Weighted Guided Image Filtering With Steering Kernel

Zhonggui Sun, Bo Han, Jie Li, Jin Zhang, and Xinbo Gao, *Senior Member, IEEE*

***Abstract*— Due to its local property, guided image filter (GIF) generally suffers from halo artifacts near edges. To make up for the deficiency, a weighted guided image filter (WGIF) was proposed recently by incorporating an edge-aware weighting into the filtering process. It takes the advantages of local and global operations, and achieves better performance in edge-preserving. However, edge direction, a vital property of the guidance image, is not considered fully in these guided filters. In order to overcome the drawback, we propose a novel version of GIF, which can leverage the edge direction more sufficiently. In particular, we utilize the steering kernel to adaptively learn the direction and incorporate the learning results into the filtering process to improve the filter's behavior. Theoretical analysis shows that the proposed method can get more powerful performance with preserving edges and reducing halo artifacts effectively. Similar conclusions are also reached through the thorough experiments including edge-aware smoothing, detail enhancement, denoising, and dehazing.**

***Index Terms*— Guided image filter, edge-preserving smoothing, steering kernel, halo artifact, denoising, dehazing.**

I. Introduction

EDGE preserving smoothing technique plays an important role in computer vision and image processing applications, such as image denoising [1]–[3], detail enhancement [4], [5], structure transferring [6], dehazing [7], image fusion [8], etc. Up to now, a large number of filters with edge-preserving smoothing property have been proposed and developed. Representative methods are anisotropic diffusion [9], bilateral filter [10] and its improved version [11], [12], weighted least square (WLS) [5], domain transform filter [13], guided image filter (GIF) [14], etc. Among them, bilateral filter and GIF have been used broadly for their simplicity and better performance.

Bilateral filter takes the weighted average of each pixel's neighbors as its filtering output, where the weights rely on spatial similarity and range similarity. For its effectiveness, bilateral filter has been widely used in image processing [2]–[4]. However, as pointed out in [14], this filter cannot keep the consistency of gradient direction at edge pixels. Consequently, it may suffer from gradient reversal artifacts near edges, especially when being utilized in detail enhancement.

To overcome the drawback of bilateral filter, GIF was proposed by He *et al.* [14]. Derived from a local linear model, it computes the filtering output by taking the content of a guidance image into consideration. And the guidance image can be the input image itself or another different one. This filter not only has edge-preserving property like bilateral filter, but also avoids the gradient reversal artifacts successfully. GIF has been applied in a wide variety of applications, including edge-aware smoothing, detail enhancement, image matting/feathering, high dynamic range compression, image fusion, denoising, dehazing, joint up sampling, etc. Meanwhile, inspired by its guided strategy, several edge-preserving methods have emerged [15]–[17]. And GIF has attracted much more attentions in image processing fields.

Note that, as pointed out in [14], GIF may suffer from halo artifacts near edges, especially when being applied in edge-aware smoothing. One major reason leading to this defect is its local property. In addition, the regularization parameter is invariable in GIF, which is another major reason inducing halo artifacts [18]. Based on the above considerations, an improved version named weighted guided image filter (WGIF) has been developed [18] by introducing an edge-aware weighting to adjust the regularization parameter adaptively. Wherein, the local and global information of the guidance image are both integrated to compute the edge-aware weighting. Compared with the traditional GIF, WGIF can preserve sharp edges and reduce halo artifacts more effectively.

As a key step in GIF and WGIF, averaging strategy (mean filter) is adopted to deal with the fact that each pixel is contained in several overlapping windows [14], [18]. This is a popular approach for overlapping windows and its effectiveness had been demonstrated widely [19]. However, due to its isotropy, mean filter assigns the same weights for all pixels in the local window, which neglects the directions of edges in the image under studied. As a result, the edges are prone to be blurred in the averaging process. Considering the edge structure plays an important role in the guided filters, it is significant to improve them by making full use of the edge direction.

Over the last few decades, with the rapid development of machine learning, numerous local kernels have been developed to describe edge structure more effectively. Representatives

Manuscript received October 29, 2018; revised April 1, 2019, May 10, 2019, and June 13, 2019; accepted July 4, 2019. Date of publication July 19, 2019; date of current version September 23, 2019. This work was supported in part by the National Key Research and Development Program of China under Grant 2016QY01W0200, and the National Natural Science Foundation of China under Grant 61432014, Grant U1605252, Grant 61772402, and Grant 11801249, and in part by the Nature Science Foundation of Shandong Province China under Grant ZR2014FM032 and Grant K17LB2501. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Khan M. Iftekharuddin. *(Corresponding author: Xinbo Gao.)*

Z. Sun, B. Han, and J. Zhang are with the School of Mathematical Sciences, Liaocheng University, Liaocheng 252000, China (e-mail: altlp@hotmail.com; lcuhanbo@gmail.com; zhangjin@lcu.edu.cn).

J. Li and X. Gao are with the School of Electronic Engineering, Xidian University, Xi'an 710071, China (e-mail: leejie@mail.xidian.edu.cn; xbgao@mail.xidian.edu.cn).

Digital Object Identifier 10.1109/TIP.2019.2928631

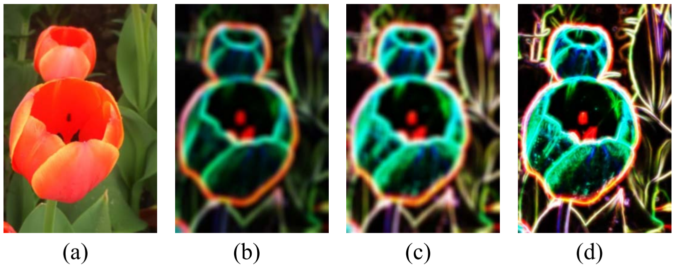


Fig. 1. Comparison of the averaged edge weight of a color image (Tulip) obtained by different strategies. (a) Input image. (b, c, d) The averaged edge weight by GIF, WGIF and our SKWGIF, respectively.

include bilateral kernel, joint bilateral kernel and steering kernel [20]. Among them, the steering kernel shows more powerful performance due to its adaptation to the local information of the image under studied. It is the adaptive property that makes it achieve better results in image denoising [20], deblurring [21] and image super-resolution [22], etc.

In this paper, to make full use of the edge direction, we utilize the steering kernel to learn local structure prior of the guidance image, and then assemble it with WGIF to well pose the deficiencies of the guided filters. For the sake of convenience, this novel method is called weighted guided image filtering with steering kernel (SKWGIF). Particularly, we replace the mean filtering strategy in GIF and WGIF with a weighted average, where the weights are described by the local steering kernel depended on the guidance image. This novel strategy enables SKWGIF to be more reliable, which can suppress the halo artifacts and preserve edges better simultaneously. SKWGIF shows distinguished behavior in many applications, such as edge-preserving smoothing, detail enhancement, denoising and dehazing. Fig. 1 provides the comparison about an edge indicator, i.e., averaged edge weight (introduced in section III). Here, the guidance image is the input image (Tulip) itself.

