

Article

An Edge-Preserving Guided Filtering Algorithm Based on Edge Tangent Flow and Side Window

Tingting Liu  and Peng Cui 

School of Computer Science and Technology, Harbin University of Science and Technology, Harbin 150080, China; 2320400063@stu.hrbust.edu.cn

* Correspondence: cuipeng@hrbust.edu.cn; Tel.: +86-18845180982

Abstract

To address the limited edge-preserving capability of traditional guided filtering caused by fixing the filtering window center at the target pixel, an edge-preserving guided filtering algorithm based on edge tangent flow and side window (EWGF) is proposed. Rather than relying on local intensity statistics, the proposed method exploits local geometric structures through the integration of edge-aware weighting, tangent direction estimation, and adaptive side window filtering. Firstly, an edge-aware weighting strategy is introduced to construct a weighted gradient representation, enabling characterization of edge intensity and local structural features. Secondly, based on the weighted gradient field, a structure tensor constrained by local geometric information is established, and edge tangent flow is estimated through eigenvalue decomposition to capture local edge orientations. Furthermore, a tangent-guided one-dimensional side window guided filtering mechanism is developed, in which the filtering direction is adaptively aligned with local edge structures to suppress noise and textures while preserving edge sharpness and structural continuity. Finally, comparative experiments are conducted on the BSD500, Set5, and Set14 datasets, with PSNR and SSIM employed as evaluation metrics. Experimental results demonstrate that, compared with the traditional guided filtering algorithm, the proposed method improves PSNR by an average of 5.69 dB and SSIM by an average of 0.11, validating its performance in structure preservation and noise suppression.

Keywords: edge preserving; guided filtering; edge tangent flow; side window filtering; structure tensor



Academic Editor: João M. F. Rodrigues

Received: 17 May 2026

Revised: 29 June 2026

Accepted: 9 July 2026

Published: 14 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

In the field of image processing, edge-preserving filters are employed in image smoothing and detail enhancement tasks. The principal aim is to remove texture information from images while maintaining the stability of structural edges, thereby improving visual image quality and the performance of subsequent processing tasks. In recent years, various methods, including bilateral texture filtering (BTF) [1], side window filtering (SWF) [2], rolling guidance filtering (RGF) [3], and relative total variation (RTV)-based filtering [4], have been proposed and applied to image denoising, HDR image tone mapping, image enhancement, and texture smoothing. Achieving an effective balance between texture suppression and structure preservation has become an important research direction in the study of edge-preserving filtering.

Guided image filtering (GF) [5] is a typical local edge-preserving filtering method that achieves efficient filtering and edge preservation by establishing a local linear model

between the input image and the guidance image. However, since GF employs fixed windows for local modeling, cross-edge information propagation is likely to occur in edge regions, resulting in edge blurring and halo artifacts. To improve the edge-preserving performance of traditional GF, side window guided filtering (SWGf) [6] introduces the side window strategy into the guided filtering framework. By constructing eight directional side window structures, SWGF reduces the structural diffusion caused by cross-edge filtering. To further enhance the structural preservation capability of guided filtering, researchers have improved GF from the perspectives of edge-aware weighting and gradient constraints. Weighted guided image filtering [7] enhances structural preservation in edge regions by introducing an edge-aware weighting mechanism. Gradient domain guided filtering [8] achieves separation of edges and detail information through multi-scale gradient constraints; however, its ability to suppress textures within structural neighborhoods remains limited. To address this issue, Reference [9] combines weighting mechanisms with gradient information to enhance the discrimination capability between flat regions and edge regions. Nevertheless, this method relies on gradient magnitudes for region discrimination, making it difficult to distinguish structural information from texture information in low-contrast regions, thereby affecting the stability of the filtering performance.

In addition to local guided filtering frameworks, some studies have further enhanced structure preservation by introducing structural constraints and optimization models. Reference [10] embeds bilateral filtering into a least-squares optimization framework; however, it tends to cause excessive texture smoothing. Reference [11] performs Poisson noise image smoothing through image segmentation, but it is prone to color rendering distortion. Reference [12] proposes a texture filtering method based on edge guidance and adaptive kernel scales, which can remove texture information, although its structure-preserving capability remains insufficient. Reference [13] improves the edge-preserving performance of guided filtering by introducing explicit first-order edge constraints and residual constraints. These studies indicate that incorporating structural priors and edge constraints can improve filtering performance; however, achieving a balance between complex texture regions and weak edge regions remains challenging. In addition, guided filtering has also been studied in the field of image enhancement. Reference [14] proposes an adaptive dynamic guided image filtering model to address the limitation that the smoothing behavior of a given operator is fixed, and this model preserves edges and extracts structures. The adaptive finite difference method [15], used for numerically solving total variation type problems in image processing tasks, produces results that depend on its convergence criteria and grid. The Gaussian high-pass guided filtering prior model [16] lacks sensitivity to weak edge regions, which may lead to the loss of low-contrast details. Although non-uniform weighted guided filtering [17] can achieve multi-scale detail enhancement, its edge-preserving capability remains limited. Reference [18] combines illumination information with region-adaptive filtering and achieves favorable denoising performance under complex illumination conditions, but its ability to preserve structural continuity still requires improvement. Furthermore, some studies have investigated isotropic filtering and optimization-based smoothing methods. Traditional median filtering [19] lacks the ability to adaptively model local structural directions. The weighted least squares filtering method based on deep unsupervised learning [20] tends to blur structural details, while the quasi weighted least squares method [21] struggles to achieve a balance between texture removal and structure preservation. The work in [22] presents an edge-preserving image filtering method based on a hyperbolic tangent penalty function, where a regularization term is incorporated into the optimization model to generate the filtered output. Although this strategy improves edge preservation, its filtering process is driven by isotropic regular-

ization and does not exploit local geometric orientation information, which may limit its ability to preserve structural continuity in complex edge regions.

Overall, although existing methods have made certain progress in edge perception, gradient constraints, and structure preservation, most of them still rely on fixed-direction or isotropic local modeling strategies. Such approaches are prone to gradient distortion and structural diffusion in complex edge regions, resulting in insufficient preservation of sharp edges. To address these issues, this paper proposes an edge-preserving guided filtering algorithm based on the edge tangent flow and side window. The proposed method leverages edge tangent flow to construct edge-preserving windows that are consistent with the local structural orientation, and performs filtering along the structural tangent direction. This design optimizes the information propagation process in edge regions and suppresses cross-edge diffusion. Consequently, while suppressing noise and textures, the proposed method further enhances the preservation of sharp edges and fine structural details in images.

The main contributions of this paper are summarized as follows: (1) A novel edge-aware weighted geometric modeling strategy is proposed to enhance the extraction of local structural information and improve the robustness of tangent direction estimation. (2) A new edge tangent flow-guided side window filtering framework is developed, in which the filtering direction is adaptively aligned with local structural orientations, reducing cross-edge information diffusion. (3) A tangent-constrained one-dimensional side window guided filtering mechanism is introduced to suppress noise and textures while preserving edge sharpness and structural continuity. (4) Experiments on BSD500, Set5, and Set14 datasets demonstrate the superiority of the proposed EWGF method in terms of edge preservation, structural fidelity, and quantitative performance compared with existing guided filtering variants.

The remainder of this paper is organized as follows. Section 2 reviews the fundamentals of guided image filtering and side window filtering. Section 3 presents the proposed edge-preserving guided filtering algorithm based on edge tangent flow and side window (EWGF), including the edge tangent flow-guided side window filtering framework, the EWGF sampling strategy, the linear coefficient optimization strategy, and the overall implementation procedure. Section 4 describes the experimental setup and provides comprehensive evaluations, including parameter sensitivity analysis, quantitative and visual comparisons, edge-preserving performance analysis, and computational complexity analysis. Finally, Section 5 concludes the paper and discusses future research directions.

2. Methods

2.1. Guide Image Filtering

Guided image filtering [5] is an edge-preserving filter that employs a local linear model to connect the guide image with the output image. It is assumed that a linear relationship exists between the guidance image and the output image within a local window ω_k centered at pixel k , as illustrated in Equation (1).

$$F_i = a_k G_i + b_k, \forall i \in \omega_k, \quad (1)$$

where F_i and G_i denote the grayscale intensities of the i -th pixel in the output image and the guidance image, respectively. The parameters a_k and b_k are linear coefficients defined within the local window, as specified by Equation (3). $E(a_k, b_k)$ represents the cost

function, signifying the squared disparity between the output image and the input image, as illustrated in Equation (2).

$$E(a_k, b_k) = \sum_{i \in \omega_k} [(a_k G_i + b_k - I_i)^2 + \zeta a_k^2], \quad (2)$$

where ζ denotes the regularization parameter, which dictates the extent of blurring in the image; I_i represents the grayscale value of the i -th pixel in the input image. The coefficients a_k and b_k are derived from linear regression, as illustrated in Equation (3).

$$\begin{cases} a_k = \frac{\frac{1}{m} \sum_{i \in \omega_k} G_i I_i - \mu_k \bar{I}_k}{\sigma_k^2 + \zeta} \\ b_k = \bar{I}_k - a_k \mu_k \end{cases}, \quad (3)$$

where μ_k and σ_k^2 represent the mean and variance of the guidance image within the local window ω_k , $\bar{I}_k$ denotes the mean of the input image within the same local window, and m indicates the number of pixels in the local window.

2.2. Side Window Filtering

The side window filtering algorithm [2] improves edge preservation by aligning the edges or corners of the window with the pixels being processed. The eight outputs, denoted as F_n , are produced by applying the filtering kernel Ψ to each window, as shown in Equation (4).

$$F_n = \Psi(I_i, \theta, \rho, r), \quad (4)$$

where θ denotes the angle between the window and the horizontal line, r signifies the window radius, and ρ belongs to the set $\{0, r\}$. By modifying θ and the location of the target pixel i , the window orientation can be adjusted by aligning one of its edges with i .

To maintain edge structures, the distance between input and output is minimized at edge locations. The filter output is expected to match or approximate the edge input. Therefore, the output of the side window with the minimum distance L_2 to the input intensity is selected as the final output, as illustrated in Equation (5).

$$F_{p_{i,j}} = \operatorname{argmin}_{n \in \Omega} \|I_{p_{i,j}} - F_n\|_2^2, \quad (5)$$

where $F_{p_{i,j}}$ represents the side window filtering output of the target pixel; Ω represents the set of side windows in eight directions $\Omega = \{L, R, U, D, NW, NE, SW, SE\}$.

