Anisotropic Joint Trilateral Rolling Filter for Image Smoothing

XIAOHUA ZHANG^{*†}
NING XIE[§]

Member,
Non-member

YUELAN XIN[‡] Non-member

(Received June 1, 2019, revised July 6, 2019)

Abstract: Filtering an image by eliminating noise and irrelevant details while preserving prominent structure edges is an important pre-processing task in many fields, such as image processing and computer vision. In this study, a novel approach called anisotropic joint trilateral rolling filter (AJTRF) is proposed for the smoothing of an image while preserving important structure edges. AJTRF works in an iterative manner and extends the joint bilateral filter to a joint trilateral filter by employing additional weights in eigen space. During filtering, the range weights are recalculated in each iteration, while the spatial Gaussian weights are computed in anisotropic orientations instead of isotropic orientations. Furthermore, the inner products between anisotropic orientations (expressed as eigenvectors) are used as weights to measure orientation similarities of pixels. Structure tensors at each pixel are calculated to determine the structure orientations, which are used as the anisotropic orientations. A range of experiments are conducted. The results demonstrate that the proposed AJTRF has stronger smoothing and structure preserving ability than established methods.

Keywords: Image smoothing, Edge-preserving, Rolling guidance filter, Anisotropic joint trilateral rolling filter, Structure tensor, Eigen space

1. Introduction

Natural imagery contains many rich visual structures at different scales. However, in the majority of applications (including image abstraction, image segmentation, and object detection *etc.*) small-scale textures are less important while prominent structures are essential. Eliminating small scale textures while preserving the main structures can be achieved by an image smoothing filter. As an important component of pre-processing, image smoothing is a widely used operation in fields such as image processing and computer vision [1]. The efficiency of most of highlevel processing depends on it. There exists a lot of traditional smoothing methods such as the box filter, weighted average filter, Gaussian filter, and bilateral filter [3] working in the spatial domain. There is also the low-pass filter in the frequency domain that is used to smooth the image by removing high frequency components. However, although these traditional smoothing filters remove finer textures and noises, most of them (except for the bilateral filter) also blur edges in the images, that is, the salient structures are lost.

One of the purposes of image smoothing is to remove the unwanted small features that may represent finer textures, noises and small objects. Another goal is to preserve the prominent structures such as region boundaries and large objects [2]. A wide range of such filters have been recently reported for edge-preserving smoothing, which re-

move noise, small scale textures, and weak edges while preserving strong structures. These filters, often called edgeaware filters, include the bilateral filter [3] and its inherited variant, the joint bilateral [4], the L_0 gradient minimization filter [2], the guided image filter [5], the domain transform filter [6], the rolling guidance filter [7], and its variant the alternating guided filter [8].

In this paper, a new edge-preserving image smoothing filter called anisotropic joint trilateral rolling filter (AJTRF) is proposed. To some extent, the AJTRF can be regarded as an extension of the rolling guidance filter [7]. Unlike the original rolling guidance filter, which uses a Gaussian filter to remove small structures, the AJTRF combines the anisotropic Gaussian weight and inner product of eigenvectors of the structure tensor to remove small textures and then restore the prominent structures. The anisotropy is computed based on the structure tensor at each pixel. Numerous experiments demonstrate that the AJTRF has strong smoothing and edge-preserving ability compared to the original rolling guidance filter. With the same standard deviation parameters, the AJTRF is even capable of edge enhancement.

The contributions of this paper are as follows:

- The structure tensor at each pixel is computed, and then the eigenvector is used to represent the orientation of that pixel.
- A trilateral filter is proposed by combining the weights of the anisotropic Gaussian in spatial space, the inner product of the eigenvector in eigen space, and the Gaussian in range space.

* Corresponding author: zhxx@cc.it-hiroshima.ac.jp

† Hiroshima Institute of Technology, Hiroshima, Japan.

‡ Qinghai Normal University, Xining, China.