The main contributions of this paper are twofold:

1) We propose a weighted averaging strategy which better respects the edge direction of guidance image to address the problem of overlapping windows.
2) We introduce this strategy into guided filters to enhance their performance near edges. Meanwhile, the outstanding behaviors of the proposed filter have been supported in both rigorous theoretical analysis and thorough experimental results.

The remainder of this paper is organized as follows. The edge-aware weighting and steering kernel are briefly reviewed in section II. Section III presents the proposed method and analyzes the edge-preserving property of it. The applications and experimental results are given in Section IV. Finally, Section V concludes this paper.

II. Related Work

A. Edge-Aware Weighting in Weighted Guided Image Filtering

Compared with the original GIF, WGIF [18] can adjust the regularization parameter adaptively by introducing the edge-aware weighting, which effectively utilizes both local and global information of the guidance image. Specifically, WGIF employs the local variance of each pixel in a guidance image to compute the edge-aware weighting $\psi_I(k)$, expressed as

$$\psi_I(k) = \frac{1}{N}\sum_{i=1}^{N}\frac{\sigma_{I,1}^2(k)+\lambda}{\sigma_{I,1}^2(i)+\lambda}. \tag{1}$$

where λ is a small constant whose value is $(0.001 \times L)^2$, while L is the dynamic range of an input image. I is a guidance image with N pixels, $\sigma_{I,1}^2(k)$ and $\sigma_{I,1}^2(i)$ are the variances of I in the 3×3 windows centered at pixels k and i respectively. The weighting $\psi_I(k)$ measures the edge strength of pixel k with respect to the whole guidance image. This value is usually larger than 1.0 when pixel k is at an edge. And $\psi_I(k)$ will be smaller than 1.0 when the pixel locates in flat area [18]. This property of $\psi_I(k)$ can well match one feature of human visual system [23]. Due to the edge-aware ability, WGIF outperforms GIF in many applications.

B. Steering Kernel

The steering kernel is one of the most attractive tools in image processing fields [20]. It considers the gradients and analyzes the radiometric similarities of pixels in a local window and usually takes the form as follows.

$$w_{ik} = \frac{\sqrt{\det(C_i)}}{2\pi h^2}\exp\left\{-\frac{(x_i-x_k)^T C_i (x_i-x_k)}{2h^2}\right\}. \tag{2}$$

where h is the smoothing parameter to control the supporting range of the kernel, x_i and x_k are pixel coordinates. C_i is a symmetric gradient covariance matrix computed from the local square window ω_i centered at pixel i, which can be estimated as

$$\hat{C}_i \approx G_i^T G_i, G_i = \begin{bmatrix} \vdots & \vdots \\ I_{k,x_1} & I_{k,x_2} \\ \vdots & \vdots \end{bmatrix}, \quad \forall k \in \omega_i. \tag{3}$$

Here, G_i is the local gradient matrix, I_{k,x_1} and I_{k,x_2} are the first derivatives along x_1 and x_2 directions at pixel k, which depend on the pixel intensity differences of the guidance image I. While this approach is simple and has nice tolerance to noise, the resulting estimate of the covariance may in general be rank deficient or unstable [20]. To obtain a stable estimate of covariance matrix, a parametric approach is adopted in [20]. Specifically, the matrix is assumed as a composition of three components as follows

$$C_i = \gamma_i U_{\theta_i} \Lambda_i U_{\theta_i}^T, \tag{4}$$

$$U_{\theta_i} = \begin{bmatrix} \cos\theta_i & \sin\theta_i \\ -\sin\theta_i & \cos\theta_i \end{bmatrix}, \tag{5}$$

$$\Lambda_i = \begin{bmatrix} \sigma_i & 0 \\ 0 & \sigma_i^{-1} \end{bmatrix}. \tag{6}$$

where γ_i is the scaling parameter, U_{θ_i} and Λ_i are the rotation and elongation matrix controlled by θ_i and σ_i, respectively. And the three parameters γ_i, θ_i and σ_i are all determined by the singular value decomposition (SVD) of the local gradient

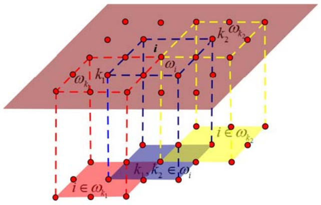


Fig. 2. Schematic illustration of the overlapping windows that cover pixel i.

matrix G_i. We refer the reader to [20] for more details. It is the adaptive property of the covariance matrix that reflects the local structure of image and leads to the kernel function spreading along the local edges. That is why the steering kernel can effectively learn edge direction from the guidance image.

III. The Proposed Method and Method Analysis

A. The Proposed Method

The GIF, WGIF and our proposed method are all based on a common assumption: the filtering output q is a linear transform of the guidance image I in a local square window. And the local linear model is formulated as

$$q_i = a_k I_i + b_k, \quad \forall i \in \omega_k, \tag{7}$$

where ω_k is still a square window centered at pixel k with a radius r. a_k and b_k are the linear coefficients assumed to be constant in ω_k, whose values are determined by minimizing a cost function. And the cost function describes the difference between the input p and the output q. Followed with the linear model (7), the cost function in GIF is defined as follows.

$$E\,(a_k, b_k) = \sum\nolimits_{i\in\omega_k} ((a_k I_i + b_k - p_i)^2 + \varepsilon a_k^2). \tag{8}$$

Here, ε is a regularization parameter penalizing large a_k and its value is fixed for all local windows.

As opposed to GIF, WGIF can adjust the regularization parameter adaptively by employing the edge-aware weighting $\psi_I(k)$ defined in equation (1). To inherit the advantages of local and global smoothing filters like WGIF, we also minimize the cost function in the form

$$E(a_k, b_k) = \sum\nolimits_{i\in\omega_k} ((a_k I_i + b_k - p_i)^2 + \frac{\varepsilon}{\psi_I(k)} a_k^2). \tag{9}$$

Accordingly, the solution is given by:

$$a_k = \frac{\frac{1}{|\omega|}\sum_{i\in\omega_k} I_i p_i - \mu_k \bar{p}_k}{\sigma_k^2 + \frac{\varepsilon}{\psi_I(k)}}, \tag{10}$$

$$b_k = \bar{p}_k - a_k \mu_k. \tag{11}$$

where μ_k and σ_k^2 are the mean and variance of the guidance image I in ω_k, $|\omega|$ and $\bar{p}_k = \frac{1}{|\omega|}\sum_{i\in\omega_k} p_i$ are the number of pixels and the mean of the input image p in ω_k, respectively.