The eight directional subwindows utilized in side window filtering are illustrated in Figure 1, which include left (L), right (R), up (U), down (D), northwest (NW), northeast (NE), southwest (SW), and southeast (SE).

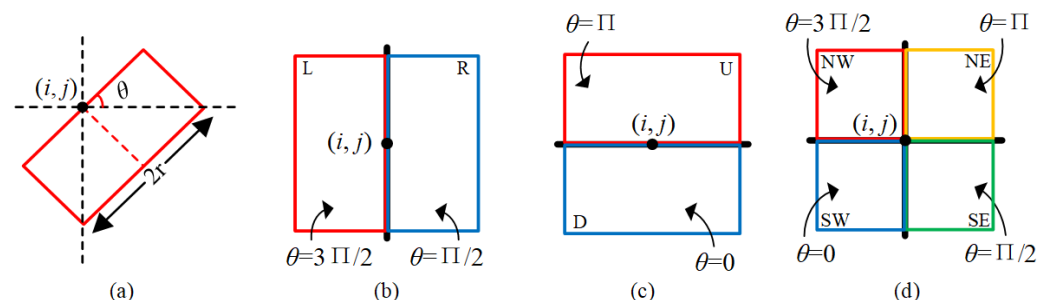


Figure 1. Eight-directional side window. (a) rotation angles; (b) L and R windows; (c) U and D windows; (d) NW, NE, SW, and SE windows.

The L , R , U , and D subwindows are designed to be half the size of the full window and are positioned to the left, right, up and down of the central pixel, respectively. This positioning is important for the preservation of sharp edges. The NW , NE , SW , and SE subwindows are sized to about one-quarter of the full window and are essential for maintaining the integrity of the corners.

3. Edge-Preserving Guided Filtering Algorithm Based on Edge Tangent Flow and Side Window

Our research proposes an edge-preserving guided filtering algorithm based on edge tangent flow and side window (EWGF) to address the issues of gradient distortion and insufficient edge preservation found in traditional guided filtering algorithms near edge regions. This chapter elaborates the implementation process of the EWGF algorithm: first, an adaptive weight calculation method utilizing edge tangent flow fields is introduced; second, a new side window sampling scheme is designed and an optimization process is established to determine linear coefficients a and b is established; finally, the overall implementation framework of the EWGF algorithm and its core methodologies are described in detail.

3.1. Edge Tangent Flow-Guided Side Window Filtering

In traditional side window filtering, edge information is represented by a set of predefined directional side window structures, while the structural orientation and local geometric characteristics of pixels are not considered. As a result, the preservation of structural details is limited, making it difficult to balance filtering performance and structure preservation. As shown in Figure 2, a scattered distribution of pixels is observed in the filtering result, indicating limited structural consistency. Filtering operations are performed along multiple directions within a two-dimensional window, which increases the computational cost.

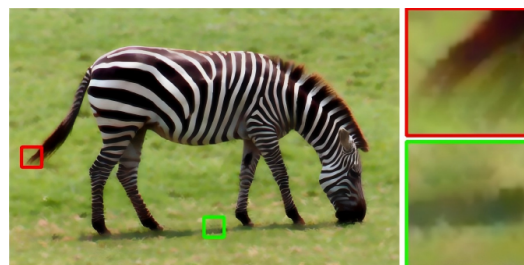


Figure 2. The traditional side window guided filtering result.

To enhance the adaptive representation of the pixel orientation information and neighborhood gradient characteristics in side window filtering, edge tangent flow (ETF) is employed as a structure-guided mechanism. Local structural information is represented along the tangent direction, while the directional distribution within a local neighborhood is characterized by the structure tensor. Local structural directions are estimated from the correspondence between the eigenvectors of the structure tensor and the local structural orientations.

Compared with traditional side window filtering based on predefined directional windows, ETF captures edge orientations and constrains filtering along tangent directions consistent with image structures. As a result, cross-edge diffusion is suppressed, structural preservation is improved, and the balance between texture filtering and structure preservation is enhanced. The mathematical formulation is given in Equation (6).

$$\mathbf{H}_{p_{i,j}} = \begin{bmatrix} \sum_{q_{i,j} \in N(p_{i,j})} \langle \nabla_x \mathbf{I}_{q_{i,j}} \cdot \nabla_x \mathbf{I}_{q_{i,j}} \rangle & \sum_{q_{i,j} \in N(p_{i,j})} \langle \nabla_x \mathbf{I}_{q_{i,j}} \cdot \nabla_y \mathbf{I}_{q_{i,j}} \rangle \\ \sum_{q_{i,j} \in N(p_{i,j})} \langle \nabla_y \mathbf{I}_{q_{i,j}} \cdot \nabla_x \mathbf{I}_{q_{i,j}} \rangle & \sum_{q_{i,j} \in N(p_{i,j})} \langle \nabla_y \mathbf{I}_{q_{i,j}} \cdot \nabla_y \mathbf{I}_{q_{i,j}} \rangle \end{bmatrix}, \quad (6)$$

where $N(p_{i,j})$ denotes the neighborhood centered at pixel $p_{i,j}$; $\nabla_x(\cdot)$ and $\nabla_y(\cdot)$ represent the first-order partial derivatives calculated in the horizontal and vertical directions, respectively; and $\langle \cdot \rangle$ indicates the scalar product. $\mathbf{H}_p$ is a symmetric, semi-positive definite matrix with two eigenvalues, $\lambda_{1,p} \geq \lambda_{2,p}$, and the corresponding eigenvectors are represented as ζ_p and τ_p . Eigenvalue decomposition is performed on the structure tensor, as shown in Equation (7).

$$\mathbf{H}_p = \lambda_{1,p} \zeta_p \zeta_p^T + \lambda_{2,p} \tau_p \tau_p^T. \quad (7)$$

The variable ζ_p represents the eigenvector in the perpendicular (tangential) direction, while τ_p denotes the eigenvector in the horizontal direction.

In traditional guided filtering, the filtering process is not anisotropic and does not account for structural orientation. The weights are determined by spatial distance, which may result in blurred edges. By contrast, the structure tensor enables anisotropic filtering, and filtering is performed along the tangent directions ζ_p , as shown in Equation (8).

$$\mathbf{I}_{p_{x,y}}^t = \sum_{p_{i,j}} w_{\zeta_p}(i, j) \cdot \mathbf{I}(x + i, y + j), \quad (8)$$

where w_{ζ_p} represents the weight in the tangential direction ζ_p , where $w_{\zeta_p} = \exp\left(-\frac{i^2 + j^2}{2\sigma^2}\right)$. Over-smoothing in the normal direction (which is perpendicular to the edge orientation) is prevented by filtering along the tangential direction, with the sharpness of edge contours and the integrity of image details being preserved.

Although cross-edge diffusion is reduced by the directional constraints provided by ETF, directional consistency does not guarantee structural consistency of the sampled pixels. In regions containing corners, texture transitions, and complex structures, pixels along the same tangent sampling line may belong to different local structures. The use of a fixed sampling window may therefore introduce structurally heterogeneous samples, reducing the preservation of structural details. To address this limitation, a side window sampling mechanism is integrated into the ETF-guided framework. Multiple candidate sampling windows are constructed along the tangent direction estimated by ETF, and the filtering response is selected from these candidates. The directional guidance provided by the ETF is combined with the local structural adaptability of the side-window mechanism, improving robustness in complex structural regions and preserving edges and structural details.

3.2. The EWGF Sampling Strategy

The sampling strategy of the EWGF algorithm consists of two forms: neighborhood point sampling and side window sampling. Figure 3 illustrates the adaptive neighborhood point sampling process of the EWGF algorithm along the tangent direction. Specifically, the red solid point O denotes the current filtering center (target pixel), and the black arrow represents the local edge tangent flow direction. The set $\{P1, P2, P3, P4\}$ denotes a neighborhood with a radius of 2 centered at O . The blue, green, orange, and purple arrows indicate the local directional fields corresponding to the sampling points within the neighborhood. By analyzing the directional information within the neighborhood, the proposed algorithm adaptively adjusts the sampling strategy according to local image structures. Consequently, the sampling process becomes more consistent with the underlying structural characteristics of the image, thereby suppressing textures and noise while better preserving structural edge continuity and directional consistency.

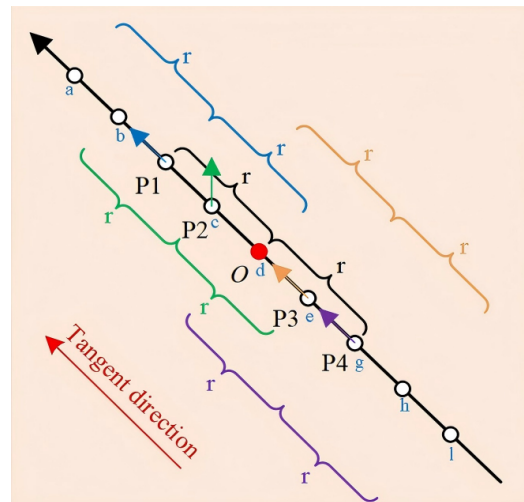


Figure 3. The neighborhood point sampling strategy of the EWGF.

As shown in Figure 3, the sampling point ($P2$), indicated by the green arrow, exhibits a directional deviation from the center point, where ($\theta' > 30^\circ$). In addition, $\{a, b, c, d\}$ denotes a neighborhood with a radius of 2 centered at pixel $P1$; $\{c, d, g, h\}$ denotes a neighborhood with a radius of 2 centered on pixel $P3$; and $\{d, e, h, l\}$ denotes a neighborhood with a radius of 2 centered on pixel $P4$. This representation illustrates the directional variations of neighboring pixels during the filtering process, as well as their directional relationships with the center point.

