 0.9.

The algorithm is tested on computer with Intel (R) Core (TM) i3-4150T CPU @ 3.00 GHz 3.00 GHz, operating system of memory 4.00 GB 64 bit, and x64 based processor.

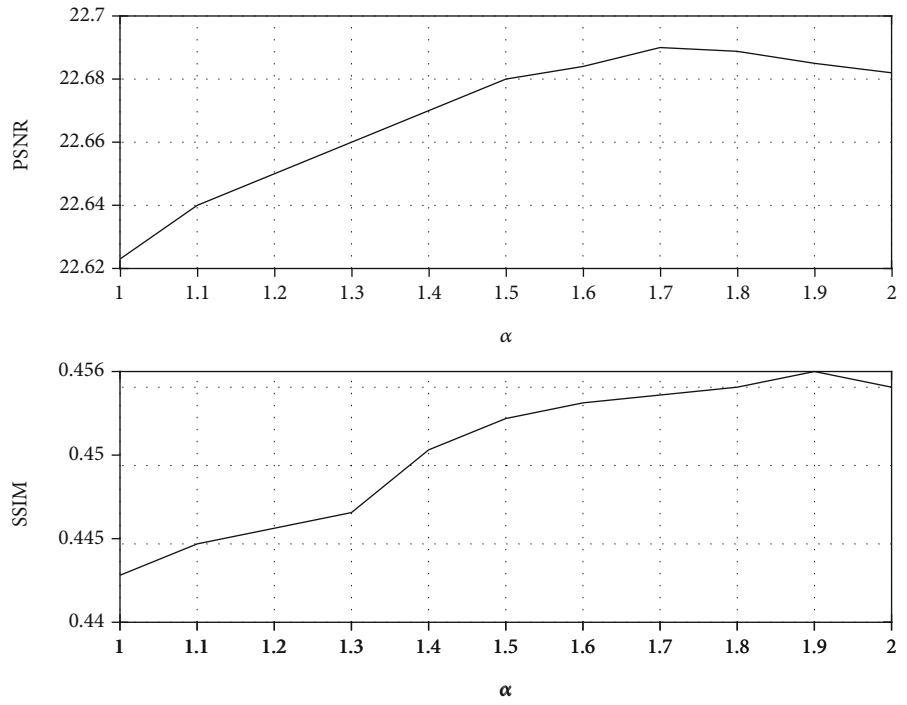
Figure 1 show that compared with other algorithms, NGDGIF-FOGDMRBF can significantly reduce halo and noise and obtain clear details.

4.2. Comparison of Different Guided Filter Algorithms. Table 1 shows the PSNR results of seven images dehazed by DCP, HE, guided image filtering algorithm, gradient domain-guided image filtering algorithm, and neural network using nonlinear gradient domain-guided image filtering algorithm, respectively.

In Table 1, we can see that NGDGIF-FOGDMRBF has the strongest ability to suppress noise among those different algorithms, which is consistent with the previous analysis.