As shown in Fig. 2, a pixel i is covered by $|\omega|$ overlapping windows that contain i. Hence the filtering output q_i in equation (7) will get different values when it is computed in different windows. To address this problem, both GIF and WGIF adopt a naive strategy, i.e., averaging all the possible values of q_i. This averaging strategy can be formulated as

$$q_i = \frac{1}{|\omega|}\sum\nolimits_{k|i\in\omega_k} (a_k I_i + b_k). \tag{12}$$

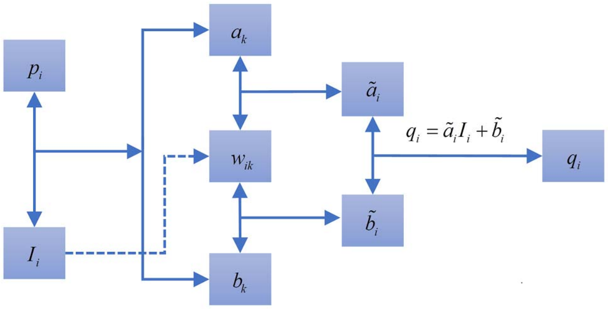


Fig. 3. The flow chart of the proposed method.

Noticing that $\sum_{k|i\in\omega_k} a_k = \sum_{k\in\omega_i} a_k$ due to the symmetry of the box window, in [14] and [18], equation (12) is reformulated by:

$$q_i = \bar{a}_i I_i + \bar{b}_i. \tag{13}$$

where $\bar{a}_i = \frac{1}{|\omega|}\sum_{k\in\omega_i} a_k$ and $\bar{b}_i = \frac{1}{|\omega|}\sum_{k\in\omega_i} b_k$ are the mean values of a_k and b_k, respectively.

Different from GIF and WGIF, they both use the averaging strategy which neglects the edge direction, our SKWGIF employ the weighted averaging method to capture the information. In particular, we select the normalized local steering kernel w_{ik} of guidance I as the weight function. Then equation (12) is reformulated as

$$q_i = \sum\nolimits_{k\in\omega_i} w_{ik}(a_k I_i + b_k). \tag{14}$$

Go a step further, equation (14) can be rewritten by

$$q_i = \tilde{a}_i I_i + \tilde{b}_i. \tag{15}$$

where $\tilde{a}_i = \sum_{k\in\omega_i} w_{ik} a_k$ and $\tilde{b}_i = \sum_{k\in\omega_i} w_{ik} b_k$ are the weighted averages of a_k and b_k, respectively. The motivation and rationality for this weighted strategy will be explicated in subsection III-B. The flow chart of the proposed method is presented in Fig. 3.

As shown in the flow chart, given a pair of images, input p and guidance I, the proposed method first computes the edge-aware weighting $\psi_I(k)$ from equation (1). And then, it estimates the linear coefficients a_k and b_k according to equations (10) and (11), respectively. Forward, after obtaining the local steering kernel w_{ik} from equation (2), the method computes the weighted average of a_k and b_k, i.e., $\tilde{a}_i$ and $\tilde{b}_i$, respectively. Finally, it outputs q_i by equation (15).

Now we summarize the proposed method in Algorithm 1. Where f_{mean} is a fast mean filter with a wide variety of $O(N)$ time method [14], and N is the number of pixels of the guidance image.

Algorithm 1 The Weighted Guided Image Filtering With Steering Kernel

```
1: Input: $p$ (filtering input image); $I$ (guidance image);
2: Parameters:
    $r$ , $r_2$ (radius of local square window $\omega_k$ );
    $\varepsilon$ (regularization parameter);
    $\lambda$ (small constant of edge-aware weighting $\psi_I(k)$ );
    $h$ (smoothing parameter that controls the supporting range of steering kernel).
3: Compute the mean matrices and correlation matrices:
    $mean_I = f_{mean}(I,r), mean_{I2} = f_{mean}(I,r_2)$
    $mean_p = f_{mean}(p,r), mean_{p2} = f_{mean}(p,r_2)$
    $corr_I = f_{mean}(I.*I,r_2), corr_{Ip} = f_{mean}(I.*p,r)$
4: Compute the variance matrices, edge-aware weighting matrices, covariance matrices and linear coefficient matrices:
    $\mathrm{var}_I = corr_I - mean_I.*mean_I, \mathrm{var}_{I2} = corr_{I2} - mean_{I2}.*mean_{I2}$
    $\psi_I = \frac{1}{N}\left(\mathrm{var}_{I_2} + \lambda\right) * sum(sum(1./(\mathrm{var}_{I_2} + \lambda)))$
    $\mathrm{cov}_{Ip} = corr_{Ip} - mean_I.*mean_p$
    $a = \mathrm{cov}_{Ip} / (\mathrm{var}_I + \varepsilon / \psi_I), b = mean_p - a.*mean_I$
5: Compute the steering kernel matrices $w_{ik}$ according to Eq. (2) and the weighted average linear coefficient matrices:
    $mean_a = \sum_{k\in\omega_i} w_{ik} a_k, mean_b = \sum_{k\in\omega_i} w_{ik} b_k$
6: Filtering output: $q = mean_a.*I + mean_b$
```

B. Method Analysis

From equation (7), the following result is supported, i.e.,

$$\nabla q_i = a_k \nabla I_i. \tag{16}$$

where ∇q_i and ∇I_i are the first derivative of the filtering output q and guidance image I at pixel i, respectively. It ensures that q has an edge only if I has an edge [14]. In this paper, we call a_k the edge weight.

For convenience of analysis, we temporarily assume that the guidance I and input p are the same image. In this case, the linear coefficient a_k can be reformulated as $\sigma_k^2/(\sigma_k^2+\varepsilon)$ and $\sigma_k^2/(\sigma_k^2+\varepsilon/\psi_I(k))$ in GIF and WGIF, respectively. As pointed out in [18], the value of $\psi_I(k)$ is usually larger than 1.0 when the pixel k is at an edge. So, a_k in WGIF is larger (closer to 1.0) than that in GIF. This enables the sharp edges can be preserved better by WGIF. As aforementioned in subsection III-A, a pixel i generally corresponds to multiple filtering outputs due to the windows overlapping. To address this problem, both GIF and WGIF adopt an averaging strategy. Therein equation (16) degenerates into

$$\nabla q_i \approx \bar{a}_i \nabla I_i. \tag{17}$$

It ensures these two guided filters perform better than the conventional edge-preserving ones [18]. However, for its isotropy, mean filter ignores the edge direction information of image inevitably. As a consequence, edges blur and halo artifacts arise in these filtering results. Hence, there is still room to improve the edge-preserving effectiveness in GIF and WGIF. Based on these considerations, our SKWGIF advocates a weighted averaging strategy that can respect the edge direction of guidance image effectively. Accordingly, equation (17) is changed into