As illustrated in Figure 4, the neighborhood point sampling procedure of the EWGF algorithm is presented. The algorithm first traverses each pixel (i, j) in the image and treats it as the center point O . The spatial position (x, y) and directional information θ of the center point are recorded, while an initial neighborhood radius is specified to constrain the sampling region. Within this neighborhood, the spatial positions (x', y') and directional information θ' of neighboring pixels are extracted and subjected to multidimensional conditional screening, including directional similarity and distance threshold constraints. The pixels satisfying these conditions are grouped into the initial neighborhood set f' , and their coordinate and directional information are retained. Subsequently, the algorithm selects the neighborhood point in f' that is most similar to the target pixel and reconstructs a refined neighborhood set f'' centered at this point, thereby enhancing local structural consistency and sampling representativeness. Finally, the algorithm outputs the neighborhood point set f satisfying the prescribed conditions.

This sampling strategy captures local texture characteristics while preserving the global image structure. Through neighborhood pixel screening and adaptive sampling, the proposed method preserves edge structures while processing high-frequency details. This mechanism balances texture suppression and structural preservation, smoothing the image while reducing structural blurring and detail loss.

After neighborhood point sampling, the EWGF algorithm employs a multi-strategy sampling mechanism for side windows along the tangential direction, and its sampling strategy is illustrated in Figure 5. In the figure, red dots represent the center points, black dots indicate the neighboring points within a radius of 2, and gray boxes denote the regions sampled by the side window.

The sampling methodology includes full-window sampling, left half-window sampling, right half-window sampling, and middle half-window sampling. Multiple sub-windows are constructed, and sampling is performed incrementally along the tangent direction. The sampling procedure begins with the full window, followed by the left half-window, the right half-window, and the middle half-window. Filtering is performed

within each sub-window, and the final output is selected from the responses of multiple side windows. This multi-branch strategy improves filter adaptability and robustness in regions with complex structures.

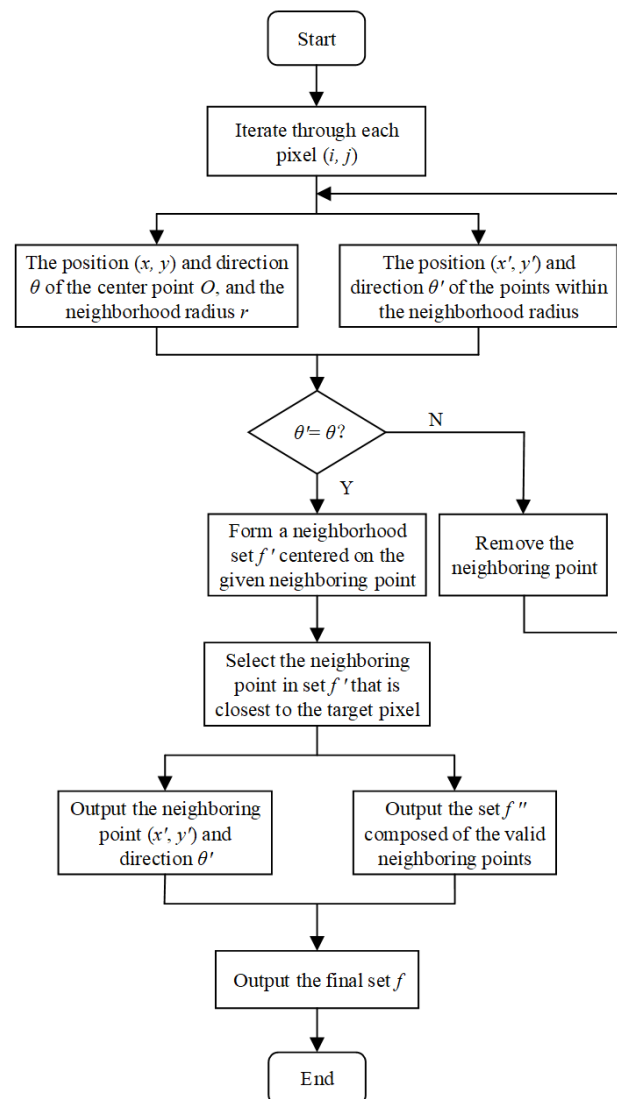


Figure 4. The flowchart shows the neighborhood point sampling of the EWGF.

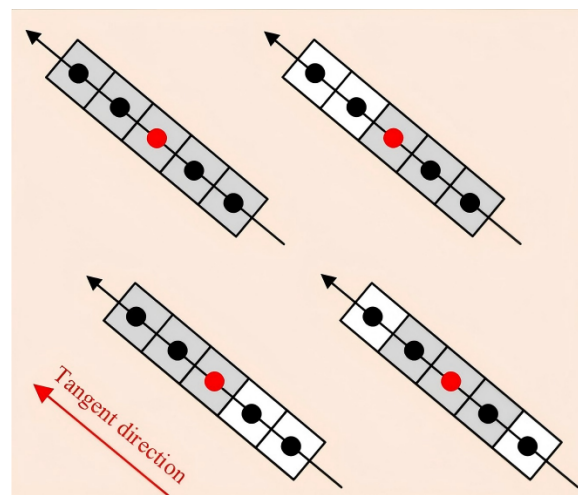


Figure 5. The side window sampling strategy of the EWGF.

3.3. EWGF Linear Coefficient Optimization Strategy

To enhance the responsiveness of image structure information during the filtering process, we introduce directional constraints over the linear coefficients used in traditional guided filtering. This approach aims to enhance the preservation of edge structures. The formulas for calculating the linear coefficients, a and b , are presented in Equation (9).

$$\begin{cases} a_{i,j} = \frac{\sigma_{i,j}^2}{\sigma_{i,j}^2 + \zeta} \\ b_{i,j} = \mu_{i,j}(1 - a_{i,j}) \end{cases}, \quad (9)$$

where ζ represents the regularization parameter to prevent division by zero; $\mu_{i,j}$ and $\sigma_{i,j}^2$ denote the mean and variance along the tangent direction, respectively, as illustrated in Equations (10) and (11).

$$\mu_{i,j} = \frac{1}{2r+1} \sum_{I \in \omega_{i,j}} I, \quad (10)$$

$$\sigma_{i,j}^2 = \frac{1}{2r+1} \sum_{I \in \omega_{i,j}} (I - \mu_{i,j})^2, \quad (11)$$

where I represents the input image and $\omega_{i,j}$ denotes the collection of pixels aligned to the local window in the tangential direction.

In the EWGF algorithm, the optimization strategy for the direction-aware linear coefficients a and b is illustrated in Figure 6.

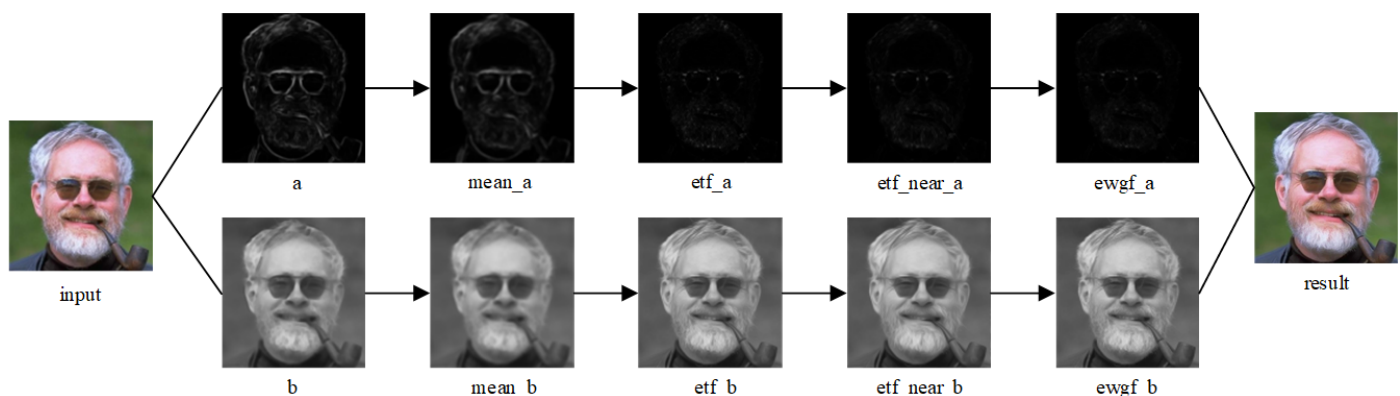


Figure 6. Optimization strategy for the linear coefficients.

The coefficients a and b are calculated using traditional guided filtering, with their average over the entire image serving as the fundamental estimate. Following this, an adaptive sampling is performed in the tangential flow direction. Within a direction-sensitive one-dimensional window, a and b are modified through the integration of local variance normalization and mean compensation techniques. A multi-tiered directed propagation system is employed to aggregate filter coefficients within the vicinity, enhancing structural consistency. The final selection of the optimal coefficients is achieved by minimizing the reconstruction error across various candidate sets, leading to anisotropic modeling. This algorithm improves edge preservation while maintaining the efficiency of traditional guided filtering.

3.4. EWGF Algorithm Implementation

The EWGF algorithm improves upon the traditional guided filtering method by utilizing geometrically adapted windows. This algorithm replaces the traditional rectangular region with linear side windows that are aligned with the major curvature direction, establishing direction-aware local geometric constraints. The local structure of an image,

influenced by differential geometry, can be understood as a surface that unfolds in the direction of the edge tangent flow. Consequently, the EWGF algorithm performs linear modeling and filtering along the tangential direction. The consistency weight of the directional field between pixel (i, j) and the central pixel (x, y) in the image is defined to measure their directional similarity, as shown in Equation (12).

$$w_{x,y} = \exp\left(-\frac{\|X_{i,j} - X_{x,y}\|^2}{\sigma_s^2}\right) \cdot \exp\left(-\frac{\|\theta_{i,j} - \theta_{x,y}\|^2}{\sigma_\theta^2}\right) \cdot Z(\|X_{i,j} - X_{x,y}\| \leq r), \quad (12)$$

where $X_{x,y}$ represents the position and magnitude of the central point (x, y) , while $X_{i,j}$ indicates the current position of the adjacent point (i, j) . $\theta_{x,y}$ denotes the tangent flow direction at the central point (x, y) , and $\theta_{i,j}$ refers to the local direction at the adjacent point (i, j) . In addition, σ_s is the spatial adjustment parameter, σ_θ is the directional difference adjustment parameter, and $Z(\cdot)$ functions as an indicator to delineate the neighborhood range.