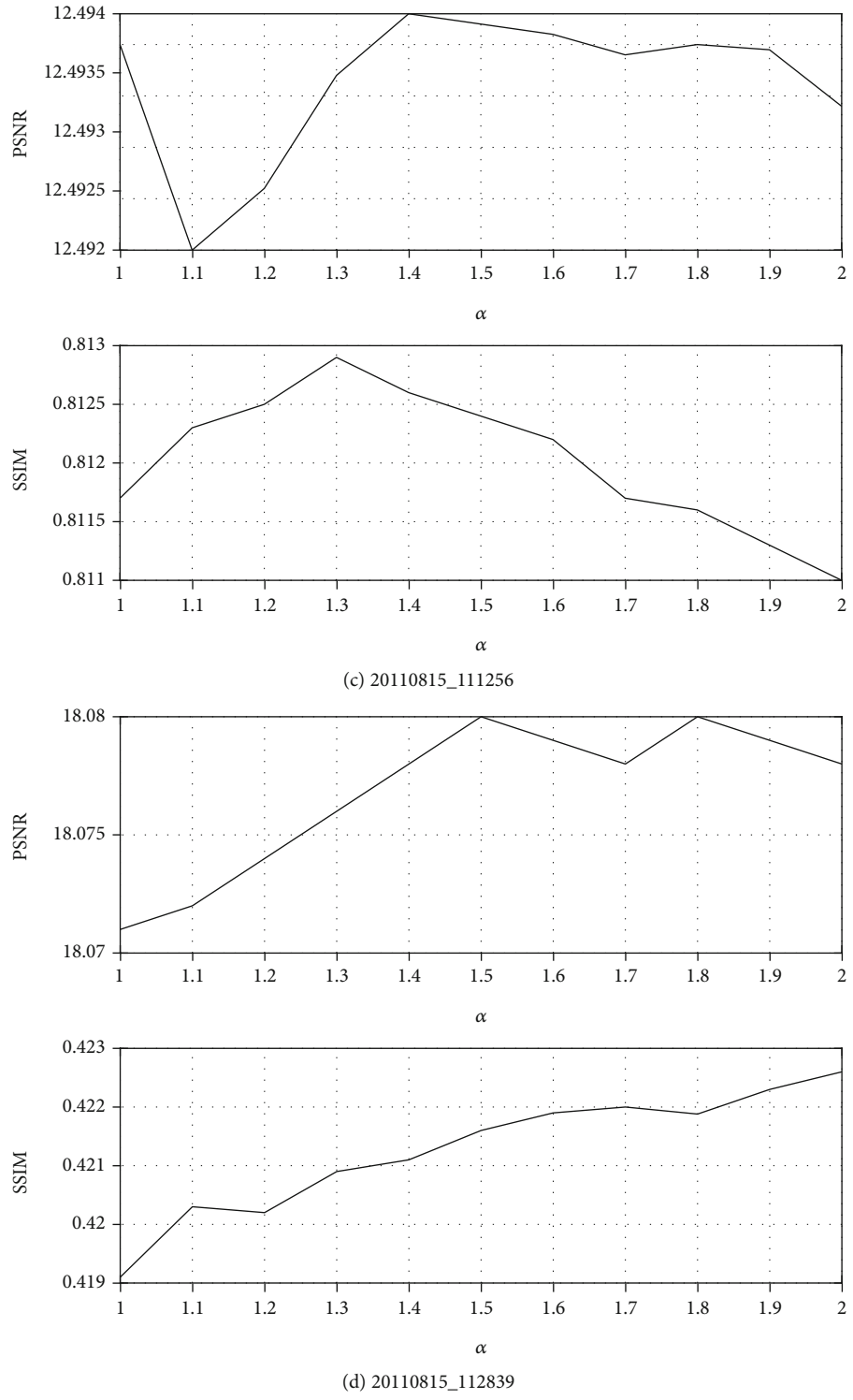


(a) 20100628_185221



(b) 20100708_100040

Figure 2: Continued.



(c) 20110815_111256

(d) 20110815_112839

Figure 2: PSNR and SSIM result comparison with different exponents.

Table 2 shows the SSIM results of seven images dehazed by DCP, HE, guided image filtering algorithm, gradient domain-guided image filtering algorithm, and neural network using nonlinear gradient domain-guided image filtering algorithm, respectively.

In Table 2, the PNSR and SSIM values of the method in this paper is the largest. Because other algorithms do not well suppress noise, and the edge part is also obscure, which cannot effectively maintain the actual structure of original images especially the details; their SSIM values are smaller. Table 2 can verify this analysis conclusion.

4.3. Impact Analysis of Exponent. Figure 2 shows the PSNR and SSIM results of four images dehazed by nonlinear

Table 3: PSNR results of different algorithms.

No.	DCP	HE	HF	MEF	GDGF
00	19.6114	18.4500	18.9750	18.8519	20.2280
01	10.3817	17.9635	9.1045	12.6416	21.5385
02	10.1464	15.5715	9.4777	14.5044	15.9377
03	12.0979	14.3046	14.4339	13.5019	15.1107
04	17.6293	14.4323	16.8970	12.7292	24.8062
05	11.6236	13.5189	16.3274	13.5376	16.3978
06	19.2815	18.3349	16.0087	20.1623	20.2834
07	15.5520	17.9859	19.0496	14.8773	23.3465
08	20.5020	16.6012	19.4656	18.7242	32.7885
09	11.4669	10.6328	10.6028	13.6934	24.5663
10	21.3094	17.5447	13.6847	11.8053	22.0709
11	13.0932	16.8343	15.7445	17.9465	25.0219
12	16.5273	12.1737	28.1119	17.9796	28.5447
13	18.9237	19.0109	17.7721	17.2217	19.0989
14	17.3505	17.1200	17.6917	15.3543	19.6345
15	17.6536	16.4935	17.2088	16.2894	19.4181

Table 4: SSIM results of different algorithms.