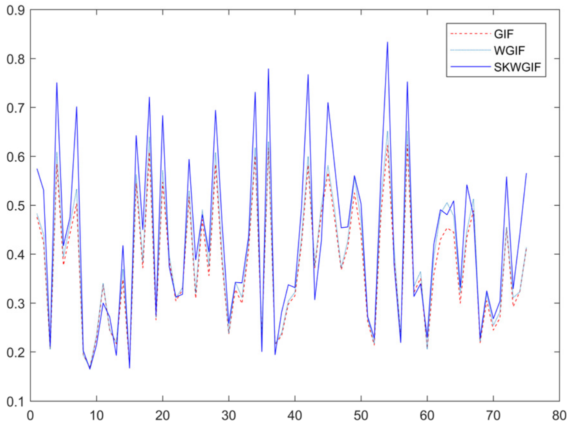


Fig. 4. Comparison of different averaging strategies.

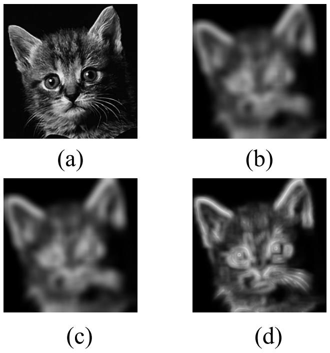


Fig. 5. Comparison of the averaged edge weights of a gray image (Cat) obtained by different strategies. (a) Input image. (b), (c), (d) The averaged edge weights by GIF, WGIF and SKWGIF, respectively.

$$\nabla q_i \approx \tilde{a}_i \nabla I_i. \tag{18}$$

where $\tilde{a}_i$ is the weighted average of a_k. For convenience, we call $\bar{a}_i$ and $\tilde{a}_i$ the averaged edge weight. Due to inheriting the edge direction, $\tilde{a}_i$ is expected larger than $\bar{a}_i$ in the sharp edges. Thus the edges can be preserved better in SKWGIF.

In order to further verify the above conclusions, we conduct the experiment on a real gray image (Cat) and the guidance is still the input image itself. Here, we divide the pixels of guidance I into two groups: one group satisfies the condition of $\psi_I(k) \geq 1$ (belong to sharp edge) and the other satisfies $\psi_I(k) < 1$ (not belong to sharp edge). Then, we sample the pixels at sharp edges randomly and compare the corresponding values of $\bar{a}_i$ and $\tilde{a}_i$. Fig. 4 illustrates that the values of $\tilde{a}_i$ in SKWGIF are generally larger than those in GIF and WGIF. Moreover, the statistical u test also shows that there is a significant difference between $\tilde{a}_i$ and $\bar{a}_i$ with a significance level of 0.05.

In addition, Fig. 5 presents the $\bar{a}_i$ and $\tilde{a}_i$, where the brighter pixels represent the larger values. Clearly, our method has a

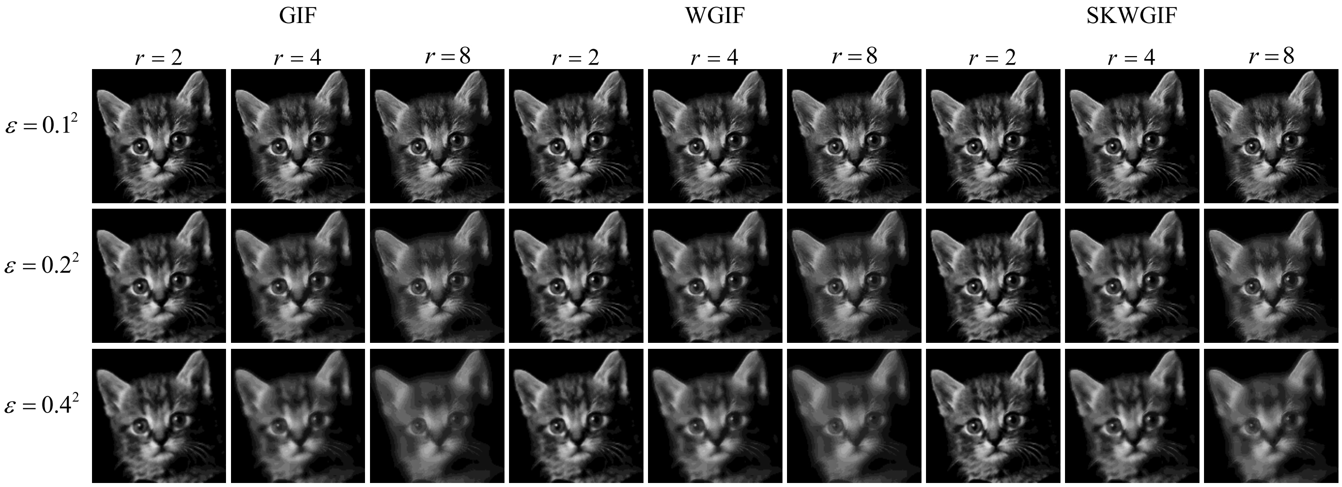


Fig. 6. Edge-preserving filtering results of GIF, WGIF and SKWGIF.

TABLE I

QUANTITATIVE ASSESSMENT OF DIFFERENT IMAGE EDGE-PRESERVING SMOOTHING METHODS

	PSNR								
	GIF			WGIF			SKWGIF		
	$r = 2$	$r = 4$	$r = 8$	$r = 2$	$r = 4$	$r = 8$	$r = 2$	$r = 4$	$r = 8$
$\varepsilon = 0.1^2$	30.98	30.17	29.71	31.29	30.42	29.92	32.24	31.22	30.79
$\varepsilon = 0.2^2$	26.85	25.26	23.90	27.26	25.59	24.14	28.46	26.55	25.11
$\varepsilon = 0.4^2$	24.53	22.53	20.67	24.91	22.81	20.84	26.28	23.83	21.78

	SSIM								
	GIF			WGIF			SKWGIF		
	$r = 2$	$r = 4$	$r = 8$	$r = 2$	$r = 4$	$r = 8$	$r = 2$	$r = 4$	$r = 8$
$\varepsilon = 0.1^2$	0.8983	0.8924	0.9043	0.9006	0.8943	0.9059	0.9150	0.8994	0.9099
$\varepsilon = 0.2^2$	0.8000	0.7531	0.7407	0.8072	0.7605	0.7474	0.8435	0.7799	0.7618
$\varepsilon = 0.4^2$	0.7228	0.6303	0.5683	0.7315	0.6393	0.5760	0.7891	0.6757	0.6039

better behavior (larger value) than GIF and WGIF in the sharp edges. And the similar conclusions are also enhanced on a color image (Tulip), which are presented in Fig. 1.