For the center point (x, y) , the set of valid sampling points is defined in Equations (13) and (14).

$$\Phi_{x,y}^l = \{(i, j) \mid \|X_{i,j} - X_{x,y}\| \leq r\}, \quad (13)$$

$$\Phi_{x,y}^s = \left\{ (i, j) \in \Phi_{x,y}^l \mid \frac{(X_{i,j} - X_{x,y}) \cdot \theta_{x,y}}{\|X_{i,j} - X_{x,y}\|} \geq \delta_s \right\}, \quad (14)$$

where $\Phi_{x,y}^l$ denotes the collection of sampled neighboring points, and the weight $w_{x,y}$ is calculated for each of these points. $\Phi_{x,y}^s$ represents the set of sampled points within the edge window, where $s \in \{M_{full}, M_{left}, M_{right}, M_{mid}\}$. Additionally, δ_s indicates the threshold for directional control.

For each sub-window, the linear coefficients a and b are computed as shown in Equations (15) and (16).

$$a_{x,y}^s = \frac{\sum_{(i,j) \in \Phi_{x,y}^s} w_{x,y} (G_{i,j} - G_{x,y}) \{I_{i,j} - \bar{I}_{x,y}\}}{\sum_{(i,j) \in \Phi_{x,y}^s} w_{x,y} (G_{i,j} - G_{x,y})^2 + \zeta}, \quad (15)$$

$$b_{x,y}^s = \bar{I}_{x,y} - a_{x,y}^s G_{x,y}, \quad (16)$$

where $a_{x,y}^s$ and $b_{x,y}^s$ denote the linear coefficients for each sub-window. $G_{x,y}$ represents the local weighted mean of the guiding image G , while $\bar{I}_{x,y}$ reflects the local weighted mean of the input image I .

A set of one-dimensional side windows in the tangential direction is constructed, as illustrated in Equation (17).

$$T = \left\{ (a_{x,y}^s, b_{x,y}^s) \mid s \in \{M_{full}, M_{left}, M_{right}, M_{mid}\} \right\}. \quad (17)$$

The ultimate filtering result in the tangent direction is obtained by minimizing the distance between the filtered output and the input image, as shown in Equation (18).

$$F_p^t = \underset{n \in T}{\operatorname{argmin}} \left\| \frac{\sum_{q \in n} a_{x,y}^s}{|n|} G_{q_{i,j}} + \frac{\sum_{q \in n} b_{x,y}^s}{|n|} - I_p \right\|_2^2, \quad (18)$$

where F_p^t represents the filtered result along the tangential direction, while n denotes a candidate window from the tangential side window set T . The algorithm compares the filtering effects of all candidate windows and selects the optimal n as the final output.

The comprehensive implementation process of the EWGF algorithm is described in Algorithm 1 as follows.

Algorithm 1. Edge-preserving guided filtering algorithm based on edge tangent flow and side window (G, I, ζ, r, γ)

Input: guidance image G , input image I , regularization parameter ζ , filtering window radius r , edge-aware weights γ .
 Define the set of spatial window directions $\Omega = \{L, R, U, D, NW, NE, SW, SE\}$;
 Compute the gradient field $\nabla G \leftarrow \text{Sobel}(G)$;
 Construct the ETF direction field $T \leftarrow \text{ETF}(\nabla G)$;
 Compute edge-aware weights γ ;
 for iterations < 5
 for each row i in image G
 for each column j in image G
 Sample neighborhood pixels along the ETF direction $\theta \leftarrow T(I)$;
 Sampling window $\Omega \leftarrow$ pixels along the θ direction within the interval $[I - r, I + r]$;
 Compute the values of a and b for the local window along the ETF direction;
 Perform a weighted average of a and b along the ETF direction;
 Compute local statistics: $\mu_{i,j}, \sigma_{i,j}^2$;
 Compute the linear coefficients: $a_{i,j} = \frac{\sigma_{i,j}^2}{\sigma_{i,j}^2 + \zeta}, b_{i,j} = \mu_{i,j}(1 - a_{i,j})$;
 end
 end
 Select the optimal values of a and b from all windows;
 Adaptively adjust the smoothing intensity based on the edge-aware weight γ ;
end
Update the guidance image using the optimal linear coefficients $a_{i,j}$ and $b_{i,j}$;
Output: filtered image F_p^t .

4. Results and Discussion

This section evaluates the performance of the proposed EWGF algorithm through quantitative and qualitative analyses. Comparisons are conducted between EWGF and representative image filtering methods on multiple datasets. Experimental data and visual results are analyzed to assess the filtering performance of the proposed method.

4.1. Datasets and Metrics

Experiments were conducted on three public benchmark datasets: BSD500, Set5, and Set14. The peak signal-to-noise ratio (PSNR) and the structural similarity index measure (SSIM) were adopted as quantitative evaluation metrics. All images were converted from the RGB color space to the YCbCr color space, and filtering was applied only to the luminance (Y) channel. This processing strategy follows the higher sensitivity of the human visual system to luminance information while preserving color fidelity.

4.1.1. Datasets

Figure 7 shows representative images from the BSD500, Set5, and Set14 datasets.

BSD500 consists of 500 natural scene images, covering a wide variety of content including humans, animals, buildings, plants, and indoor and outdoor scenes. The image resolutions are not uniform, with common dimensions around 481×321 pixels. Owing to its large sample size and high diversity, this dataset offers the strongest statistical representativeness and was therefore adopted as the primary benchmark for quantitative evaluation.

Set5 comprises only five natural images, with content including a baby, a butterfly, and human portraits. The resolutions are relatively low and vary across images, typically around 512×512 pixels or smaller. Given the limited sample size, quantitative results derived from Set5 are susceptible to fluctuations caused by individual images; consequently, this dataset is more suitable for visual comparison than for numerical analysis.

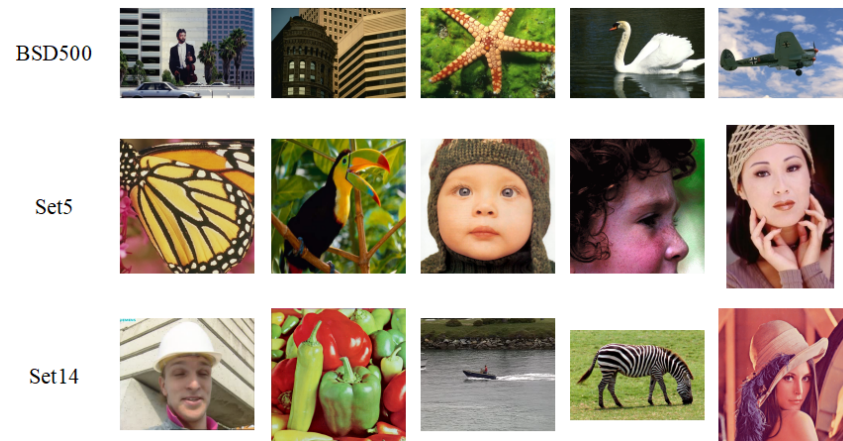


Figure 7. Representative images from the BSD500, Set5, and Set14 datasets.

Set14 contains 14 images, covering both natural scenes and man-made objects, such as humans, animals, buildings, bridges, and still-life subjects. The image resolutions also vary, roughly ranging from 200×200 to 600×500 pixels. Similar to Set5, the small sample size restricts the statistical reliability of quantitative comparisons, although the dataset remains valuable for qualitative visual assessment.

4.1.2. Evaluation Metrics

PSNR measures image reconstruction quality by evaluating the ratio of the maximum possible signal power to the noise power. Higher PSNR values indicate lower reconstruction error. The mathematical formulation of PSNR is given in Equation (19).

$$PSNR = 10 \log_{10} \left(\frac{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} I(i, j)_{max}^2}{MSE} \right). \quad (19)$$

The MSE term in Equation (19) is defined in Equation (20).

$$MSE = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \left(I(i, j) - I'(i, j) \right)^2, \quad (20)$$

where M and N denote the numbers of rows and columns of the image, respectively; $I(i, j)_{max}$ is the maximum pixel value of the input image; $I(i, j)$ represents the pixel value of the input image at location (i, j) ; and $I'(i, j)$ represents the pixel value of the output image at location (i, j) .

SSIM evaluates structural similarity by comparing the luminance, contrast, and structural information of two images within local windows. Higher SSIM values indicate greater structural similarity between the processed image and the reference image. The mathematical formulation of SSIM is given in Equation (21).

$$SSIM = \frac{\left(2\mu_x\mu_y + \left(K_1(2^b - 1) \right)^2 \right) \left(2\sigma_{xy} + \left(K_2(2^b - 1) \right)^2 \right)}{\left(\mu_x^2 + \mu_y^2 + \left(K_1(2^b - 1) \right)^2 \right) \left(\sigma_x^2 + \sigma_y^2 + \left(K_2(2^b - 1) \right)^2 \right)}, \quad (21)$$

where μ_x and μ_y denote the mean values of the input and output images, respectively; σ_x and σ_y denote the standard deviations of the input and output images, respectively; K_1 and K_2 are constants, whose empirical values are typically set to 0.01 and 0.03, respectively; and b denotes the bit depth of the image.

4.2. Analysis of Experimental Parameters

The experiments in this study were implemented on a computer platform with an Intel® Core™ i9-2.20 GHz CPU, 16 GB RAM, a 64-bit Windows 11 operating system, and MATLAB 2020a.

To evaluate the performance of the proposed algorithm, our study selected twelve representative edge-preserving filtering methods for comprehensive comparison, including: BTF [1], RGF [3], RTV [4], GF [5], SWGF [6], anisotropic and isotropic TV (AITV) [11], edge guidance filtering (EGF) [12], weighted least square filter via deep unsupervised learning (DWLS) [20], intensity and scale adjustable edge-preserving smoothing filter (ISES) [23], edge-aware filter based on adaptive patch variance weighted average (GVWA) [24], semantic-aware structure-preserving median morpho-filtering (SASTF) [25], and generalized framework for image smoothing (GFIS) [26].

To ensure fairness of the comparisons, the parameter configurations for the aforementioned comparison methods are summarized in Table 1. The parameter settings of GF and SWGF are excluded, as they are presented in the subsequent parameter comparison. All parameters were employed in accordance with the respective original publications or source code implementations.

Table 1. Parameter selection of the ten comparison methods.