I	DCP	HE	HF	MEF	GDGF
00	0.8530	0.5726	0.6381	0.8678	0.8678
01	0.4960	0.5330	0.4680	0.6257	0.6583
02	0.4495	0.7480	0.5059	0.7628	0.7628
03	0.6509	0.6993	0.6972	0.6979	0.6979
04	0.7968	0.5510	0.8782	0.6236	0.7976
05	0.5065	0.6152	0.7761	0.7025	0.7859
06	0.7531	0.7794	0.8413	0.8624	0.9060
07	0.6140	0.6619	0.7448	0.7811	0.8243
08	0.7291	0.6327	0.9131	0.8404	0.9360
09	0.6156	0.3674	0.7623	0.8597	0.8741
10	0.6799	0.7317	0.6920	0.6817	0.7549
11	0.6639	0.6359	0.8101	0.6823	0.8537
12	0.5282	0.2992	0.8585	0.8024	0.8874
13	0.7375	0.3719	0.6980	0.7833	0.8961
14	0.5187	0.6394	0.6377	0.6944	0.7234
15	0.6848	0.4502	0.8125	0.8878	0.9058

gradient domain-guided image filtering algorithm with different exponent α.

Figure 2 shows that with the increase of exponent α, the PSNR and SSIM are increased. However, when exponent α exceeds a certain threshold, the PSNR and SSIM begin to decrease. Since linear model is a special case of nonlinear model, the nonlinear model has the advantages of wider parameter selection range and more flexible.

4.4. Comparison of Results of Different Algorithms. Tables 3 and 4 list the PSNR and SSIM results of 16 images dehazed by DCP, HE, HF, MEF, and FOGDMRBF-GDGF, respectively.

Tables 3 and 4 show that FOGDMRBF-GDGF has the strongest ability to suppress noise among those different algorithms consistent with the previous analysis. Other algorithms produce obscure edges different from the actual structure of original images especially the details. So their SSIM value is smaller.

4.5. Impact Analysis of Parameters. The PSNR with different regularization parameters for image No.05 is listed in Table 5. With the increase of regularization parameter, the PSNR is improved. When the regularization parameter exceeds a certain threshold, the PSNR begins to decrease.

The optimal regularization parameter calculated by FOGDMRBF is also 0.04. By the neural network, the regularization parameter can be dynamically adjusted with different images and adaptive to different detail layers. The neural network optimization has the advantages of wider parameter selection range and more flexible.

In formula (4), a determines the gradient preservation of the final image and represents the image edge preservation. When a is small, the gradient is small, the image edge is blurred, and the smoothing force is greater, and vice versa. The regularization parameter ε is used to prevent a from becoming too large; thus, it is less than 1. The smaller ε is, the smaller the superposition smoothing effect is, and vice versa. Therefore, the guiding filter uses a and ε to determine the edge preservation and smoothness of the output image.

Table 5: PSNR results of different regularization parameters.

ε	FSNR
0.01	16.3668
0.03	16.3912
0.04	16.3978
0.05	16.3963
0.06	16.3934
0.1	16.3830
0.5	16.3724
1	16.3709

5. Conclusion

The research work of this paper is summarized as follows. A nonlinear gradient domain-guided image filtering algorithm was proposed. The optimal value of model parameters is proved. Fractional-order calculus is applied to gradient descent with momentum algorithm for training neural network. The new algorithm is used to adjust the weights of the neural network to improve its learning speed and performance for optimizing gradient domain-guided filtering. The convergence of the FOGDM-RBF is proved.

Future research will continue to improve the neural network for better accuracy and convergence speed for better dehazing more complex images.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

The work is supported by the National Natural Science Foundation of China (51879119), Natural Science Foundation of Fujian Province (2018J05085), the high level research and cultivation fund of transportation engineering discipline in Jimei University (HHXY2020003), National Natural Science Foundation Cultivation Project of Jimei University (ZP2020005).

References

[1] Q. Lei, C. J. Shi, and T. T. Chen, "Single sea image haze removal method based on segmentation of sky area," *Computer Engineering*, vol. 41, no. 5, pp. 237–242, 2015.

[2] F. C. Dong, *Video image defogging method research outside the ship cabin [M.S. thesis]*, Control Science and Engineering, Harbin Engineering University, Harbin, China, 2018.

[3] K. L. Wu, Z. H. Tong, P. S. Zhang, Y. N. Liu, and Z. Liu, "Recognition of marine vessels under complex weather conditions based on deep learning," *Science Technology and Engineering*, vol. 19, no. 3, pp. 130–135, 2019.