Obviously, similarly as the theoretical analysis, the experiments on the real images also indicate that SKWGIF can achieve a better edge-preserving smoothing result than GIF and WGIF.

IV. APPLICATIONS AND EXPERIMENTAL RESULTS

To further verify the effectiveness of the proposed SKWGIF, we conduct a series of experiments for thorough applications include image edge-aware smoothing, detail enhancement, denoising and dehazing. In each application, we compare SKWGIF with GIF and WGIF, and employ PSNR and SSIM [24] indices to evaluate their objective performance.

Followed with [14], the pixel intensities are all scaled in [0, 1] in these experiments. As similarly in [20], the size and global smooth parameter h of the steering kernel w_{ik} are set to $(2r+1) \times (2r+1)$ and 2.4, respectively. The only difference is that, w_{ik} is obtained by an iteration process from the noisy image in [20], while the process is manipulated only once in our SKWGIF because the kernel is derived from the guidance image which is trustworthy relatively. Besides h and w_{ik}, there are still two parameters, i.e., the radius r and smoothing parameter ε, needing to be determined in SKWGIF. Note that, as a kind of fundamental tool, the guided filters can be used in many applications, and the two parameters various widely according to different applications, which generally depend on the experimental experience [14], [18]. As indicated in the references and our experiments, a larger r (or similarly ε) will lead to a smoother result than a smaller one. For the sake of fairness, we respect the parameter settings of [14] and [18] in our experiments.

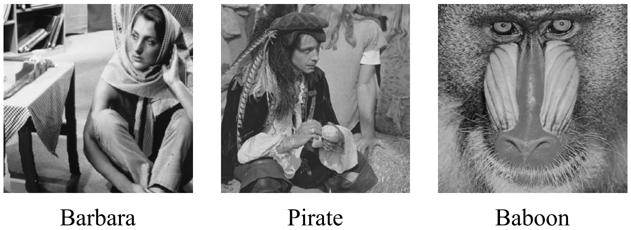


Fig. 7. Test images of edge-aware smoothing.

TABLE II

QUANTITATIVE ASSESSMENT OF DIFFERENT IMAGE EDGE-PRESERVING SMOOTHING METHODS.

		PSNR								
		GIF			WGIF			SKWGIF		
		$r=2$	$r=4$	$r=8$	$r=2$	$r=4$	$r=8$	$r=2$	$r=4$	$r=8$
$\varepsilon=0.1^2$	Barbara	30.78	30.13	29.54	31.39	30.60	29.83	31.80	31.34	30.76
	Pirate	31.64	30.10	28.98	31.82	30.24	29.07	32.80	31.17	29.84
	Baboon	28.26	27.73	27.36	28.52	27.94	27.53	28.91	28.32	27.95
$\varepsilon=0.2^2$	Barbara	25.47	24.41	23.23	26.23	25.01	23.63	26.64	25.84	24.55
	Pirate	28.65	26.58	24.82	28.87	26.73	24.93	30.12	27.83	25.79
	Baboon	23.93	23.05	22.41	24.22	23.29	22.60	24.66	23.64	22.95
$\varepsilon=0.4^2$	Barbara	22.46	21.14	19.57	22.95	21.54	19.84	23.28	22.39	20.71
	Pirate	27.13	24.78	22.70	27.27	24.88	22.77	28.70	26.04	23.63
	Baboon	21.84	20.86	20.13	22.10	20.98	20.22	22.46	21.27	20.49

		SSIM								
		GIF			WGIF			SKWGIF		
		$r=2$	$r=4$	$r=8$	$r=2$	$r=4$	$r=8$	$r=2$	$r=4$	$r=8$
$\varepsilon=0.1^2$	Barbara	0.9593	0.9425	0.9366	0.9609	0.9436	0.9367	0.9663	0.9525	0.9403
	Pirate	0.9261	0.8746	0.8524	0.9274	0.8759	0.8533	0.9447	0.8979	0.8570
	Baboon	0.9170	0.8799	0.8654	0.9188	0.8806	0.8649	0.9332	0.8934	0.8675
$\varepsilon=0.2^2$	Barbara	0.9042	0.8556	0.8114	0.9123	0.8644	0.8189	0.9252	0.8905	0.8382
	Pirate	0.8821	0.7802	0.7019	0.8846	0.7837	0.7053	0.9173	0.8291	0.7244
	Baboon	0.8086	0.7062	0.6440	0.8163	0.7143	0.6512	0.8530	0.7452	0.6616
$\varepsilon=0.4^2$	Barbara	0.8383	0.7458	0.6350	0.8493	0.7591	0.6483	0.8698	0.8105	0.6943
	Pirate	0.8533	0.7139	0.5843	0.8553	0.7168	0.5873	0.9002	0.7815	0.6225
	Baboon	0.7123	0.5522	0.4437	0.7189	0.5591	0.4501	0.7769	0.6045	0.4667

A. Edge-Aware Smoothing

As aforementioned, the edge-preserving is an important property of SKWGIF. Here, we evaluate the property on a gray image (Cat), which is given in Fig. 5 (a). And the guidance is also the input itself. The radius r of the local windows and the parameter ε selected here are both followed with [14]. Fig. 6 shows the filtered results of GIF, WGIF and SKWGIF. We observe that SKWGIF can preserve the sharp edges better and reduce the halo artifacts appeared in GIF and WGIF effectively.

The PSNR and SSIM values of the results are reported in Table I. It can be seen that, with the same parameters, SKWGIF shows the best performance among the filters related.

Furthermore, we conduct the experiment on several commonly used images shown in Fig. 7. Table II reports the PSNR and SSIM results. It also substantiates that SKWGIF outperforms GIF and WGIF in edge-aware smoothing.

B. Detail Enhancement

Given an input image p to be enhanced, the existing edge-preserving smoothing filters usually decompose it into a base layer q containing large scale variations in intensity, and a detail layer d capturing the small scale details [5]. This process can be formulated as

$$p = q + d. \tag{19}$$

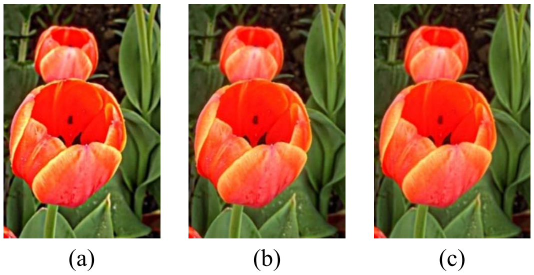

(a) (b) (c)

Fig. 8. Detail enhancement. (a, b, c) Enhanced images obtained by GIF,WGIF and SKWGIF, respectively.