Algorithm	Parameter 1	Parameter 2	Parameter 3	Parameter 4
BTF	$iter = 5$	$k = 5$		
RGF	$iter = 4$	$\sigma_s = 3$	$\sigma_r = 0.1$	
RTV	$iter = 4$	$\sigma = 3$	$\zeta = 0.02$	$\lambda = 0.01$
AITV	$iter = 300$	$k = 16$	$\lambda = 5$	
EGF	$i = 1$	$\lambda = 0.01$		
DWLS	$\lambda = 10$	$\alpha = 0.08$		
ISES	$\sigma_s = 5$	$p = 5$		
GVWA	$iter = 10$	$\sigma_s = 2$	$s = 0.25$	
SASTF	$iter = 3$	$w_{sk} = 5$		
GFIS	$iter = 10$	$\lambda = 0.5$		

In experiments, two parameters influencing performance are the radius parameter (r) and the regularization parameter (ζ). The radius parameter was set to values of 2, 4, and 8, while the regularization parameter was assigned values of 0.1², 0.2², and 0.4². In addition to the main comparison parameters, two additional weighting parameters are introduced in Equation (12) of the proposed EWGF algorithm: the spatial regulation parameter σ_s and the directional difference regulation parameter σ_θ . Their settings and determination procedures are described as follows.

Spatial regulation parameter (σ_s): This parameter controls the decay rate of the spatial Gaussian kernel and determines the contribution of pixels located at different distances from the center pixel to the filtering result. In this study, the spatial standard deviation σ_s is associated with the window radius r , and its formulation is given in Equation (22).

$$\sigma_s = \frac{r}{2}. \quad (22)$$

The rationale for this setting is that when a sampling point is located at the window boundary (i.e., at a distance r from the center pixel), the corresponding spatial weight decays to $e^{-4} \approx 1.83\%$. The support region is therefore concentrated around the center pixel, preserving edge structures and reducing the influence of distant pixels. This parameter setting was validated through a grid search on the validation set, where σ_s varied from 0.5 to 5.0 with a step size of 0.5. The results indicate that the adopted setting achieves the best performance.

Directional difference regulation parameter (σ_θ): This parameter controls the decay rate of the directional Gaussian kernel and determines the extent to which pixels with tangent-flow directions inconsistent with that of the center pixel are suppressed during filtering. σ_θ is defined by Equation (23).

$$\sigma_\theta = \frac{\pi}{6}. \quad (23)$$

The value of σ_θ was determined through a grid-search procedure conducted on a validation set comprising 100 images selected from the BSD500 dataset. With $r = 4$, $\zeta = 0.2^2$, and $\sigma_s = \frac{r}{2}$ fixed, σ_θ was varied from 0.1 to 1.0 radians in increments of 0.1. The average PSNR was adopted as the evaluation criterion. The experimental results showed that optimal performance was achieved when $\sigma_\theta = \frac{\pi}{6}$. Smaller values imposed strict directional constraints, resulting in reduced denoising, whereas larger values relaxed the directional constraints and led to edge blurring. The parameter was fixed in all subsequent experiments, whereas σ_s was determined by r . No parameter tuning was performed for individual images, ensuring the fairness of the experiments.

A comparative analysis of the filtering results achieved under various parameter configurations for grayscale and color images is presented in Figures 8 and 9. This analysis includes the use of GF, SWGF, and the proposed EWGF algorithm. In all experiments, the input image served as the guidance image. By adjusting the radius and regularization parameters, differences in edge preservation and texture filtering performance between the filters were evaluated.

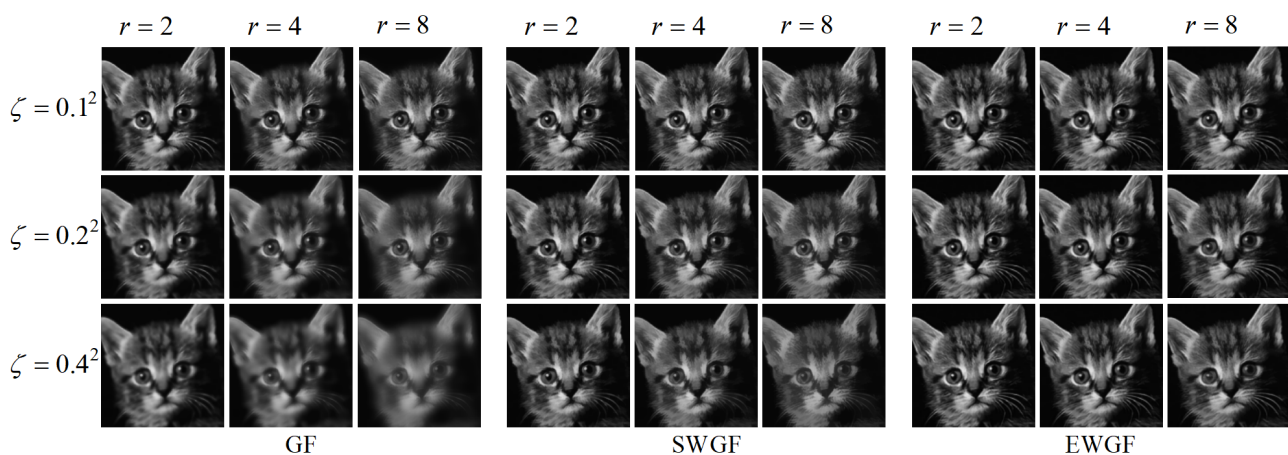


Figure 8. The filtering results of the grayscale image under different parameter settings.

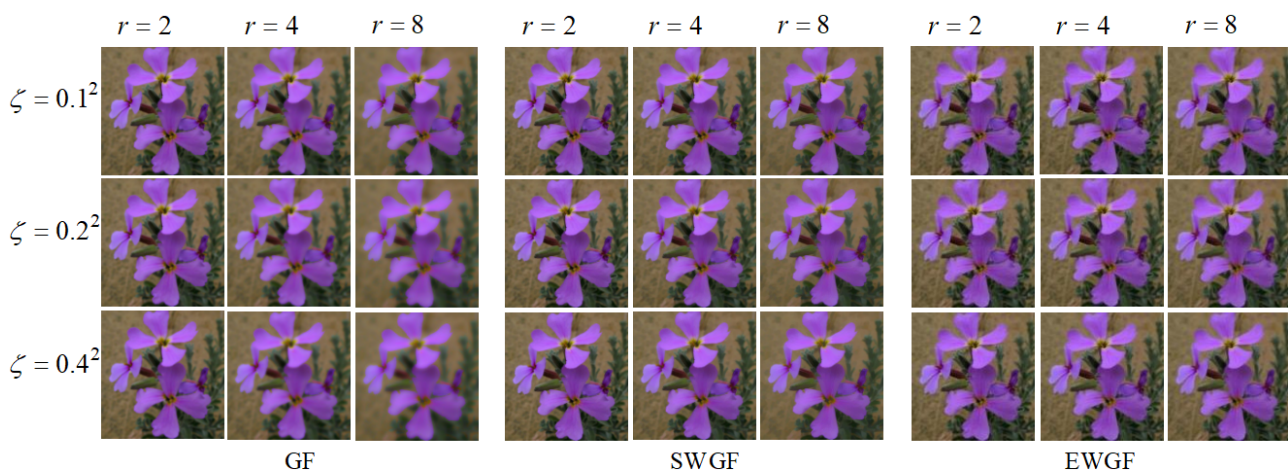


Figure 9. The filtering results of the color image under different parameter settings.

The filtering results of the grayscale and color images show that, as the parameters r and ζ increase, the outputs of all three filters exhibit different degrees of blurring. Among them, GF produces the strongest blurring effect. Under large radius and regularization parameter settings, image details are lost and the clarity of edge contours decreases. By introducing the side window mechanism, SWGF achieves better edge preservation than GF while maintaining the filtering effect. However, structural blurring still exists, and the recovery of fine details remains limited. In contrast, under the same parameter settings, the proposed EWGF algorithm suppresses structural blurring while maintaining the filtering effect. In edge regions, EWGF preserves image details and edge sharpness, and the structural information of the image is retained.

Figure 10 illustrates the variation curves of PSNR with respect to the filter radius r under different regularization parameters ζ . The curves are generated based on the average PSNR values obtained from all 500 images in the BSD500 dataset. It can be observed that, as either ζ or r increases, the PSNR generally exhibits a smooth downward trend without severe fluctuations or abnormal oscillations, indicating that the proposed algorithm possesses robust stability against parameter variations.

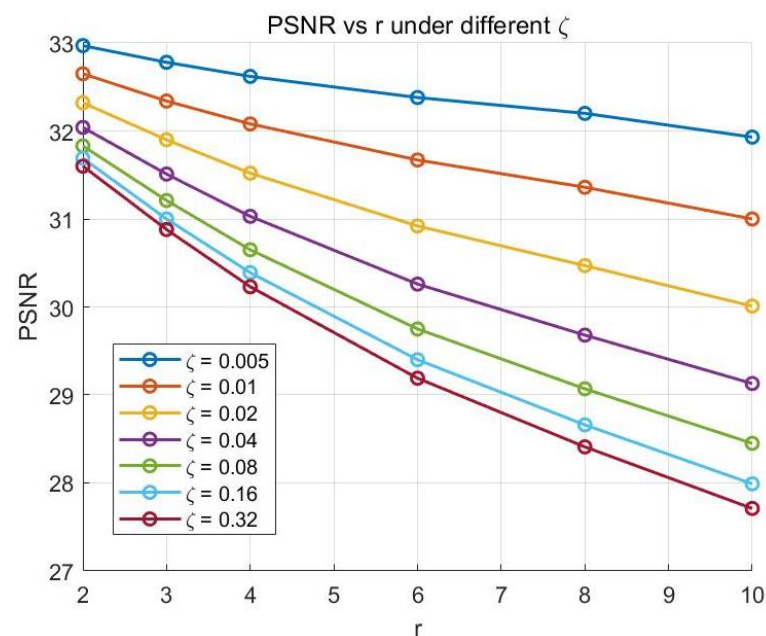


Figure 10. PSNR versus r for different values of ζ .

Under different parameter combinations, the curves maintain similar intervals and consistent descending tendencies, suggesting that the performance variation is predictable rather than sensitive to a specific parameter. Such consistency in trends and smoothness of variation further demonstrate the strong robustness of the proposed method. Even when the parameter values are large, the PSNR remains within an acceptable range, indicating that the algorithm can still maintain stable restoration performance over a wide parameter range.