§ University of Electronic Science and Technology of China, Chengdu, China.

Figure 1: Data flow of the proposed anisotropic joint trilateral rolling filter.

- The trilateral filter is embedded in the guided rolling filter framework to smooth finer textures while preserving prominent structures.

Figure 1 illustrates a data flow in the proposed filter. A rectangle with rounded corners shows the input or output image. Other rectangles indicate data during computation. Solid arrows demonstrate data flow directions and a circled arrow expresses that the data flows in and out from itself.

In Section 2, the relevant literature is briefly reviewed. Because the proposed filter is based on the bilateral filter and rolling guidance filter, these are described in Section 3. The structure tensor computation, trilateral filter, and the AJTRF are detailed in Section 4. Section 5 demonstrates the experimental results and discusses the properties of the AJTRF. Finally, the conclusion and future work are given in Section 6.

2. Related Work

A wide range of edge-preserving filters have recently been reported. These filters detect the differences between neighboring pixels and use such differences to compute weights or classify them as strong or weak edges. These filters can be roughly categorized into two types: optimization-based and local average-based. Optimization-based methods, such as the total variation [9], the weighted least squares [10] [11], and the L_0 gradient minimization [2] [12]~[14] optimize the designed energy function which consists of a data fidelity term and a smoothing regularized term. The data term describes the fidelity of the smoothed result to the original image and is represented by the total squared errors between the resultant and original images. The regularized term can be the L_0 , L_1 , and L_2 norms. As indicated in Ref. [14], because optimization-based approaches globally control the gradient of an image by minimizing the energy function, they have a strong capability for preserving the main structures in the image. However, they are less effective in dealing with local and high contrast noises.

Local average-based methods, such as the bilateral filter [3] [4], guided filter [5] [8], and domain transform filter [6] are widely used for their simplicity and effectiveness in removing small details and noise. These types of approach trade-off between removing small scale textures and preserving strong edges. Hence, the majority of these filters blur edges and cannot preserve the strong edges well.

Most closely related to the filter proposed here are the bilateral filter and the rolling guidance filter. The bilateral

filter [3] [4] is a broadly used non-linear filter that calculates the output at each pixel as a Gaussian weighted average of pixels in a window according to their geometric distances and intensity differences. A pixel close to the center of the window has large weight, and a pixel having small intensity difference with the center pixel also has a large contribution to the output. Conversely, a pixel far from the center (or having large intensity difference) has a small contribution to the output. However, the bilateral filter has a defect that tends to blur strong edges. A recently reported rolling guidance filter (RGF) [7] [8] [15] is in fact an application of a joint bilateral filter in an iterative framework. It consists of two steps: a small structure removal step and an edge recovery step. The rolling guidance filter is a framework used to filter an image with control of detail smoothing under a scale measure. It uses an isotropic Gaussian filter to remove small structures and all pixels are processed equally in the first step. This leads to some structure edges not being recovered effectively in the second step.

This paper is also related to the anisotropy [16] in the filtering. An anisotropic Gaussian in spatial space and an inner product of vectors in eigen space are combined to remove small scale textures according to their orientations computed using structure tensors [17]. The prominent structures are then recovered iteratively using the joint trilateral filter.

3. Rolling Guidance Filter

The original RGF [7] uses an iterative framework that converges quickly. It is an application of the joint bilateral filter [4]. Note, some authors refer to it as the cross bilateral filter. The word “joint” is used in this paper to maintain consistency with Ref. [4]. The filter processes images by controlling the detailed smoothing under a scale measure. The RGF is composed of two steps: small structure removal by Gaussian filter and large edge recovery by joint bilateral filter. Since Gaussian filtering is strongly related to structure scale determination, RGF is powerful for removing small structures. Suppose the input image is denoted as I and the output image is G , any pixel location can be represented as a row vector using 2D coordinates $p = (x, y)$. Given a standard deviation σ_s , and any pixel q in a neighborhood region Ω centered at pixel p , the Gaussian filter in spatial space may be expressed as:

$$G(p) = \frac{1}{K_p} \sum_{q \in \Omega} \exp\left(-\frac{\|p - q\|^2}{2\sigma_s^2}\right) I(q) \quad (1)$$

where K_p is a normalization term to guarantee the sum of the Gaussian weights in the kernel window to be 1.0. If the scale of a structure in the given image is smaller than the standard deviation, this structure will be removed by the Gaussian filter. It is known that Gaussian filtering removes small scale structure. However, it also blurs strong edges and large scale structures.

Although Gaussian filtering blurs large scale structures, it only weakens these edges and structures. It does not eliminate them, and structural information is retained in the output. The second step of the original RGF is to recover these

edges with large scale structure in the original image. The RGF applies an iterative scheme to update the result in the t th iteration. Given the input I and the previous result image U^{t-1} , the updated image U^t in the t th iteration is computed by a joint bilateral filter, as shown in the following:

$$U^t(p) = \frac{1}{K_p} \sum_{q \in \Omega} \exp\left(-\frac{\|p - q\|^2}{2\sigma_s^2}\right) \exp\left(-\frac{\|U^{t-1}(p) - U^{t-1}(q)\|^2}{2\sigma_r^2}\right) I(q) \quad (2)$$

where K_p is similar to that in Eq. (1) and is a normalization term represented as:

$$K_p = \sum_{q \in \Omega} \exp\left(-\frac{\|p - q\|^2}{2\sigma_s^2}\right) \exp\left(-\frac{\|U^{t-1}(p) - U^{t-1}(q)\|^2}{2\sigma_r^2}\right) \quad (3)$$

The standard deviation parameter σ_s is the same as that given in Eq. (1), which controls the spatial weight for each pixel q in the kernel window Ω , while parameter σ_r controls the range weight in the same kernel window. During filtering, each iteration uses the previous result as a guided image. U^0 in Eq. (2) is initialized as a Gaussian filtered result G in Eq. (1), and it can also be initialized as a constant image C in practice. It is observed that Eq. (2) always filters the original input image I , but changes the guidance image U^{t-1} , which contains large structure information existing in the original image and with less small scale textures being smoothed.

4. Anisotropic Joint Trilateral Rolling Filter

Although the RGF performs well in most cases, the Gaussian multiplier term for the spatial domain is an isotropic one and has limited smoothing ability. The local directionality in a local window may have more consistency than the intensity variation. The original RGF does not consider the pixel orientation in the local window. This leads to weak smoothing ability for areas with abundant textures. To solve this problem, the AJTRF employs an anisotropic Gaussian filter and the inner product of the eigenvectors representing the orientations in the local window (instead of the isotropic Gaussian filter in spatial domain). This is implemented by changing the isotropic Gaussian term in Eq. (2) to a combination of anisotropic Gaussian term in the spatial domain and the inner product in eigen space. Vectors $\pm v$ are considered as having the same direction. Hence, the word ‘‘orientation’’ is used instead of ‘‘direction’’.

4.1 Structure Tensor To express the orientation at each pixel, the gradient vector at this pixel can be used as the orientation. However, the gradient vectors are prone to be affected by noises. Instead, the structure tensors were computed and orientation information was collected from the tensors to reduce the effects of noises. A structure tensor is calculated at a pixel p as $T(p) = g^T(p)g(p)$, where $g(p) = (I_x(p), I_y(p))$ is the gradient vector at pixel p . The resultant structure tensor is a symmetrical matrix and has

the following form:

$$T(p) = \begin{pmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{pmatrix} \quad (4)$$