Here, d is defined as the difference between p and q: $d = p - q$. Then, the enhanced image I_{enh} is formulated as the combination of boosted d and q:

$$I_{enh} = q + nd. \tag{20}$$

where n is a positive constant called an amplification factor [18].

When GIF, WGIF and SKWGIF are applied to detail enhancement, the guidance image is the input image (Tulip) itself, which is presented in Fig. 1 (a). In the same way in WGIF, we set the parameters $r = 16$, $\varepsilon = 1/128$ and $n = 5$, respectively. And Fig. 8 shows the enhanced results. Considering the input Tulip is a blurred image, followed with [18], we employ BIQI [25] to evaluate the enhanced

TABLE III
OBJECTIVE EVALUTION ON THE ENHANCED IMAGES IN FIG. 8.

	BIQI		
	GIF	WGIF	SKWGIF
$\varepsilon = 1/128, r = 16$	33.04	33.94	41.45

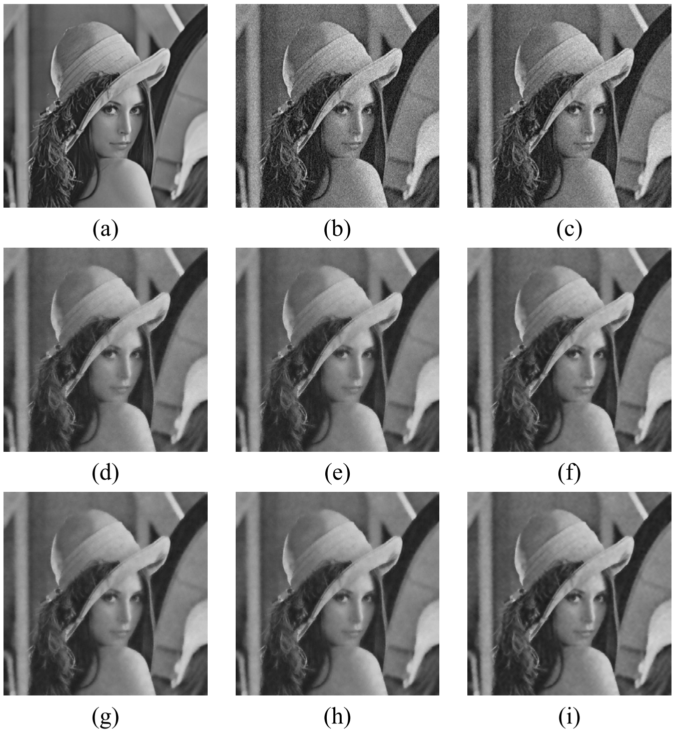


Fig. 9. Image denoising. (a) Original image (Lena). (b) Noisy image (polluted by Gaussian noise with standard deviation 25). (c) Preprocessed result of (b) with a 5 × 5 Gaussian filter. (d, e, f) and (g, h, i) are the outputs of GIF, WGIF and SKWGIF by taking (b) and (c) as the guidance image, respectively.

images and the results are reported in Table III. With the index, a higher value represents a higher quality. Obviously, the SKWGIF gives the best result among the filters related.

C. Image Denoising

To further illustrate the effectiveness of the proposed SKWGIF, we take an image denosing experiment on a noisy image (Lena). For the sake of fairness, we set the parameters with the same values ($r = 4, \varepsilon = 0.2^2$) to compare the performance of GIF, WGIF and SKWGIF. Considering only one image presented in this case, we first adopt a naive method to conduct the experiment, i.e., taking the noisy image itself as the guidance. From equations (10) and (11), we know that the mean (μ) and variance (σ^2) of the guidance image both play important roles in GIF, WGIF and the proposed SKWGIF. To improve the reliability of the two statistical parameters, we preprocess the noisy image and take the result as the guidance to proceed to the experiment. The experimental results are reported in Fig. 9 and Table IV in terms of quantitative evaluation and visual perception. Due to the excellent edge-preserving smoothing property, SKWGIF achieves the best outputs among the three contestents. Meanwhile, the experimental results indicate that a proper preprocessing for guidance can improve the quality of the final results actually. It also demonstrates that the performances of these guided filters are highly dependent on the quality of the guidance image.

TABLE IV
PSNR AND SSIM OF THE RESTORED IMAGES IN FIG. 9.

	Guided with (b)		Guided with (c)	
	PSNR	SSIM	PSNR	SSIM
Noisy image	20.26	0.5902	20.26	0.5902
Guidance	20.26	0.5902	24.04	0.6654
GIF	27.67	0.8505	28.44	0.8602
WGIF	27.83	0.8524	28.57	0.8622
SKWGIF	28.27	0.8631	29.26	0.8749

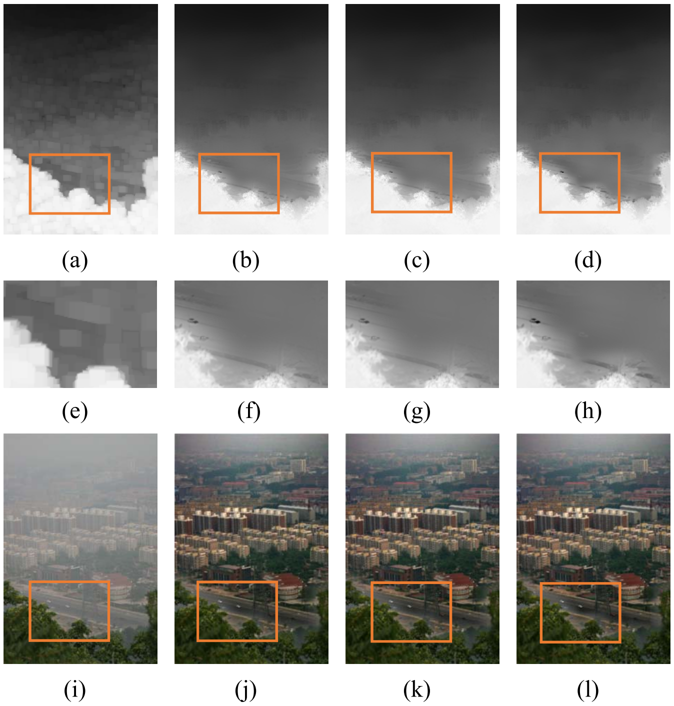


Fig. 10. Single image haze removal. (i) Hazy image (Canon). (a) Raw transmission map [7]. (b, c, d) Refined transmission maps by GIF, WGIF and SKWGIF, respectively. (e, f, g, h) Images are corresponding to (a, b, c, d,), respectively. (j, k, l) Recovered images by using (b, c, d), respectively.