Figure 11 presents the heat map of SSIM under different combinations of the filter radius r and regularization parameter ζ , which is used to analyze the parameter sensitivity and structure-preserving capability of the proposed algorithm. The results were obtained on the BSD500 dataset using all 500 images, and the reported SSIM values represent the average performance over the entire dataset.

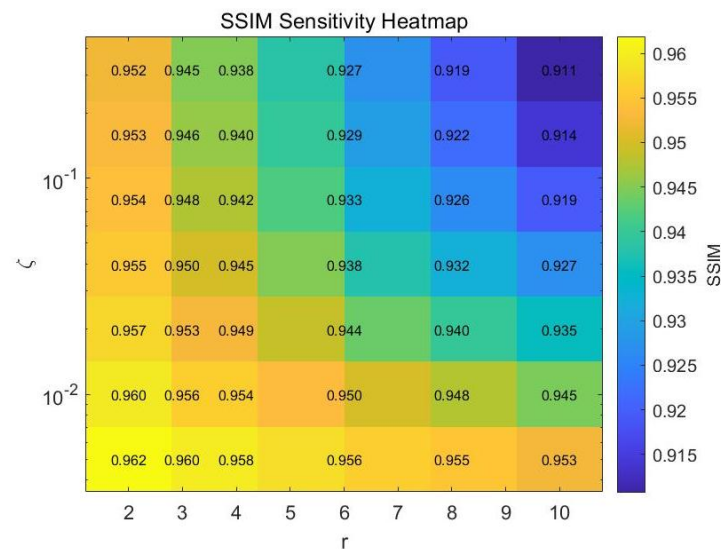


Figure 11. Heatmap of SSIM variations under different combinations of r and ζ .

When ζ is fixed, the SSIM gradually decreases as r increases, indicating that a larger filter radius weakens the ability to preserve structural details. Similarly, when r is fixed, the SSIM also decreases with increasing ζ , suggesting that strong regularization suppresses detail information and reduces structural similarity.

From the distribution of values, higher SSIM values are concentrated in the region with small ζ and r (the yellow-colored region), which is beneficial for preserving structural information. In contrast, the region with larger ζ and r (the purple-colored region) exhibits lower SSIM values, indicating that excessive smoothing leads to structural degradation. The variations in the heat map are continuous and smooth without abnormal fluctuations, and the SSIM remains at a high level over a wide parameter range, demonstrating that the proposed algorithm has robust stability against parameter variations.

4.3. Analysis of the Results

4.3.1. Quantitative Evaluation

Table 2 presents the quantitative comparison of the GF, SWGF, and EWGF algorithms on the BSD500 dataset, where all 500 images were used for evaluation. For each parameter configuration, the mean and standard deviation (Mean $\pm$ Std) of PSNR and SSIM are reported, where the mean reflects the average filtering performance over the entire dataset and the standard deviation indicates the performance variability across different images. The optimal result for each metric is highlighted in bold.

Table 2. Comparison of PSNR (Mean $\pm$ Std) and SSIM (Mean $\pm$ Std) for different filtering algorithms under varying parameter settings.

ζ	Algorithm	PSNR (Mean $\pm$ Std)/dB			SSIM (Mean $\pm$ Std)		
		$r = 2$	$r = 4$	$r = 8$	$r = 2$	$r = 4$	$r = 8$
$\zeta = 0.1^2$	GF	28.91 $\pm$ 2.27	27.94 $\pm$ 2.00	27.14 $\pm$ 1.69	0.9050 $\pm$ 0.0507	0.8935 $\pm$ 0.0547	0.8940 $\pm$ 0.0525
	SWGF	30.48 $\pm$ 2.42	29.13 $\pm$ 2.21	28.02 $\pm$ 1.89	0.9312 $\pm$ 0.0380	0.9115 $\pm$ 0.0472	0.9037 $\pm$ 0.0492
	EWGF	32.65 $\pm$ 2.41	32.08 $\pm$ 2.43	31.36 $\pm$ 2.34	0.9595 $\pm$ 0.0208	0.9536 $\pm$ 0.0246	0.9477 $\pm$ 0.0274
$\zeta = 0.2^2$	GF	26.37 $\pm$ 2.64	24.87 $\pm$ 2.39	23.55 $\pm$ 2.11	0.8579 $\pm$ 0.0768	0.8271 $\pm$ 0.0876	0.8136 $\pm$ 0.0889
	SWGF	28.51 $\pm$ 2.85	26.37 $\pm$ 2.61	24.60 $\pm$ 2.29	0.9054 $\pm$ 0.0551	0.8622 $\pm$ 0.0744	0.8345 $\pm$ 0.0824
	EWGF	32.04 $\pm$ 2.67	31.03 $\pm$ 2.79	29.68 $\pm$ 2.77	0.9553 $\pm$ 0.0244	0.9450 $\pm$ 0.0314	0.9323 $\pm$ 0.0378
$\zeta = 0.4^2$	GF	24.90 $\pm$ 2.95	23.15 $\pm$ 2.75	21.57 $\pm$ 2.54	0.8235 $\pm$ 0.0954	0.7764 $\pm$ 0.1126	0.7484 $\pm$ 0.1190
	SWGF	27.36 $\pm$ 3.16	24.77 $\pm$ 2.96	22.67 $\pm$ 2.68	0.8879 $\pm$ 0.0666	0.8244 $\pm$ 0.0950	0.7767 $\pm$ 0.1099
	EWGF	31.69 $\pm$ 2.84	30.39 $\pm$ 3.04	28.66 $\pm$ 3.08	0.9529 $\pm$ 0.0264	0.9396 $\pm$ 0.0356	0.9217 $\pm$ 0.0449

As shown in Table 2, the EWGF algorithm achieves the highest mean PSNR and SSIM under all tested parameter settings. For example, when $\zeta = 0.1^2$ and $r = 2$, EWGF achieves a PSNR of 32.65 dB and an SSIM of 0.9595, exceeding those obtained by GF and SWGF. Similar performance is observed for the remaining parameter combinations, indicating the effectiveness of the proposed method in preserving image structures and edges. When the radius parameter r is fixed, increasing the regularization parameter ζ from 0.1^2 to 0.4^2 leads to a decrease in the mean PSNR and SSIM for all three methods. Similarly, when ζ is fixed, increasing the filtering radius r from 2 to 8 also reduces the mean PSNR and SSIM. This performance degradation is attributed to stronger smoothing, which suppresses image noise while reducing fine structural details and edge information. In addition, the standard deviations of EWGF are comparable to or smaller than those of GF and SWGF under most parameter settings. These results indicate that EWGF not only provides higher average image quality, but also maintains more stable filtering performance across images with different contents. Therefore, the proposed method demonstrates robustness and edge-preserving capability over a range of parameter settings.

A performance comparison between the EWGF algorithm and ten other algorithms, which include both traditional filtering methods and the latest filtering methods, is presented in Table 3. The evaluation was conducted on the BSD500 dataset, where all 500 images were used for evaluation, and the performance was measured using PSNR and SSIM. The optimal result for each metric is highlighted in bold.

Table 3. Comparison of PSNR and SSIM evaluation metrics for 11 algorithms.

Algorithm	PSNR/dB	SSIM
BTF	23.21	0.7630
RGF	25.77	0.8316
RTV	23.98	0.7813
AITV	23.17	0.7534
EGF	23.66	0.7752
DWLS	23.29	0.7825
ISES	25.04	0.8394
GVWA	22.08	0.7549
SASTF	15.77	0.6212
GFIS	25.32	0.8313
EWGF	31.06	0.9453

The experimental results show that the proposed EWGF algorithm achieves higher PSNR and SSIM values than the other ten comparison algorithms. In addition, the algorithm maintains a balance between texture suppression and structural preservation. In terms of visual performance and quantitative evaluation metrics, the proposed method exhibits measurable performance characteristics.

4.3.2. Visual Comparison of the Results

To evaluate the visual performance of the EWGF algorithm, three sets of comparative experiments are conducted to assess its image filtering performance and edge-preserving capability. A representative image from each of the Set5, Set14, and BSD500 datasets is selected for testing. Figure 12 shows the visual comparison results obtained on a representative image from the Set5 dataset, demonstrating the performance differences between the proposed EWGF algorithm and traditional and recently developed edge-preserving filtering methods.

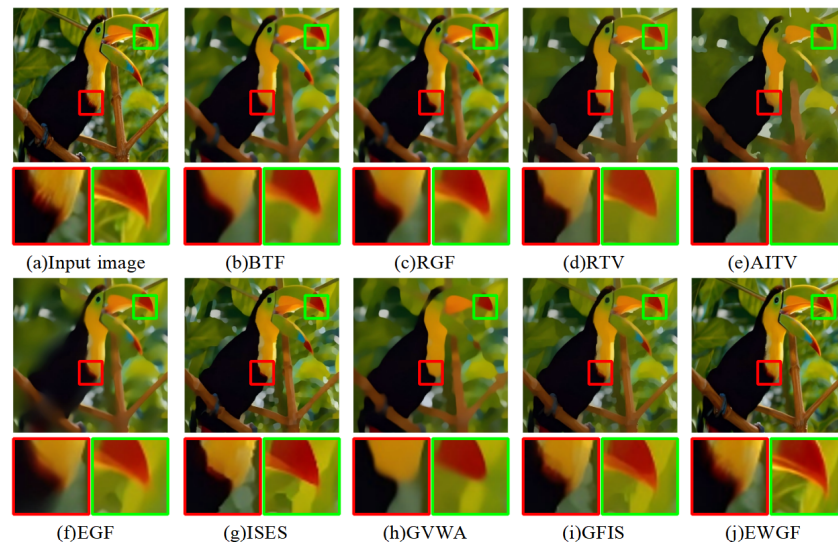


Figure 12. Comparison of filtering results from the Set5 dataset.

As shown in the figure, although BTF, RGF, and RTV remove a large amount of texture information, a considerable amount of structural information is inevitably lost during the filtering process. In the results produced by AITV, edge blurring is observed, and abnormal color rendering leads to noticeable color distortion. Obvious block artifacts are generated by ISES, accompanied by unsatisfactory preservation of edge structures. Although the overall filtering performance is improved by EGF and GFIS, the edge structures are still not adequately preserved. Similarly, GVWA loses a substantial amount of structural information during filtering. In contrast, the proposed EWGF algorithm preserves small-scale structures and sharp edges while removing texture information.