Note, a structure tensor is always a symmetric 2×2 matrix and the smallest eigenvalue is always 0. In practice, the tensor is pre-smoothed using a Gaussian kernel with a spatial standard deviation σ_t to make the tensor insensitive to noise and to collect the structure orientation information from neighborhood pixels in the kernel window. In all the experiments conducted here, the standard deviation was allowed to be the same as in Eq. (2), that is, $\sigma_t = \sigma_s$. Because the tensor is a 2×2 symmetrical matrix, its eigenvalues $\mu_1, \mu_2 (\mu_1 \geq \mu_2 \geq 0)$ and eigenvector u, v are easily computed analytically. Eigenvector u is the major vector and corresponds to eigenvalue μ_1 . The eigenvalues are expressed as:

$$\mu_{1,2} = \frac{I_x^2 + I_y^2 \pm \sqrt{(I_x^2 - I_y^2)^2 + 4(I_x I_y)^2}}{2} \quad (5)$$

Accordingly, the corresponding eigenvectors are $u = (I_x I_y, \mu_1 - I_x)$ and $v = (\mu_1 - I_x, -I_x I_y)$. The eigenvector u is the orientation with the highest gray value fluctuations, while eigenvector v shows the preferred local orientation. It is noted that a matrix can be expressed mathematically by its eigenvalues and eigenvectors as:

$$T(p) = \mu_1 u^T u + \mu_2 v^T v \quad (6)$$

The eigenvalues and eigenvectors contain very important structure information in the kernel window, as described in Ref. [18] [20]. After filtering, it is beneficial if the structure in the local kernel window has coherence orientation as that in the original image. To this end, the eigenvectors u, v representing the orientations are unchanged, but the eigenvalues μ_1, μ_2 are modified as:

$$\lambda_1 = \begin{cases} \gamma + (1 - \gamma) \exp\left(-\frac{\delta}{(\mu_1 - \mu_2)^2}\right) & , \mu_1 > \mu_2 \\ \gamma & , \mu_1 = \mu_2 \end{cases} \quad (7)$$

$$\lambda_2 = \gamma$$

where parameter γ is as $0 < \gamma \ll 1$, and both λ_1 and λ_2 are restricted in the range $[\gamma, 1]$, and parameter δ can be fixed as 0.00001. Using the modified eigenvalues, the structure tensor in Eq. (6) can be rewritten as a semi-definite symmetric matrix $A(p)$ in the same form:

$$A(p) = \lambda_1 u^T u + \lambda_2 v^T v \quad (8)$$

The above matrix $A(p)$ will be embedded in our anisotropic trilateral filter for enhancing large scale structure. Since the eigenvalues are converted from μ_1, μ_2 to λ_1, λ_2 by Eq. (7), the proportionality between μ_1, μ_2 is scaled up when $\mu_1 > \mu_2$. This leads to the enhancement of anisotropy during filtering.

Algorithm 1: Anisotropic joint trilateral rolling filter (AJTRF)

Input:	original image I , standard deviations σ_s, σ_r , and eigenvalue parameter γ , the number of iteration N
Output:	the result image U
1.	For all pixels p in image I , compute gradients and find their structure tensors $T(p)$ as in Eq. (4) and then smooth them. After computing the eigenvalues and eigenvectors, adjust the eigenvalues as λ_1, λ_2 using Eq. (7). Finally, compute matrices $A(p)$ in Eq. (8) and orientation similarities.
2.	Initialize U^0 as an anisotropic Gaussian filtered result of I , or as a constant image.
3.	Do iterations to compute U^t using Eq. (11) and (12). Note, I is always used as input. U^{t-1} is used as guidance for joint bilateral filtering.
4.	Substitute the final result of the above iteration into U^{N-1} into the output image U .

4.2 Anisotropic Trilateral Filter In the proposed AJTRF, the matrix $A(p)$ in Eq. (8) was embedded into the Gaussian term in Eq. (1) for each pixel. Moreover, the orientation similarity describing the consistency between orientations of pixels was also combined into the filter. The weight in the range space is the same as that in the traditional bilateral filter.