D. Single Image Haze Removal

In [7], He *et al.* employed the dark channel prior to estimate the transmission map, and used it to restore the hazy image. Compared with traditioanal approachs, this method is simpler and more effective. And in [14], they introduced GIF to refine the transmission map and achieved better performance. Similarly, in this experiment, we employ GIF, WGIF and SKWGIF to refine the transmission map, respectively. Followed with [14], the radius r of local window and the regularization parameter ε are set to 20 and 10^{-3} respectively. Fig. 10 shows their performance in the refined results and recovered images. Apparently, in the filters involved, SKWGIF delivers best accuracy in the refined transmission map and provides best recovered image.

E. Discussion on Computational Complexity

As outlined in Algorithm 1, the major computational cost of the proposed SKWGIF comes from the $5th$ step, i.e., learning the steering kernel and computing the weighted average of the linear coefficients. Given an image with N pixels and the local window radius r, it takes about $O\left(Nr^2\right)$ to learn the steering kernel (including estimate the gradient covariance matrix and compute the weights according to equations (4) and (2) respectively). And then, it also takes about $O\left(Nr^2\right)$ to perform the weighted averaging processes. While for the computation of the other steps (the $3th$, $4th$ and $6th$) in Algorithm 1, they are all $O\left(N\right)$ computational complexity. Therefore, the main computational complexity of the proposed SKWGIF is approximately $O\left(Nr^2\right)$. Furthermore, the learning process is generally implemented in iterative style. As shown in steering kernel regression (SKR) [20], the maximum number of iteration t can be up to 15. Thus, it is highly time-consuming compared with the $O\left(N\right)$ complexity of GIF and WGIF.

To improve the speed of SKWGIF, a feasible method is reducing the computational complexity of the learning process of steering kernel. However, this is very challenging and worthy of a specific research [26]. For instance, in terms of SVD employed in SKWGIF, although there are some acceleration algorithms [27], [28], they are mainly for the large scale matrices, and not suitable for the small scale local gradient matrices in the steering kernel. Thus, in this work, to speed up SKWGIF, we select another way, i.e., modifying algorithm implementation.

Specifically, considering the credibility of the guidance image, we adopt non-iterative method to address the computational bottleneck of SKWGIF. Meanwhile, with the help of SPMD (Single Program, Multiple Date), the parallel style is implemented to further speedup SKWGIF. Finally, compared with it in [20], the complexity for learning steering kernel has been reduced to $1/kt$ times theoretically in our SKWGIF (k is the amount of CPU cores).

Due to its excellent performance and efficiency in denoising, BM3D filter [29] has been widely approved in practice. To verify the efficiency of SKWGIF, we conduct the experiment with SKWGIF, BM3D and other three filters related, including SKR [20], GIF and WGIF. The local window radius r in the guided filters (GIF, WGIF and SKWGIF) is appointed from 2 to 20. For the sake of fairness, we adopt the non-iternative method in SKR, which is same as that of in SKWGIF. All the filters are implemented in MATLAB R2013b, and perform on a PC equipped with Intel Quad Core 2.1GHz and 16GB of RAM. Fig. 11 shows the CPU time they spent on a 512 × 512 image. It can be seen that, due to their linear time complexity and independence to the radius, GIF and WGIF become the most efficient methods among the contestants. Meanwhile, with the help of the naïve accelerating strategies mentioned above, our SKWGIF reduced the run time significantly compared with SKR. However, it cannot be denied that there is still a large gap between SKWGIF and its two guided counterparts. Thankfully, it is a kind of comfort that SKWGIF has a competitive run time with BM3D, a state-of-the-art denoising filter. To be noted, like GIF and WGIF, SKWGIF is also versatile, which can be used in many of applications (beyond denoising). We also noticed that, a more efficient learning-steering-kernel method with Graphical Processing Unit (GPU) has been proposed in [30], which can be borrowed to further accelerate SKWGIF directly.

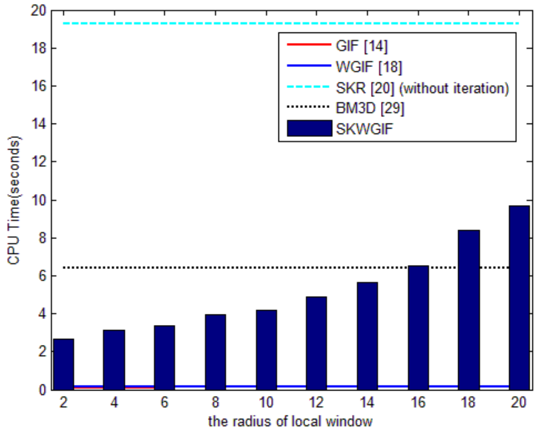


Fig. 11. Comparsion of CPU time of different methods.

V. Conclusion

In this paper, we presented a novel image edge-preserving smoothing method, the weighted guided image filtering with steering kernel (SKWGIF). In contrast to guided image filtering (GIF) and weighted guided image filtering (WGIF), SKWGIF employs steering kernel to learn the local structure prior of guidance image and integrates it into the filtering process to enhance the filters performance in the edges. Experimental results in edge-aware smoothing, detail enhancement, denoising and dehazing all have indicated the particularly effectiveness of SKWGIF. It should be noted that, due to the complexity of steering kernel regression algorithm itself, SKWGIF has more complexity than GIF and WGIF inevitably. To further accelerate SKWGIF is worthy investigating in future work.

Acknowledgment

The authors would like to thank the editor and anonymous reviewers for their insightful comments and suggestions which have greatly improved the paper.

References

[1] P. Charbonnier, L. Blanc-Feraud, G. Aubert, and M. Barlaud, “Deterministic edge-preserving regularization in computed imaging,” *IEEE Trans. Image Process.*, vol. 6, no. 2, pp. 298–311, Feb. 1997.

[2] M. Zhang and B. K. Gunturk, “Multiresolution bilateral filtering for image denoising,” *IEEE Trans. Image Process.*, vol. 17, no. 12, pp. 2324–2333, Dec. 2008.

[3] B. Zhang and J. P. Allebach, “Adaptive bilateral filter for sharpness enhancement and noise removal,” *IEEE Trans. Image Process.*, vol. 17, no. 5, pp. 664–678, May 2008.

[4] R. Fattal, M. Agrawala, and S. Rusinkiewicz, “Multiscale shape and detail enhancement from multi-light image collections,” *ACM Trans. Graph.*, vol. 26, no. 3, p. 51, 2007.

[5] Z. Farbman, R. Fattal, D. Lischinshi, and R. Szeliski, "Edge-preserving decompositions for multi-scale tone and detail manipulation," *ACM Trans. Graph.*, vol. 27, no. 3, p. 67, Aug. 2008.