Figure 13 presents the filtering results for an image from the Set14 dataset, providing a visual comparison among different edge-preserving filtering methods.

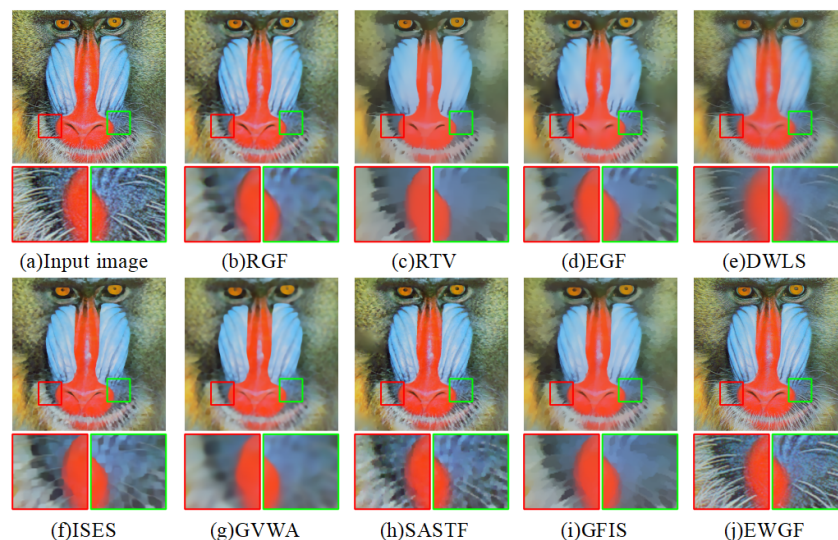


Figure 13. Comparison of filtering results from the Set14 dataset.

As indicated in the figure, the loss of structural information is suffered by RTV, GVWA, and GFIS during the filtering process, resulting in the complete blurring of fine hair details. Noticeable block effects are exhibited in the filtering results of ISES and SASTF, with small-scale structures failing to be preserved. Only slight improvements in overall filtering performance are shown by RGF and EGF, although most structural information remains obscured. Some progress is demonstrated by DWLS compared to the mentioned algorithms,

but preservation of small-scale structures remains inadequate. In contrast, the integrity of small-scale structures and edge clarity are maintained by the EWGF algorithm, while texture information is removed.

Figure 14 shows the visual comparison results for a representative image from the BSD500 dataset.

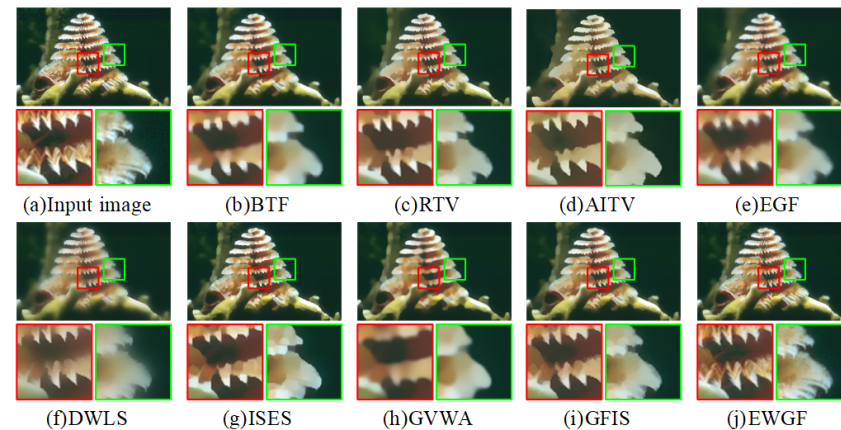


Figure 14. Comparison of filtering results from the BSD500 dataset.

As illustrated in the image, numerous sharp and complex structures are not only obscured but also distortion of the overall image information is caused by the filtering results of BTF, EGF, and GVWA. Although the preservation of small-scale structures has been improved to a certain extent by RTV and DWLS, the overall images still appear blurred. Edge blurring is induced, and color rendering anomalies presenting tonal deviation are shown by AITV. Considerable clumping is demonstrated by ISES and GFIS while texture is being filtered out. In contrast, the texture information is eliminated by the EWGF algorithm, while the integrity and stability of the edge structure are preserved.

4.3.3. Edge-Preserving Performance Analysis

Figure 15 presents three types of edge images used for edge-preserving performance evaluation. Figure 15a corresponds to the Ramp edge, Figure 15e corresponds to the Roof edge, and Figure 15h corresponds to the Line edge. These edge types cover common grayscale transition patterns in images, including gradual gradient transitions, sharp peak variations, and narrow edge variations.

Figure 15 presents the filtering results and local enlarged views of three representative edge types for evaluating the edge-preserving performance of different algorithms. Figure 15b–d correspond to the ramp edge and its corner region; Figure 15f,g correspond to the roof edge and its peak region; and Figure 15i,j correspond to the line edge and its enlarged details. In the figure, the black curve represents the original image, the blue curve represents the GF output, the red curve represents the SWGF output, and the green curve represents the output of the EWGF algorithm.

The comparison results indicate that GF exhibits deviations for all three edge types, demonstrating limited edge-preserving capability and resulting in edge blurring or structural information loss. SWGF improves upon GF, and its output curves are closer to those of the original image, indicating enhanced edge preservation. However, deviations still appear in some detailed regions. In contrast, the output curves of EWGF almost overlap with those of the original image. For ramp, roof, and line edges, the principal contours and peak details are preserved, demonstrating the edge-preserving capability and stability of the EWGF algorithm for different edge structures.

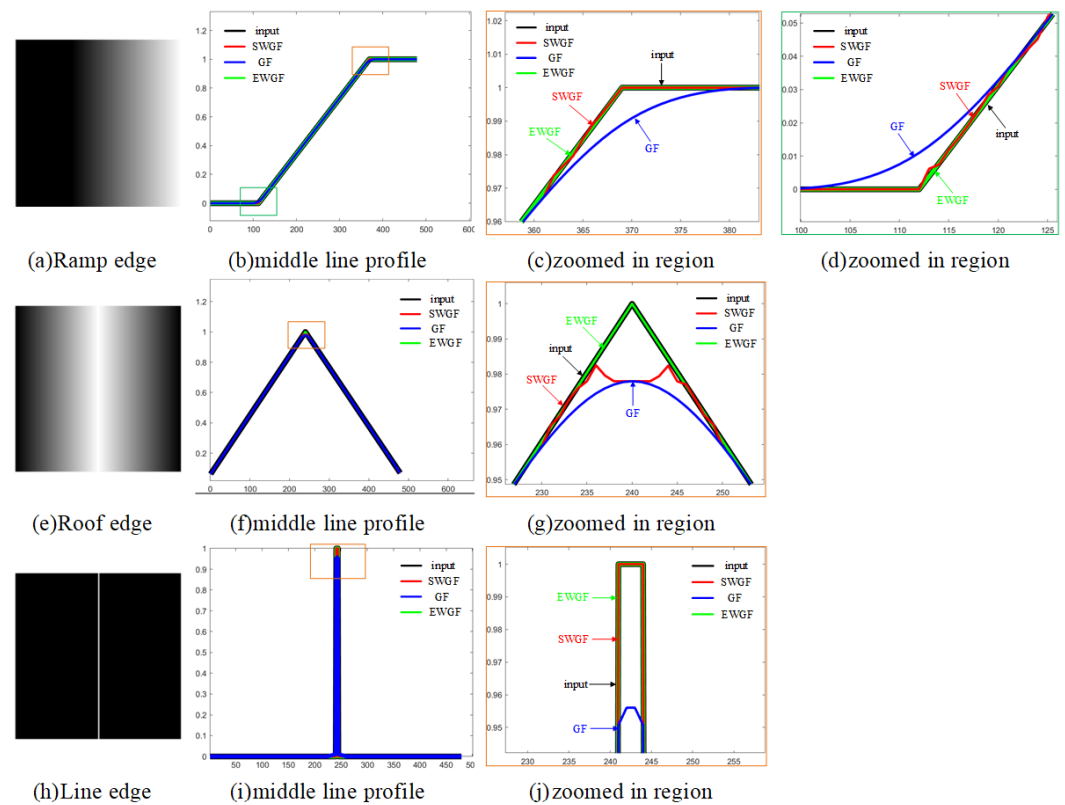


Figure 15. Comparison of test results for different types of 2D edge signals.

4.3.4. Analysis of Time Complexity

Table 4 presents a comparison of the runtime T_s of GF, SWGF, and EWGF under different parameter settings for filtering a 321×481 pixel image from the BSD500 dataset.

Table 4. Comparison of runtime T_s of different types of filtering algorithms under varying parameter settings.

ζ	Algorithm	T_s/s		
		$r = 2$	$r = 4$	$r = 8$
$\zeta = 0.1^2$	GF	0.313	0.296	0.300
	SWGF	5.183	5.879	7.084
	EWGF	3.745	4.005	4.172
$\zeta = 0.2^2$	GF	0.331	0.346	0.325
	SWGF	5.178	5.646	7.525
	EWGF	3.939	4.076	4.211
$\zeta = 0.4^2$	GF	0.336	0.326	0.278
	SWGF	5.245	5.587	7.315
	EWGF	3.936	4.031	4.157

In terms of runtime, the processing time of GF, SWGF, and EWGF increases with the regularization parameter ζ and the window radius r . GF achieves the highest computational efficiency but exhibits lower filtering performance. SWGF has the highest computational cost and provides edge-preserving capability. For example, when $r = 8$ and $\zeta = 0.2^2$, the runtime of SWGF is 7.525 s. EWGF balances filtering performance and computational efficiency. The incorporation of ETF construction and directional side-window sampling increases the runtime compared with GF. However, the runtime remains lower than that of SWGF under the same parameter settings, as shown in Table 4. The results indicate

that improvements in edge preservation and noise suppression are achieved without a corresponding increase in computational cost.

This document examines the time complexity of the EWGF algorithm in comparison to the GF and SWGF methods. The three methods are affected by three parameters during computation: the search window radius r , the pixel count in the image block n , and the regularization parameter ζ . The time complexity for a noisy grayscale or color image with dimensions MN is detailed in Table 5.