Note, the squared distance $\|p - q\|^2$ between pixel p and q in the spatial domain can be expressed as $(p - q)(p - q)^T$ in vector form, where T denotes the matrix or vector's transpose. The method for computation of squared distance is the same for all pixels q without regard for the orientation. The pixels in the neighborhood may have different weights if their orientations are different with that of pixel at the window center. The two perpendicular eigenvectors of the tensor were used to span a region. Consequently, the shape of the distribution of weights is an elliptical region because the two eigenvalues are different. Hence, the region is an anisotropic one, instead of a circular dome that is isotropic. Inspired by the work of [17] [18], the anisotropic property is obtained by defining the squared distance as $(p - q)A(p)(p - q)^T$, where $A(p)$ is a semi-definite symmetric matrix that is defined in an elliptical region centered at pixel p , as shown in Eq. (8).

Because the matrix $A(p)$ only constrains the filter in an elliptical region, the orientations at pixels in this elliptical region may be different. To enhance the coherence of pixels, an inner product of the eigenvectors at pixel p and q as a weight is also embedded into the filter to evaluate the similarity of orientations. Suppose the tensors $T(p)$ and $T(q)$ have eigenvectors $\mathbf{u}_p$ and $\mathbf{u}_q$ corresponding to the largest eigenvalues, respectively, the inner product is $w_c = \langle \mathbf{u}_p, \mathbf{u}_q \rangle$. Note, all eigenvectors are unit vectors. The inner product is the cosine of the angle between the two eigenvectors, which can be rewritten as $w_c = \cos(\mathbf{u}_p, \mathbf{u}_q)$. Because the inner product or the cosine may be negative, its absolute is taken as the weight in practice. The weight w_c is a measurement to describe how two vectors are similar. When w_c is very large (near to 1.0), the two eigenvectors $\mathbf{u}_p$ and $\mathbf{u}_q$ are similar in orientation. In contrast, if it is very small (near to 0.0), the two vectors are almost perpendicular to each other, that is, they are very different.

Combining the anisotropic Gaussian weights and the orientation similarity measures with the weights in the range space, given an original input image, the output of trilateral

filter is expressed as:

$$J(p) = \frac{1}{K_p} \sum_{q \in \Omega} \exp\left(-\frac{(p - q)A(p)(p - q)^T}{2\sigma_s^2}\right) \cdot |\cos(\mathbf{u}_p, \mathbf{u}_q)| \exp\left(-\frac{\|I(p) - I(q)\|^2}{2\sigma_r^2}\right) I(q) \quad (9)$$

and

$$K_p = \sum_{q \in \Omega} \exp\left(-\frac{(p - q)A(p)(p - q)^T}{2\sigma_s^2}\right) \cdot |\cos(\mathbf{u}_p, \mathbf{u}_q)| \exp\left(-\frac{\|I(p) - I(q)\|^2}{2\sigma_r^2}\right) \quad (10)$$

where the notations of σ_s and σ_r are standard deviation as in Eq. (2). $I(p)$ and $I(q)$ are pixel values or colors at pixel p and q , respectively.

4.3 Anisotropic Joint Trilateral Rolling Filter After preparation of the anisotropic trilateral filter, it was subsequently embedded into the RGF to construct an AJTRF. As indicated in the original RGF [7], an iterative scheme was also used for image smoothing and edge recovery. The AJTRF is expressed as follows using the notation from the above equations Eq. (9) and (10).