[6] S. Bae, S. Paris, and F. Durand, "Two-scale tone management for photographic look," *ACM Trans. Graph.*, vol. 25, no. 3, pp. 637–645, 2006.

[7] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 12, pp. 2341–2353, Dec. 2011.

[8] S. Li, X. Kang, and J. Hu, "Image fusion with guided filtering," *IEEE Trans. Image Process.*, vol. 22, no. 7, pp. 2864–2875, Jul. 2013.

[9] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 7, pp. 629–639, Jul. 1990.

[10] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. IEEE Int. Conf. Comput. Vis.*, Jan. 1998, pp. 836–846.

[11] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama, "Digital photography with flash and no-flash image pairs," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 664–672, Aug. 2004.

[12] S. Paris and F. Durand, "A fast approximation of the bilateral filter using a signal processing approach," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 568–580.

[13] E. S. L. Gastal and M. M. Oliveira, "Domain transform for edge-aware image and video processing," *ACM Trans. Graph.*, vol. 30, no. 4, p. 69, 2011.

[14] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 6, pp. 1397–1409, Jun. 2013.

[15] Y. Li, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep joint image filtering," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2016, pp. 154–169.

[16] S. Gu, W. Zuo, S. Guo, Y. Chen, C. Chen, and L. Zhang, "Learning dynamic guidance for depth image enhancement," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 712–721.

[17] H. Wu, S. Zheng, J. Zhang, and K. Huang, "Fast end-to-end trainable guided filter," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2018, pp. 1838–1847.

[18] Z. Li, J. Zheng, Z. Zhu, W. Yao, and S. Wu, "Weighted guided image filtering," *IEEE Trans. Image Process.*, vol. 24, no. 1, pp. 120–129, Jan. 2015.

[19] V. Katkovnik, A. Foi, K. Egiazarian, and J. Astola, "From local kernel to nonlocal multiple-model image denoising," *Int. J. Comput. Vis.*, vol. 86, no. 1, pp. 1–32, 2010.

[20] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Trans. Image Process.*, vol. 16, no. 2, pp. 349–366, Feb. 2007.

[21] H. Takeda, S. Farsiu, and P. Milanfar, "Deblurring using regularized locally adaptive kernel regression," *IEEE Trans. Image Process.*, vol. 17, no. 4, pp. 550–563, Apr. 2008.

[22] K. Zhang, X. Gao, D. Tao, and X. Li, "Single image super-resolution with non-local means and steering kernel regression," *IEEE Trans. Image Process.*, vol. 21, no. 11, pp. 4544–4556, Nov. 2012.

[23] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, Nov. 1998.

[24] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[25] A. K. Moorthy and A. C. Bovik, "A two-step framework for constructing blind image quality indices," *IEEE Signal Process. Lett.*, vol. 17, no. 5, pp. 513–516, May 2010.

[26] H. J. Seo and P. Milanfar, "Using local regression kernels for statistical object detection," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2008, pp. 2380–2383.

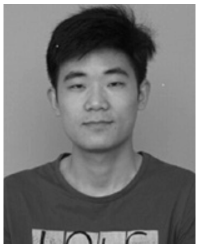
[27] M. Holmes, A. Gray, and C. Isbell, "Fast SVD for large-scale matrices," in *Proc. Workshop Efficient Mach. Learn. (NIPS)*, Dec. 2007, pp. 249–252.

[28] M. Kurucz, A. Benczúr, and K. Csalogány, "Methods for large scale SVD with missing values," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, Aug. 2007, pp. 31–38.

[29] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.

[30] V. Sairam, M. S. Rao, G. D. Khalandhar, L. Srikanth, P. K. Baruah, and R. R. Sarma, "A computationally efficient parallel kernel regression for image reconstruction," in *Proc. IEEE Int. Conf. High Perform. Comput.*, 2012, pp. 1–5.

Zhonggui Sun received the B.Sc. degree in mathematics from Shandong Normal University, China, in 1996, the M.Sc. degree in computer software and theory from Sichuan Normal University, China, in 2006, and the Ph.D. degree in computer applications from the Department of Computer Science and Engineering, Nanjing University of Aeronautics and Astronautics (NUAA), China, in 2014. He is currently an Associate Professor with the Department of Mathematics Science, Liaocheng University (LU). His research interests include image processing and machine learning.



Bo Han received the B.Sc. degree in mathematics from Binzhou University, Binzhou, China, in 2017. He is currently pursuing the M.Sc. degree in mathematics with Liaocheng University, Liaocheng, China. His research interests include image processing and machine learning.

Jie Li received the B.Eng. degree in electronic engineering, the M.Sc. degree in signal and information processing, and the Ph.D. degree in circuit and systems from Xidian University, Xi'an, China, in 1995, 1998, and 2004, respectively. She is currently a Professor with the School of Electronic Engineering, Xidian University. Her research interests include image processing and machine learning. In these areas, she has published around 50 technical articles in refereed journals and proceedings including IEEE T-NNLS, T-IP, T-CSVT, and *Information Sciences*.

Jin Zhang received the B.Sc. degree in mathematics from Taishan College (TC), China, in 2004, the M.Sc. degree in basic mathematics from Liaocheng University (LU), China, in 2009, and the Ph.D. degree in computational mathematics from the Department of Mathematical Science, Dalian University of Technology (DUT), China, in 2017. She is currently a Lecturer with the Department of Mathematics Science, LU. Her research interests include nonlinear optimization for image processing and machine learning.

Xinbo Gao (M'02–SM'07) received the B.Eng., M.Sc., and Ph.D. degrees in signal and information processing from Xidian University, Xi'an, China, in 1994, 1997, and 1999, respectively. From 1997 to 1998, he was a Research Fellow with the Department of Computer Science, Shizuoka University, Shizuoka, Japan. From 2000 to 2001, he was a Post-Doctoral Research Fellow with the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong. Since 2001, he has been with the School of Electronic Engineering, Xidian University. He is currently a Cheung Kong Professor of Ministry of Education, a Professor of pattern recognition and intelligent system, and the Director of the State Key Laboratory of Integrated Services Networks, Xi'an, China. He has published six books and around 200 technical articles in refereed journals and proceedings. His current research interests include multimedia analysis, computer vision, pattern recognition, machine learning, and wireless communications. He is a fellow of the Institute of Engineering and Technology and the Chinese Institute of Electronics. He served as the General Chair/Co-Chair, the Program Committee Chair/Co-Chair, or a PC Member for around 30 major international conferences. He is on the Editorial Boards of several journals, including *Signal Processing* (Elsevier) and *Neuro Computing* (Elsevier).