Table 5. Comparison of time complexity of the three algorithms.

Algorithm	Time Complexity
GF	$\sum_{i,j \in MN} n_{GF} \times r^2$
SWGF	$\sum_{i,j \in MN} n_{SWGF} \times r^2$
EWGF	$\sum_{i,j \in MN} n_{EWGF} \times r^2$

Figures 8 and 9 and the quantitative results in Table 2 show that EWGF performs grayscale and color image filtering under the same asymptotic time complexity as GF and SWGF. The results indicate improvements in edge preservation and noise suppression. Table 4 shows that the runtime of EWGF is lower than that of SWGF.

The computational cost of EWGF relative to GF is attributed to ETF construction and directional side window sampling, which are introduced for edge perception and structural preservation. The increase in computational cost is accompanied by improvements in filtering quality, edge preservation, and denoising performance. This work focuses on improving filtering quality and structural fidelity instead of real-time processing performance. Therefore, the current implementation is suitable for image processing tasks such as offline image enhancement, denoising, and structure-preserving image analysis, where filtering quality is more important than execution speed.

In conclusion, the performance of EWGF is evaluated in terms of PSNR, SSIM, and runtime T_s . Compared with GF and SWGF, EWGF achieves improvements in image quality and edge preservation, while its runtime is lower than that of SWGF. The results show that EWGF balances filtering performance, image quality preservation, and computational efficiency.

5. Conclusions

An edge-preserving guided filtering algorithm based on edge tangent flow and side window (EWGF) is introduced. The proposed method incorporates edge tangential flow analysis, local structure tensors, and an adaptive weighting mechanism. By converting traditional two-dimensional filtering methods into one-dimensional filtering along the tangential direction of the image structure, noise in the structural direction is reduced while edge sharpness at the borders is maintained. This approach addresses the common issues of edge blurring and structural degradation in complex gradient areas that are encountered by traditional guided filtering techniques. Comprehensive experimental evaluation confirms the superior performance of the EWGF algorithm in terms of both visual quality and objective metrics. When compared with the GF and SWGF algorithms, average improvements of 5.69 dB and 4.19 dB in PSNR, and 0.11 and 0.07 in SSIM, are achieved by the proposed method, respectively. Furthermore, a enhancement in computational efficiency of 33.6% is realized compared to the SWGF algorithm. The EWGF algorithm also exhibits superior structural preservation and noise suppression capabilities by processing complex structures that involve the ramp edge, the roof edge, and the line edge.

Although the EWGF algorithm achieves superior edge-preserving performance and structural fidelity, several limitations remain. First, the incorporation of edge tangent flow

estimation and side window optimization introduces additional computational complexity, which may limit the efficiency of the algorithm when processing high-resolution images or large-scale datasets. Second, the filtering performance is influenced by the selection of parameters, and suboptimal parameter settings may reduce the balance between edge preservation and smoothing capability. Furthermore, the current method has been primarily validated on natural image datasets, and its generalization to other imaging modalities, such as medical, remote sensing, or low-light images, requires further investigation.

Future work will focus on developing adaptive parameter selection mechanisms based on image characteristics (e.g., local structure and ETF directional coherence), accelerating the computational framework, and extending the proposed method to a broader range of image processing applications.

Author Contributions: Conceptualization, T.L.; Methodology, T.L.; Software, T.L.; Validation, T.L.; Formal analysis, T.L.; Investigation, T.L.; Resources, P.C.; Data curation, P.C.; Writing—original draft preparation, T.L.; Writing—review and editing, P.C. and T.L.; Visualization, T.L.; Supervision, P.C.; Project administration, P.C.; funding acquisition, not applicable. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding. The APC was funded by the authors.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The BSD500, Set14, and Set5 datasets used in this study are publicly available. The experimental data generated during the current study are available from the authors upon reasonable request.

Acknowledgments: The authors thank the developers of the BSD500, Set14, and Set5 datasets. During the preparation of this study, the authors used ChatGPT (version GPT-5.5) for spelling and grammar checking. The authors have reviewed and edited the output and take full responsibility for the content of this publication.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

EWGF	Guided filtering based on edge tangent flow and side window
SWGF	Side window guided filtering
GF	Guided filtering
ETF	Edge tangent flow
PSNR	Peak signal-to-noise ratio
SSIM	Structural similarity index measure

Notations

The principal notations employed in this work are defined as follows:

I	Input image
F	Output image
G	Guidance image
$\mu_{i,j}$	Local mean
$\sigma_{i,j}^2$	Local variance
r	Local window radius
ζ	Regularization parameter
a, b	Linear coefficients in the guided filter

γ	Edge-aware weights
ζ_p, τ_p	Tangent and normal eigenvectors
$\lambda_{1,p}, \lambda_{2,p}$	Tangent and normal eigenvalues
$\omega_{i,j}$	Local window pixel set

References

1. Cho, H.; Lee, H.; Kang, H.; Lee, S. Bilateral texture filtering. *ACM Trans. Graph.* **2014**, *33*, 1–8. [\[CrossRef\]](#)
2. Yin, H.; Gong, Y.H.; Qiu, G.P. Side window filtering. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 8758–8766.
3. Zhang, Q.; Shen, X.Y.; Xu, L.; Jia, J.Y. Rolling guidance filter. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 815–830.
4. Xu, L.; Yan, Q.; Xia, Y.; Jia, J.Y. Structure extraction from texture via relative total variation. *ACM Trans. Graph.* **2012**, *31*, 1–10. [\[CrossRef\]](#)
5. He, K.M.; Sun, J.; Tang, X.O. Guided image filtering. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1397–1409. [\[CrossRef\]](#) [\[PubMed\]](#)
6. Yin, H.; Gong, Y.H.; Qiu, G.P. Side window guided filtering. *Signal Process.* **2019**, *165*, 315–330. [\[CrossRef\]](#)
7. Li, Z.; Zheng, J.; Zhu, Z.; Yao, W.; Wu, S. Weighted guided image filtering. *IEEE Trans. Image Process.* **2015**, *24*, 120–129. [\[CrossRef\]](#) [\[PubMed\]](#)
8. Kou, F.; Chen, W.; Wen, C.; Li, Z. Gradient domain guided image filtering. *IEEE Trans. Image Process.* **2015**, *24*, 4528–4539. [\[CrossRef\]](#) [\[PubMed\]](#)
9. Wang, B.; Wang, Y.H.; Sui, X.B.; Liu, Y.; Chen, Q. Gradient domain weighted guided image filtering. *Signal Image Video Process.* **2023**, *17*, 4097–4105. [\[CrossRef\]](#)
10. Liu, W.; Zhang, P.P.; Chen, X.G.; Shen, C.; Huang, X.; Yang, J. Embedding bilateral filter in least squares for efficient edge-preserving image smoothing. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *30*, 23–35. [\[CrossRef\]](#)
11. Bui, K.; Lou, Y.F.; Park, F.; Xin, J. Difference of anisotropic and isotropic TV for segmentation under blur and Poisson noise. *Front. Comput. Sci.* **2023**, *5*, 1131317. [\[CrossRef\]](#)
12. Sun, B.C.; Qi, Y.H.; Zhang, G.Y.; Liu, Y. Edge guidance filtering for structure extraction. *Vis. Comput.* **2023**, *39*, 5327–5342.
13. Li, J. Enhanced edge-perceptual guided image filtering. *arXiv* **2023**, arXiv:2310.10387.
14. Long, J.W.; Chen, S.; Liu, Y.Q.; Zhang, K.; Luo, Q. Adaptive dynamic guided image filtering for edge preservation and structure extraction. *Signal Process.* **2026**, *239*, 110290. [\[CrossRef\]](#)
15. Jacumin, T.; Langer, A. An adaptive finite difference method for total variation minimization. *Numer. Algorithms* **2026**, *101*, 1337–1372.
16. Zhao, L.; He, C.J. Gaussian highpass guided image filtering. *Digit. Signal Process.* **2025**, *165*, 105344. [\[CrossRef\]](#)
17. Lu, P.; Mu, Y.; Gu, C.J.; Fu, S.; Cheng, Q.; Zhao, K.; Shen, X. Multi-scale infrared image enhancement based on non-uniform weighted guided filtering. *Opt. Lasers Eng.* **2025**, *186*, 108797. [\[CrossRef\]](#)
18. Ma, M.H.; Hu, X.J.; Zhang, R.P.; Chen, C.; Yu, H. Rendered image denoising method with filtering guided by lighting information. *Optoelectron. Lett.* **2025**, *21*, 242–248. [\[CrossRef\]](#)
19. Weiss, B. Fast isotropic median filtering. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*; ACM: New York, NY, USA, 2025; pp. 1–10.
20. Yang, Y.; Wu, D.; Zeng, L.; Li, Z. Weighted least square filter via deep unsupervised learning. *Multimed. Tools Appl.* **2024**, *83*, 31361–31377.
21. Liu, W.; Zhang, P.P.; Qin, H.X.; Huang, X.; Yang, J.; Ng, M. Fast global image smoothing via quasi weighted least squares. *Int. J. Comput. Vis.* **2024**, *132*, 6039–6068. [\[CrossRef\]](#)
22. Yang, Y.; Ma, W.; Gao, W.; Zeng, L.; Pan, Z. Hyperbolic tangent penalty function for edge-preserving image filtering. *Digit. Signal Process.* **2024**, *153*, 104591. [\[CrossRef\]](#)
23. Mishiba, K. Intensity and scale adjustable edge-preserving smoothing filter. *IEEE Access* **2024**, *12*, 89183–89190. [\[CrossRef\]](#)
24. Galetto, F.J.; Deng, G.; Al-Nasrawi, M.; Waheed, W. Edge-aware filter based on adaptive patch variance weighted average. *IEEE Access* **2021**, *9*, 118291–118306. [\[CrossRef\]](#)
25. Pradhan, K.; Patra, S. Semantic-aware structure-preserving median morpho-filtering. *Vis. Comput.* **2024**, *40*, 505–521.
26. Liu, W.; Zhang, P.P.; Lei, Y.J.; Huang, X.; Yang, J.; Ng, M. A generalized framework for edge-preserving and structure-preserving image smoothing. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 6631–6648. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.