[4] Z. B. Li, G. Y. Li, B. S. Niu, and F. Peng, "Sea cucumber image dehazing method by fusion of retinex and dark channel," *IFAC Papers OnLine*, vol. 51-17, pp. 796–801, 2018.

[5] F. Ding, *System Identification – Iterative Search Principle and Identification Methods*, Science Press, Beijing, China, 2018.

[6] L. J. Liu, F. Ding, L. Xu, J. Pan, A. Alsaedi, and T. Hayat, "Maximum likelihood recursive identification for the multivariate equation-error autoregressive moving average systems using the data filtering," *IEEE Access*, vol. 7, pp. 41154–41163, 2019.

[7] F. Kou, W. H. Chen, C. Y. Wen, and Z. G. Li, "Gradient domain guided image filtering," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 4528–4539, 2015.

[8] P. X. Zhuang, Q. S. Liu, and X. H. Ding, "Pan-GGF: a probabilistic method for pan-sharpening with gradient domain guided image filtering," *Signal Processing*, vol. 156, pp. 177–190, 2019.

[9] P. X. Zhuang, X. W. Zhu, and X. H. Ding, "MRI reconstruction with an edge-preserving filtering prior," *Signal Processing*, vol. 155, pp. 346–357, 2019.

[10] H. Yin, Y. H. Gong, and G. P. Qiu, "Side window guided filtering," *Signal Processing*, vol. 165, pp. 315–330, 2019.

[11] F. Kou, Z. G. Li, C. Y. Wen, and W. H. Chen, "Edge-preserving smoothing pyramid based multi-scale exposure fusion," *Journal of Visual Communication and Image Representation*, vol. 53, pp. 235–244, 2018.

[12] P. Yin, S. Zhang, J. Lyu, S. Osher, Y. Qi, and J. Xin, "Blended coarse gradient descent for full quantization of deep neural networks," *Research in the Mathematical Sciences*, vol. 6, no. 1, 2019.

[13] M. Kobayashi, "Gradient descent learning for quaternionic Hopfield neural networks," *Neurocomputing*, vol. 260, pp. 174–179, 2017.

[14] L. Wang, Y. Yang, R. Min, and S. Chakradhar, "Accelerating deep neural network training with inconsistent stochastic gradient descent," *Neural Networks*, vol. 93, pp. 219–229, 2017.

[15] W. Wu, N. M. Zhang, Z. X. Li, L. Li, and Y. Liu, "Convergence of gradient method with momentum for back-propagation neural networks," *Journal of computational mathematics*, vol. 26, no. 4, pp. 613–623, 2008.

[16] D. UTOMO, "Stock price prediction using back propagation neural network based on gradient descent with momentum and adaptive learning rate," *Journal of Internet Banking and Commerce*, vol. 22, no. 3, pp. 1–16, 2017.

[17] C. T. Kinh, L. V. Hien, and T. D. Ke, "Power-rate synchronization of fractional-order nonautonomous neural networks with heterogeneous proportional delays," *Neural Processing Letters*, vol. 47, no. 1, pp. 139–151, 2018.

[18] J. Wang, Y. Q. Wen, Y. D. Gou, Z. Ye, and H. Chen, "Fractional-order gradient descent learning of BP neural networks with Caputo derivative," *Neural Networks*, vol. 89, pp. 19–30, 2017.

[19] S. Khan, I. Naseem, M. A. Malik, R. Togneri, and M. Bennamoun, "A fractional gradient descent-based RBF neural network," *Circuits, Systems, and Signal Processing*, vol. 37, no. 12, pp. 5311–5332, 2018.

[20] G. L. Yang, B. J. Zhang, and Z. Y. Sang, "A Caputo-type fractional order gradient descent learning of BP neural networks," in *14th International Symposium Sapporo, Hakodate, and Muroran*, vol. 2017, pp. 547–554, Springer, Hokkaido, Japan, 2017.

[21] ISCAS, "DATA," 2012, http://videoprocessing.ucsd.edu/~kgibson/cem.htm.

[22] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2341–2353, 2011.

[23] M. Abdullah-al-Wadud, M. H. Kabir, M. Akber Dewan, and O. Chae, "A dynamic histogram equalization for image contrast enhancement," *IEEE Transactions on Consumer Electronics*, vol. 53, no. 2, pp. 593–600, 2007.

[24] L. Yu, X. B. Liu, and G. Z. Liu, "A new dehazing algorithm based on overlapped sub-block homomorphic filtering," in *Eighth International Conference on Machine Vision (ICMV 2015)*, vol. 9875, article 987502, International Society for Optics and Photonics, Barcelona Spain, 2015.