Then, Eq. (2) and Eq. (3) can be rewritten as:

$$U^t(p) = \frac{1}{K_p} \sum_{q \in \Omega} \exp\left(-\frac{(p - q)A(p)(p - q)^T}{2\sigma_s^2}\right) \cdot |\cos(\mathbf{u}_p, \mathbf{u}_q)| \exp\left(-\frac{\|U^{t-1}(p) - U^{t-1}(q)\|^2}{2\sigma_r^2}\right) I(q) \quad (11)$$

and

$$K_p = \sum_{q \in \Omega} \exp\left(-\frac{(p - q)A(p)(p - q)^T}{2\sigma_s^2}\right) \cdot |\cos(\mathbf{u}_p, \mathbf{u}_q)| \exp\left(-\frac{\|U^{t-1}(p) - U^{t-1}(q)\|^2}{2\sigma_r^2}\right) \quad (12)$$

The difference between Eq. (11), (12), and Eq. (2), (3) is that the semi-definite symmetric matrix $A(p)$ is incorporated into the Gaussian multiplier term for the spatial domain. The inner product between eigenvectors is also used as a weight to evaluate the orientation similarity between two pixels. In the above equations, $U^t(p)$ is the result of pixel p at the t th iteration. Note, a previous filtered result

Figure 2: Experimental results: (a) original images; (b) result of rolling guidance filter, corresponding to (a); (c) result of anisotropic rolling filter without orientation similarity; and (d) result of the proposed anisotropic joint trilateral rolling filter corresponding to (a).

Figure 3: Experimental results: (a) original images; (b) result of rolling guidance filter; (c) result of L_0 gradient minimization for comparison; (d) result of the proposed anisotropic joint trilateral rolling filter.

$U^{t-1}(p)$ is used as a guide for the range space during filtering.

When incorporating the above matrix $A(p)$ into the Gaussian function in the spatial domain, as shown in Eq. (11), the region of Gaussian weights becomes an elliptical region

instead of a circular domain. Because the weights distribute along with the main orientation, it is obvious that the first term in Eq. (11) is an anisotropic function. Moreover, the absolute cosine computed as an inner product of eigenvectors measures the orientation similarity. Thus, the filter de-

Figure 4: Experimental results: (a), (d), and (g) are original images; (b), (e), and (h) are results of rolling guidance filter, corresponding to the left original images; (c), (f), and (i) are results of the proposed anisotropic joint trilateral rolling filter, corresponding to the leftmost images.

Figure 5: The averaged pixel differences between iterations.

veloped was called the anisotropic joint trilateral rolling filter, abbreviated to AJTRF. Note, when the matrix $A(p)$ becomes an identity matrix and the absolute cosine term is ignored, the AJTRF is identical to the original RGF. The working procedure of the proposed AJTRF is presented in Algorithm 1.

5. Experimental Results

To verify the effectiveness of the proposed filter, a range of experiments have been conducted on different images. The

proposed algorithm, written in the C language, was implemented on a Notebook computer with Windows 10. Before the processing, all pixel values were normalized in the range $[0, 1]$. Figure 2 shows one of the experimental results with a natural scene in Fig. 2(a), which is an original color image containing sky, trees, a house and grass. (b) is the result of the original RGF, while (c) is the result of the AJTRF without orientation similarity weight. (d) is the result of the AJTRF with similarity weight. In the experiment which produced Fig. 2(b), (c), and (d), the same spatial standard deviation $\sigma_s = 3.0$ and range standard deviation $\sigma_r = 0.05$ were used. For producing the results of Fig. 2(c) and (d), additional parameters, standard deviation $\sigma_t = 3.0$ used for smoothing tensors, $\gamma = 0.01$ and $\delta = 0.00001$ for eigenvalues adjustment. All results are produced by running the algorithms for 5 iterations. By visually observing (b) and (d), the proposed filter smoothed more details while preserving and enhancing more prominent structures. Note, the white square on the red wall of the barn in Fig. 2(c) has some spurs, which are not present in Fig. 2(d). This is believed to be due to the embedding of the orientation similarity measurement into the filter, which cancels the spurs.

Figure 3 is another experimental result. Figure 3(a) is

Figure 6: Experimental results: (a) and (d) are original images; (b) and (e) are images with added Gaussian noise; (c) and (f) are non-photo-realistic results of the proposed anisotropic joint trilateral rolling filter.

an original image with finer texture and large scale structure. Figure 3(b) is the result of the RGF and (d) is the result produced by the AJTRF. The algorithm also runs for 5 iterations. The parameters are the same as those used in Fig. 2 except for $\sigma_r = 0.2$. Both algorithms recovered the large structures well. However, the AJTRF removes the small-scale textures better. The result of the L_0 gradient minimization [2] is also given in Fig. 3(c) for comparison. The regularization parameter $\lambda = 0.18$ produces the best result. Because L_0 gradient minimization tries to preserve the limited strong edges and flatten the small-scale textures, many short strong edges are retained in the result and black grooves are averaged with the neighborhood pixels.

Figure 4 demonstrates more experimental results. The first column shows original images; the second column is the results from the RGF; the third column shows results from the AJTRF. All results are produced by running 5 iterations. The standard deviation is $\sigma_s = 3.0$ in spatial space for all rows; $\sigma_r = 0.075$ for the first two rows, while $\sigma_r = 0.15$ for the last row. The standard deviation $\sigma_t = 3.0$ for tensor smoothing. Note, the algorithm converges in approximately 5 iterations as indicated in Fig. 5, which describes the variations of the average pixel absolute differences between iterations for Fig. 2.

The above experimental results demonstrate that the proposed AJTRF outperforms the original RGF when using the same parameters. This is believed to be due to the anisotropy of the proposed AJTRF, which captures more orientation consistent information during filtering. Because the anisotropic Gaussian filter smooths the image along with the structure orientation, only small details are removed while most of the prominent structures are preserved according to the setting of parameters. At each iteration, the joint trilateral filter gradually recovers the removed large edges and corners.

The AJTRF also has an interesting application for producing non-photo-realistic rendering images, as shown in Fig. 6. In this figure, (a) and (d) are original images, (b) and (e) are images with added Gaussian noise; while (c) and (f) are rendered images with stream flow patterns obtained when using small standard deviation for spatial space. Both results are produced by setting the standard deviation $\sigma_s = 0.5$ in spatial space, $\sigma_r = 0.2$ in range space, and $\sigma_t = 10.0$ in tensor space. The stream patterns visualize the potential flow in the images.

6. Conclusions

In this paper, a new image filter was presented called the AFTRF, which can be used for image smoothing -removal of small scale structure- while preserving prominent edge structure. The filter can be regarded as an extension of the original RGF using structure tensors to capture orientation. The experimental results demonstrated that the proposed AJTRF has stronger smoothing and edge preserving ability than the original RGF when given the same parameter values. Methods to accelerate the anisotropic computation are under consideration for future work.

Acknowledgment

This work is part of the research supported by the Nature Science Foundation of Qinghai Province under Grant No. 2016-ZJ-745 and the National Nature Science Foundation of China under Grant Nos. 61662062 and 61602088.

References

- [1] R. C. Gonzalez and R. E. Woods, "Digital Image Processing", Pearson Prentice Hall, New Jersey, Third Edition, 2010.

- [2] L. Xu, C. Lu, Y. Xu and J. Jia, "Image smoothing via L_0 gradient minimization", *ACM Transaction on Graphics (TOG)*, 30(6), 174, 2011.
- [3] C. Tomasi, and R. Manduchi, "Bilateral filtering for gray and color images", *ICCV* 839-846, 1998.
- [4] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe and K. Toyama, "Digital photograph with flash and no-flash image pairs", *ACM Transaction on Graphics (TOG)*, 23(3), 664-672, 2004.
- [5] K. He, J. Sun and X. Tang, "Guided image filter", *Computer Vision-ECCV 2010*, pp.1-14, 2010.
- [6] E. S. L. Gastal and M. M. Oliveira, "Domain transform for edge-aware image and video processing", *ACM Transaction on Graphics (TOG)*, 30(4), 69, 2011.
- [7] Q. Zhang, X. Shen, L. Xu and J. Jia, "Rolling guidance filter", *Computer Vision-ECCV 2014*, pp.851-830, 2014.
- [8] A. Toet, "Alternating guided image filter", *PeerJ Computer Science*, DOI 10.7717/peerj-cs.72, 2016.
- [9] L. I. Rudin, S. Osher and E. Fatemi, "Nonlinear total variation based noise removal algorithms", *Physica D, Nonlinear Phenom*, 60(1-4), 259-268, 1992.
- [10] Z. Farberman, R. Fattal, D. Lischinski and R. Szeliski, "Edge-preserving decompositions for multi-scale tone and detail manipulation", *ACM Transaction on Graphics (TOG)*, 27(3), Article No. 67, 2008.
- [11] D. Min, S. Choi, J. Lu, B. Ham, K. Sohn and M. N. Do, "Fast global image smoothing based on weighted least squares", *IEEE Trans. Image Process.*, 23(12), 5638-5653, 2014.
- [12] X. Cheng, M. Zeng and X. Liu, "Feature-preserving filtering with L_0 gradient minimization", *Computer & Graphics*, 38, 150-157, 2014.
- [13] R. M. H. Nguyen and M. S. Brown, "Fast and effective L_0 gradient minimization by region fusion", *Proc. IEEE Int. Conf. Comput. Vis.*, 208-216, 2015.
- [14] W. Huang, W. Bi, G. Gao, Y. Zhang and Z. Zhu, "Image smoothing via a scale-aware filter and L_0 norm", *Image Processing IET*, 12(9), 1511-1518, 2018.
- [15] P. Kniefacz and W. Kropatsch, "Smooth and iteratively restore: a simple and fast edge-preserving smoothing model", *Proceedings of the 39th annual workshop of the Austrian association for pattern recognition (OAGM2015)*, 1-9 eprint arXiv:1505.06702.
- [16] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion", *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 12(7), 629-639, 1990.
- [17] X. Zhang, Y. Yoshida and Ning Xie, "Anisotropic bandpass trilateral filter for creating artistic painting from real image", *2011 IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications*, DOI:10.1109/TrustCom.2011.200, 2011.
- [18] K. Inoue and K. Urahama, "Anisotropic bilateral filters for edge-preserving strip enhancement", *ITE*, 58(1), 115-120, 2004.

[19] J. Werckert, "Coherence-enhancing diffusion of color image", *Image and Vision Computing*, 17(3), 201-212, 1999.

[20] J. Werckert, "Coherence-enhancing shock filters", *Pattern Recognition, Lect. Notes in Comput. Sc.*, 2781, 1-8, 2003.

Xiaohua Zhang (Member) received his Dr. Eng. degree in information science and engineering from the Tokyo Institute of Technology, Japan, in 2000. After working at NHK Engineering System Inc. from 2000 to 2003, he joined the faculty of the Hiroshima Institute of Technology as a professor. His research interests include computer graphics, image processing, computer vision, pattern recognition, and machine learning. He is a member of the IEEE, the IEICE, the ITE, the IIEEJ, and the IIAE.

Yuelan Xin (Non-member) received a B.S. degree in physics from Shaanxi Normal University, China, in 1997, an M.S. degree in computer science and engineering from Qinghai Normal University, China, in 2009. She is currently an associate professor at Qinghai Normal University. Her research interests include image processing and pattern recognition.

Ning Xie (Non-member) received his M. E. from the Department of Computer Science, Tokyo Institute of Technology, Tokyo, Japan, in 2009, and a Dr. Eng. degree from the same department in 2012. From 2012 to 2014, he was a postdoctoral fellow in the same institute. In 2014, he was appointed as assistant professor at Tongji University. In 2016, he moved to the University of Electronic Science and Technology of China as an associate professor. His research interests include machine learning, computer graphics, and computer vision.